

NVIDIA Performance Primitives (NPP)
Version 8.0

January 28, 2016

Contents

| | | |
|----------|--|-----------|
| 1 | NVIDIA Performance Primitives | 1 |
| 1.1 | What is NPP? | 2 |
| 1.2 | Documentation | 2 |
| 1.3 | Technical Specifications | 3 |
| 1.4 | Files | 3 |
| 1.4.1 | Header Files | 3 |
| 1.4.2 | Library Files | 3 |
| 1.5 | Supported NVIDIA Hardware | 4 |
| 2 | General API Conventions | 5 |
| 2.1 | Memory Management | 6 |
| 2.1.1 | Scratch Buffer and Host Pointer | 6 |
| 2.2 | Function Naming | 7 |
| 2.3 | Integer Result Scaling | 7 |
| 2.4 | Rounding Modes | 8 |
| 2.4.1 | Rounding Mode Parameter | 8 |
| 3 | Signal-Processing Specific API Conventions | 9 |
| 3.1 | Signal Data | 10 |
| 3.1.1 | Parameter Names for Signal Data | 10 |
| 3.1.1.1 | Source Signal Pointer | 10 |
| 3.1.1.2 | Destination Signal Pointer | 10 |
| 3.1.1.3 | In-Place Signal Pointer | 10 |
| 3.1.2 | Signal Data Alignment Requirements | 11 |
| 3.1.3 | Signal Data Related Error Codes | 11 |
| 3.2 | Signal Length | 11 |
| 3.2.1 | Length Related Error Codes | 11 |
| 4 | Imaging-Processing Specific API Conventions | 13 |

| | | |
|----------|--|-----------|
| 4.1 | Function Naming | 14 |
| 4.2 | Image Data | 14 |
| 4.2.1 | Line Step | 15 |
| 4.2.2 | Parameter Names for Image Data | 15 |
| 4.2.2.1 | Passing Source-Image Data | 15 |
| 4.2.2.2 | Passing Destination-Image Data | 16 |
| 4.2.2.3 | Passing In-Place Image Data | 18 |
| 4.2.2.4 | Passing Mask-Image Data | 18 |
| 4.2.2.5 | Passing Channel-of-Interest Data | 18 |
| 4.2.3 | Image Data Alignment Requirements | 18 |
| 4.2.4 | Image Data Related Error Codes | 19 |
| 4.3 | Region-of-Interest (ROI) | 19 |
| 4.3.1 | ROI Related Error Codes | 19 |
| 4.4 | Masked Operation | 20 |
| 4.5 | Channel-of-Interest API | 20 |
| 4.5.1 | Select-Channel Source-Image Pointer | 20 |
| 4.5.2 | Select-Channel Source-Image | 20 |
| 4.5.3 | Select-Channel Destination-Image Pointer | 20 |
| 4.6 | Source-Image Sampling | 21 |
| 4.6.1 | Point-Wise Operations | 21 |
| 4.6.2 | Neighborhood Operations | 21 |
| 4.6.2.1 | Mask-Size Parameter | 21 |
| 4.6.2.2 | Anchor-Point Parameter | 22 |
| 4.6.2.3 | Sampling Beyond Image Boundaries | 22 |
| 5 | Module Index | 23 |
| 5.1 | Modules | 23 |
| 6 | Data Structure Index | 25 |
| 6.1 | Data Structures | 25 |
| 7 | Module Documentation | 27 |
| 7.1 | NPP Core | 27 |
| 7.1.1 | Detailed Description | 28 |
| 7.1.2 | Function Documentation | 28 |
| 7.1.2.1 | nppGetGpuComputeCapability | 28 |
| 7.1.2.2 | nppGetGpuDeviceProperties | 28 |
| 7.1.2.3 | nppGetGpuName | 28 |

| | | |
|----------|------------------------------------|----|
| 7.1.2.4 | nppGetGpuNumSMs | 28 |
| 7.1.2.5 | nppGetLibVersion | 29 |
| 7.1.2.6 | nppGetMaxThreadsPerBlock | 29 |
| 7.1.2.7 | nppGetMaxThreadsPerSM | 29 |
| 7.1.2.8 | nppGetStream | 29 |
| 7.1.2.9 | nppGetStreamMaxThreadsPerSM | 29 |
| 7.1.2.10 | nppGetStreamNumSMs | 29 |
| 7.1.2.11 | nppSetStream | 30 |
| 7.2 | NPP Type Definitions and Constants | 31 |
| 7.2.1 | Define Documentation | 37 |
| 7.2.1.1 | NPP_MAX_16S | 37 |
| 7.2.1.2 | NPP_MAX_16U | 37 |
| 7.2.1.3 | NPP_MAX_32S | 37 |
| 7.2.1.4 | NPP_MAX_32U | 37 |
| 7.2.1.5 | NPP_MAX_64S | 37 |
| 7.2.1.6 | NPP_MAX_64U | 37 |
| 7.2.1.7 | NPP_MAX_8S | 37 |
| 7.2.1.8 | NPP_MAX_8U | 37 |
| 7.2.1.9 | NPP_MAXABS_32F | 37 |
| 7.2.1.10 | NPP_MAXABS_64F | 37 |
| 7.2.1.11 | NPP_MIN_16S | 38 |
| 7.2.1.12 | NPP_MIN_16U | 38 |
| 7.2.1.13 | NPP_MIN_32S | 38 |
| 7.2.1.14 | NPP_MIN_32U | 38 |
| 7.2.1.15 | NPP_MIN_64S | 38 |
| 7.2.1.16 | NPP_MIN_64U | 38 |
| 7.2.1.17 | NPP_MIN_8S | 38 |
| 7.2.1.18 | NPP_MIN_8U | 38 |
| 7.2.1.19 | NPP_MINABS_32F | 38 |
| 7.2.1.20 | NPP_MINABS_64F | 38 |
| 7.2.2 | Enumeration Type Documentation | 38 |
| 7.2.2.1 | NppCmpOp | 38 |
| 7.2.2.2 | NppGpuComputeCapability | 39 |
| 7.2.2.3 | NppHintAlgorithm | 39 |
| 7.2.2.4 | NppiAlphaOp | 39 |
| 7.2.2.5 | NppiAxis | 40 |

| | | |
|----------|------------------------|----|
| 7.2.2.6 | NppiBayerGridPosition | 40 |
| 7.2.2.7 | NppiBorderType | 40 |
| 7.2.2.8 | NppiDifferentialKernel | 40 |
| 7.2.2.9 | NppiHuffmanTableType | 41 |
| 7.2.2.10 | NppiInterpolationMode | 41 |
| 7.2.2.11 | NppiMaskSize | 41 |
| 7.2.2.12 | NppiNorm | 42 |
| 7.2.2.13 | NppRoundMode | 42 |
| 7.2.2.14 | NppStatus | 43 |
| 7.2.2.15 | NppsZCType | 45 |
| 7.3 | Basic NPP Data Types | 46 |
| 7.3.1 | Typedef Documentation | 47 |
| 7.3.1.1 | Npp16s | 47 |
| 7.3.1.2 | Npp16u | 47 |
| 7.3.1.3 | Npp32f | 47 |
| 7.3.1.4 | Npp32fc | 47 |
| 7.3.1.5 | Npp32s | 47 |
| 7.3.1.6 | Npp32sc | 48 |
| 7.3.1.7 | Npp32u | 48 |
| 7.3.1.8 | Npp32uc | 48 |
| 7.3.1.9 | Npp64f | 48 |
| 7.3.1.10 | Npp64fc | 48 |
| 7.3.1.11 | Npp64s | 48 |
| 7.3.1.12 | Npp64sc | 48 |
| 7.3.1.13 | Npp64u | 48 |
| 7.3.1.14 | Npp8s | 48 |
| 7.3.1.15 | Npp8u | 48 |
| 7.3.2 | Function Documentation | 48 |
| 7.3.2.1 | __align__ | 48 |
| 7.3.2.2 | __align__ | 49 |
| 7.3.3 | Variable Documentation | 49 |
| 7.3.3.1 | Npp16sc | 49 |
| 7.3.3.2 | Npp16uc | 49 |
| 7.3.3.3 | Npp8uc | 49 |
| 7.4 | Memory Management | 50 |
| 7.4.1 | Detailed Description | 52 |

| | | |
|----------|----------------------------------|----|
| 7.4.2 | Function Documentation | 52 |
| 7.4.2.1 | nppiFree | 52 |
| 7.4.2.2 | nppiMalloc_16s_C1 | 52 |
| 7.4.2.3 | nppiMalloc_16s_C2 | 53 |
| 7.4.2.4 | nppiMalloc_16s_C4 | 53 |
| 7.4.2.5 | nppiMalloc_16sc_C1 | 53 |
| 7.4.2.6 | nppiMalloc_16sc_C2 | 53 |
| 7.4.2.7 | nppiMalloc_16sc_C3 | 54 |
| 7.4.2.8 | nppiMalloc_16sc_C4 | 54 |
| 7.4.2.9 | nppiMalloc_16u_C1 | 54 |
| 7.4.2.10 | nppiMalloc_16u_C2 | 55 |
| 7.4.2.11 | nppiMalloc_16u_C3 | 55 |
| 7.4.2.12 | nppiMalloc_16u_C4 | 55 |
| 7.4.2.13 | nppiMalloc_32f_C1 | 55 |
| 7.4.2.14 | nppiMalloc_32f_C2 | 56 |
| 7.4.2.15 | nppiMalloc_32f_C3 | 56 |
| 7.4.2.16 | nppiMalloc_32f_C4 | 56 |
| 7.4.2.17 | nppiMalloc_32fc_C1 | 57 |
| 7.4.2.18 | nppiMalloc_32fc_C2 | 57 |
| 7.4.2.19 | nppiMalloc_32fc_C3 | 57 |
| 7.4.2.20 | nppiMalloc_32fc_C4 | 57 |
| 7.4.2.21 | nppiMalloc_32s_C1 | 58 |
| 7.4.2.22 | nppiMalloc_32s_C3 | 58 |
| 7.4.2.23 | nppiMalloc_32s_C4 | 58 |
| 7.4.2.24 | nppiMalloc_32sc_C1 | 59 |
| 7.4.2.25 | nppiMalloc_32sc_C2 | 59 |
| 7.4.2.26 | nppiMalloc_32sc_C3 | 59 |
| 7.4.2.27 | nppiMalloc_32sc_C4 | 59 |
| 7.4.2.28 | nppiMalloc_8u_C1 | 60 |
| 7.4.2.29 | nppiMalloc_8u_C2 | 60 |
| 7.4.2.30 | nppiMalloc_8u_C3 | 60 |
| 7.4.2.31 | nppiMalloc_8u_C4 | 61 |
| 7.5 | Data Exchange and Initialization | 62 |
| 7.5.1 | Detailed Description | 62 |
| 7.6 | Set | 63 |
| 7.6.1 | Detailed Description | 69 |

| | | |
|----------|------------------------|----|
| 7.6.2 | Function Documentation | 69 |
| 7.6.2.1 | nppiSet_16s_AC4MR | 69 |
| 7.6.2.2 | nppiSet_16s_AC4R | 70 |
| 7.6.2.3 | nppiSet_16s_C1MR | 70 |
| 7.6.2.4 | nppiSet_16s_C1R | 70 |
| 7.6.2.5 | nppiSet_16s_C2R | 71 |
| 7.6.2.6 | nppiSet_16s_C3CR | 71 |
| 7.6.2.7 | nppiSet_16s_C3MR | 71 |
| 7.6.2.8 | nppiSet_16s_C3R | 72 |
| 7.6.2.9 | nppiSet_16s_C4CR | 72 |
| 7.6.2.10 | nppiSet_16s_C4MR | 72 |
| 7.6.2.11 | nppiSet_16s_C4R | 73 |
| 7.6.2.12 | nppiSet_16sc_AC4R | 73 |
| 7.6.2.13 | nppiSet_16sc_C1R | 73 |
| 7.6.2.14 | nppiSet_16sc_C2R | 74 |
| 7.6.2.15 | nppiSet_16sc_C3R | 74 |
| 7.6.2.16 | nppiSet_16sc_C4R | 74 |
| 7.6.2.17 | nppiSet_16u_AC4MR | 75 |
| 7.6.2.18 | nppiSet_16u_AC4R | 75 |
| 7.6.2.19 | nppiSet_16u_C1MR | 75 |
| 7.6.2.20 | nppiSet_16u_C1R | 76 |
| 7.6.2.21 | nppiSet_16u_C2R | 76 |
| 7.6.2.22 | nppiSet_16u_C3CR | 76 |
| 7.6.2.23 | nppiSet_16u_C3MR | 77 |
| 7.6.2.24 | nppiSet_16u_C3R | 77 |
| 7.6.2.25 | nppiSet_16u_C4CR | 77 |
| 7.6.2.26 | nppiSet_16u_C4MR | 78 |
| 7.6.2.27 | nppiSet_16u_C4R | 78 |
| 7.6.2.28 | nppiSet_32f_AC4MR | 78 |
| 7.6.2.29 | nppiSet_32f_AC4R | 79 |
| 7.6.2.30 | nppiSet_32f_C1MR | 79 |
| 7.6.2.31 | nppiSet_32f_C1R | 80 |
| 7.6.2.32 | nppiSet_32f_C2R | 80 |
| 7.6.2.33 | nppiSet_32f_C3CR | 80 |
| 7.6.2.34 | nppiSet_32f_C3MR | 81 |
| 7.6.2.35 | nppiSet_32f_C3R | 81 |

| | | |
|----------|-------------------|----|
| 7.6.2.36 | nppiSet_32f_C4CR | 81 |
| 7.6.2.37 | nppiSet_32f_C4MR | 82 |
| 7.6.2.38 | nppiSet_32f_C4R | 82 |
| 7.6.2.39 | nppiSet_32fc_AC4R | 82 |
| 7.6.2.40 | nppiSet_32fc_C1R | 83 |
| 7.6.2.41 | nppiSet_32fc_C2R | 83 |
| 7.6.2.42 | nppiSet_32fc_C3R | 83 |
| 7.6.2.43 | nppiSet_32fc_C4R | 84 |
| 7.6.2.44 | nppiSet_32s_AC4MR | 84 |
| 7.6.2.45 | nppiSet_32s_AC4R | 84 |
| 7.6.2.46 | nppiSet_32s_C1MR | 85 |
| 7.6.2.47 | nppiSet_32s_C1R | 85 |
| 7.6.2.48 | nppiSet_32s_C2R | 85 |
| 7.6.2.49 | nppiSet_32s_C3CR | 86 |
| 7.6.2.50 | nppiSet_32s_C3MR | 86 |
| 7.6.2.51 | nppiSet_32s_C3R | 86 |
| 7.6.2.52 | nppiSet_32s_C4CR | 87 |
| 7.6.2.53 | nppiSet_32s_C4MR | 87 |
| 7.6.2.54 | nppiSet_32s_C4R | 87 |
| 7.6.2.55 | nppiSet_32sc_AC4R | 88 |
| 7.6.2.56 | nppiSet_32sc_C1R | 88 |
| 7.6.2.57 | nppiSet_32sc_C2R | 88 |
| 7.6.2.58 | nppiSet_32sc_C3R | 89 |
| 7.6.2.59 | nppiSet_32sc_C4R | 89 |
| 7.6.2.60 | nppiSet_32u_AC4R | 89 |
| 7.6.2.61 | nppiSet_32u_C1R | 90 |
| 7.6.2.62 | nppiSet_32u_C2R | 90 |
| 7.6.2.63 | nppiSet_32u_C3R | 90 |
| 7.6.2.64 | nppiSet_32u_C4R | 91 |
| 7.6.2.65 | nppiSet_8s_AC4R | 91 |
| 7.6.2.66 | nppiSet_8s_C1R | 91 |
| 7.6.2.67 | nppiSet_8s_C2R | 92 |
| 7.6.2.68 | nppiSet_8s_C3R | 92 |
| 7.6.2.69 | nppiSet_8s_C4R | 92 |
| 7.6.2.70 | nppiSet_8u_AC4MR | 93 |
| 7.6.2.71 | nppiSet_8u_AC4R | 93 |

| | | |
|----------|------------------------|-----|
| 7.6.2.72 | nppiSet_8u_C1MR | 93 |
| 7.6.2.73 | nppiSet_8u_C1R | 94 |
| 7.6.2.74 | nppiSet_8u_C2R | 94 |
| 7.6.2.75 | nppiSet_8u_C3CR | 94 |
| 7.6.2.76 | nppiSet_8u_C3MR | 95 |
| 7.6.2.77 | nppiSet_8u_C3R | 95 |
| 7.6.2.78 | nppiSet_8u_C4CR | 95 |
| 7.6.2.79 | nppiSet_8u_C4MR | 96 |
| 7.6.2.80 | nppiSet_8u_C4R | 96 |
| 7.7 | Copy | 97 |
| 7.7.1 | Function Documentation | 106 |
| 7.7.1.1 | nppiCopy_16s_AC4MR | 106 |
| 7.7.1.2 | nppiCopy_16s_AC4R | 107 |
| 7.7.1.3 | nppiCopy_16s_C1C3R | 107 |
| 7.7.1.4 | nppiCopy_16s_C1C4R | 108 |
| 7.7.1.5 | nppiCopy_16s_C1MR | 108 |
| 7.7.1.6 | nppiCopy_16s_C1R | 108 |
| 7.7.1.7 | nppiCopy_16s_C3C1R | 109 |
| 7.7.1.8 | nppiCopy_16s_C3CR | 109 |
| 7.7.1.9 | nppiCopy_16s_C3MR | 109 |
| 7.7.1.10 | nppiCopy_16s_C3P3R | 110 |
| 7.7.1.11 | nppiCopy_16s_C3R | 110 |
| 7.7.1.12 | nppiCopy_16s_C4C1R | 110 |
| 7.7.1.13 | nppiCopy_16s_C4CR | 111 |
| 7.7.1.14 | nppiCopy_16s_C4MR | 111 |
| 7.7.1.15 | nppiCopy_16s_C4P4R | 111 |
| 7.7.1.16 | nppiCopy_16s_C4R | 112 |
| 7.7.1.17 | nppiCopy_16s_P3C3R | 112 |
| 7.7.1.18 | nppiCopy_16s_P4C4R | 112 |
| 7.7.1.19 | nppiCopy_16sc_AC4R | 113 |
| 7.7.1.20 | nppiCopy_16sc_C1R | 113 |
| 7.7.1.21 | nppiCopy_16sc_C2R | 113 |
| 7.7.1.22 | nppiCopy_16sc_C3R | 114 |
| 7.7.1.23 | nppiCopy_16sc_C4R | 114 |
| 7.7.1.24 | nppiCopy_16u_AC4MR | 114 |
| 7.7.1.25 | nppiCopy_16u_AC4R | 115 |

| | | |
|----------|--------------------|-----|
| 7.7.1.26 | nppiCopy_16u_C1C3R | 115 |
| 7.7.1.27 | nppiCopy_16u_C1C4R | 115 |
| 7.7.1.28 | nppiCopy_16u_C1MR | 116 |
| 7.7.1.29 | nppiCopy_16u_C1R | 116 |
| 7.7.1.30 | nppiCopy_16u_C3C1R | 116 |
| 7.7.1.31 | nppiCopy_16u_C3CR | 117 |
| 7.7.1.32 | nppiCopy_16u_C3MR | 117 |
| 7.7.1.33 | nppiCopy_16u_C3P3R | 117 |
| 7.7.1.34 | nppiCopy_16u_C3R | 118 |
| 7.7.1.35 | nppiCopy_16u_C4C1R | 118 |
| 7.7.1.36 | nppiCopy_16u_C4CR | 118 |
| 7.7.1.37 | nppiCopy_16u_C4MR | 119 |
| 7.7.1.38 | nppiCopy_16u_C4P4R | 119 |
| 7.7.1.39 | nppiCopy_16u_C4R | 119 |
| 7.7.1.40 | nppiCopy_16u_P3C3R | 120 |
| 7.7.1.41 | nppiCopy_16u_P4C4R | 120 |
| 7.7.1.42 | nppiCopy_32f_AC4MR | 120 |
| 7.7.1.43 | nppiCopy_32f_AC4R | 121 |
| 7.7.1.44 | nppiCopy_32f_C1C3R | 121 |
| 7.7.1.45 | nppiCopy_32f_C1C4R | 121 |
| 7.7.1.46 | nppiCopy_32f_C1MR | 122 |
| 7.7.1.47 | nppiCopy_32f_C1R | 122 |
| 7.7.1.48 | nppiCopy_32f_C3C1R | 122 |
| 7.7.1.49 | nppiCopy_32f_C3CR | 123 |
| 7.7.1.50 | nppiCopy_32f_C3MR | 123 |
| 7.7.1.51 | nppiCopy_32f_C3P3R | 123 |
| 7.7.1.52 | nppiCopy_32f_C3R | 124 |
| 7.7.1.53 | nppiCopy_32f_C4C1R | 124 |
| 7.7.1.54 | nppiCopy_32f_C4CR | 124 |
| 7.7.1.55 | nppiCopy_32f_C4MR | 125 |
| 7.7.1.56 | nppiCopy_32f_C4P4R | 125 |
| 7.7.1.57 | nppiCopy_32f_C4R | 125 |
| 7.7.1.58 | nppiCopy_32f_P3C3R | 126 |
| 7.7.1.59 | nppiCopy_32f_P4C4R | 126 |
| 7.7.1.60 | nppiCopy_32fc_AC4R | 126 |
| 7.7.1.61 | nppiCopy_32fc_C1R | 127 |

| | | |
|----------|--------------------|-----|
| 7.7.1.62 | nppiCopy_32fc_C2R | 127 |
| 7.7.1.63 | nppiCopy_32fc_C3R | 127 |
| 7.7.1.64 | nppiCopy_32fc_C4R | 128 |
| 7.7.1.65 | nppiCopy_32s_AC4MR | 128 |
| 7.7.1.66 | nppiCopy_32s_AC4R | 128 |
| 7.7.1.67 | nppiCopy_32s_C1C3R | 129 |
| 7.7.1.68 | nppiCopy_32s_C1C4R | 129 |
| 7.7.1.69 | nppiCopy_32s_C1MR | 129 |
| 7.7.1.70 | nppiCopy_32s_C1R | 130 |
| 7.7.1.71 | nppiCopy_32s_C3C1R | 130 |
| 7.7.1.72 | nppiCopy_32s_C3CR | 130 |
| 7.7.1.73 | nppiCopy_32s_C3MR | 131 |
| 7.7.1.74 | nppiCopy_32s_C3P3R | 131 |
| 7.7.1.75 | nppiCopy_32s_C3R | 131 |
| 7.7.1.76 | nppiCopy_32s_C4C1R | 132 |
| 7.7.1.77 | nppiCopy_32s_C4CR | 132 |
| 7.7.1.78 | nppiCopy_32s_C4MR | 132 |
| 7.7.1.79 | nppiCopy_32s_C4P4R | 133 |
| 7.7.1.80 | nppiCopy_32s_C4R | 133 |
| 7.7.1.81 | nppiCopy_32s_P3C3R | 133 |
| 7.7.1.82 | nppiCopy_32s_P4C4R | 134 |
| 7.7.1.83 | nppiCopy_32sc_AC4R | 134 |
| 7.7.1.84 | nppiCopy_32sc_C1R | 134 |
| 7.7.1.85 | nppiCopy_32sc_C2R | 135 |
| 7.7.1.86 | nppiCopy_32sc_C3R | 135 |
| 7.7.1.87 | nppiCopy_32sc_C4R | 135 |
| 7.7.1.88 | nppiCopy_8s_AC4R | 136 |
| 7.7.1.89 | nppiCopy_8s_C1R | 136 |
| 7.7.1.90 | nppiCopy_8s_C2R | 136 |
| 7.7.1.91 | nppiCopy_8s_C3R | 137 |
| 7.7.1.92 | nppiCopy_8s_C4R | 137 |
| 7.7.1.93 | nppiCopy_8u_AC4MR | 137 |
| 7.7.1.94 | nppiCopy_8u_AC4R | 138 |
| 7.7.1.95 | nppiCopy_8u_C1C3R | 138 |
| 7.7.1.96 | nppiCopy_8u_C1C4R | 138 |
| 7.7.1.97 | nppiCopy_8u_C1MR | 139 |

| | | |
|-----------|---------------------------|-----|
| 7.7.1.98 | nppiCopy_8u_C1R | 139 |
| 7.7.1.99 | nppiCopy_8u_C3C1R | 139 |
| 7.7.1.100 | nppiCopy_8u_C3CR | 140 |
| 7.7.1.101 | nppiCopy_8u_C3MR | 140 |
| 7.7.1.102 | nppiCopy_8u_C3P3R | 140 |
| 7.7.1.103 | nppiCopy_8u_C3R | 141 |
| 7.7.1.104 | nppiCopy_8u_C4C1R | 141 |
| 7.7.1.105 | nppiCopy_8u_C4CR | 141 |
| 7.7.1.106 | nppiCopy_8u_C4MR | 142 |
| 7.7.1.107 | nppiCopy_8u_C4P4R | 142 |
| 7.7.1.108 | nppiCopy_8u_C4R | 142 |
| 7.7.1.109 | nppiCopy_8u_P3C3R | 143 |
| 7.7.1.110 | nppiCopy_8u_P4C4R | 143 |
| 7.8 | Convert | 144 |
| 7.8.1 | Function Documentation | 152 |
| 7.8.1.1 | nppiConvert_16s16u_C1Rs | 152 |
| 7.8.1.2 | nppiConvert_16s32f_AC4R | 152 |
| 7.8.1.3 | nppiConvert_16s32f_C1R | 153 |
| 7.8.1.4 | nppiConvert_16s32f_C3R | 153 |
| 7.8.1.5 | nppiConvert_16s32f_C4R | 153 |
| 7.8.1.6 | nppiConvert_16s32s_AC4R | 154 |
| 7.8.1.7 | nppiConvert_16s32s_C1R | 154 |
| 7.8.1.8 | nppiConvert_16s32s_C3R | 154 |
| 7.8.1.9 | nppiConvert_16s32s_C4R | 155 |
| 7.8.1.10 | nppiConvert_16s32u_C1Rs | 155 |
| 7.8.1.11 | nppiConvert_16s8s_C1RSfs | 155 |
| 7.8.1.12 | nppiConvert_16s8u_AC4R | 156 |
| 7.8.1.13 | nppiConvert_16s8u_C1R | 156 |
| 7.8.1.14 | nppiConvert_16s8u_C3R | 157 |
| 7.8.1.15 | nppiConvert_16s8u_C4R | 157 |
| 7.8.1.16 | nppiConvert_16u16s_C1RSfs | 157 |
| 7.8.1.17 | nppiConvert_16u32f_AC4R | 158 |
| 7.8.1.18 | nppiConvert_16u32f_C1R | 158 |
| 7.8.1.19 | nppiConvert_16u32f_C3R | 158 |
| 7.8.1.20 | nppiConvert_16u32f_C4R | 159 |
| 7.8.1.21 | nppiConvert_16u32s_AC4R | 159 |

| | | |
|----------|---------------------------|-----|
| 7.8.1.22 | nppiConvert_16u32s_C1R | 159 |
| 7.8.1.23 | nppiConvert_16u32s_C3R | 160 |
| 7.8.1.24 | nppiConvert_16u32s_C4R | 160 |
| 7.8.1.25 | nppiConvert_16u32u_C1R | 160 |
| 7.8.1.26 | nppiConvert_16u8s_C1RSfs | 161 |
| 7.8.1.27 | nppiConvert_16u8u_AC4R | 161 |
| 7.8.1.28 | nppiConvert_16u8u_C1R | 161 |
| 7.8.1.29 | nppiConvert_16u8u_C3R | 162 |
| 7.8.1.30 | nppiConvert_16u8u_C4R | 162 |
| 7.8.1.31 | nppiConvert_32f16s_AC4R | 162 |
| 7.8.1.32 | nppiConvert_32f16s_C1R | 163 |
| 7.8.1.33 | nppiConvert_32f16s_C1RSfs | 163 |
| 7.8.1.34 | nppiConvert_32f16s_C3R | 163 |
| 7.8.1.35 | nppiConvert_32f16s_C4R | 164 |
| 7.8.1.36 | nppiConvert_32f16u_AC4R | 164 |
| 7.8.1.37 | nppiConvert_32f16u_C1R | 165 |
| 7.8.1.38 | nppiConvert_32f16u_C1RSfs | 165 |
| 7.8.1.39 | nppiConvert_32f16u_C3R | 165 |
| 7.8.1.40 | nppiConvert_32f16u_C4R | 166 |
| 7.8.1.41 | nppiConvert_32f32s_C1RSfs | 166 |
| 7.8.1.42 | nppiConvert_32f32u_C1RSfs | 167 |
| 7.8.1.43 | nppiConvert_32f8s_AC4R | 167 |
| 7.8.1.44 | nppiConvert_32f8s_C1R | 167 |
| 7.8.1.45 | nppiConvert_32f8s_C1RSfs | 168 |
| 7.8.1.46 | nppiConvert_32f8s_C3R | 168 |
| 7.8.1.47 | nppiConvert_32f8s_C4R | 169 |
| 7.8.1.48 | nppiConvert_32f8u_AC4R | 169 |
| 7.8.1.49 | nppiConvert_32f8u_C1R | 169 |
| 7.8.1.50 | nppiConvert_32f8u_C1RSfs | 170 |
| 7.8.1.51 | nppiConvert_32f8u_C3R | 170 |
| 7.8.1.52 | nppiConvert_32f8u_C4R | 170 |
| 7.8.1.53 | nppiConvert_32s16s_C1RSfs | 171 |
| 7.8.1.54 | nppiConvert_32s16u_C1RSfs | 171 |
| 7.8.1.55 | nppiConvert_32s32f_C1R | 172 |
| 7.8.1.56 | nppiConvert_32s32u_C1Rs | 172 |
| 7.8.1.57 | nppiConvert_32s8s_AC4R | 172 |

| | | |
|----------|---------------------------|-----|
| 7.8.1.58 | nppiConvert_32s8s_C1R | 173 |
| 7.8.1.59 | nppiConvert_32s8s_C3R | 173 |
| 7.8.1.60 | nppiConvert_32s8s_C4R | 173 |
| 7.8.1.61 | nppiConvert_32s8u_AC4R | 174 |
| 7.8.1.62 | nppiConvert_32s8u_C1R | 174 |
| 7.8.1.63 | nppiConvert_32s8u_C3R | 174 |
| 7.8.1.64 | nppiConvert_32s8u_C4R | 175 |
| 7.8.1.65 | nppiConvert_32u16s_C1RSfs | 175 |
| 7.8.1.66 | nppiConvert_32u16u_C1RSfs | 175 |
| 7.8.1.67 | nppiConvert_32u32f_C1R | 176 |
| 7.8.1.68 | nppiConvert_32u32s_C1RSfs | 176 |
| 7.8.1.69 | nppiConvert_32u8s_C1RSfs | 177 |
| 7.8.1.70 | nppiConvert_32u8u_C1RSfs | 177 |
| 7.8.1.71 | nppiConvert_8s16s_C1R | 177 |
| 7.8.1.72 | nppiConvert_8s16u_C1Rs | 178 |
| 7.8.1.73 | nppiConvert_8s32f_AC4R | 178 |
| 7.8.1.74 | nppiConvert_8s32f_C1R | 178 |
| 7.8.1.75 | nppiConvert_8s32f_C3R | 179 |
| 7.8.1.76 | nppiConvert_8s32f_C4R | 179 |
| 7.8.1.77 | nppiConvert_8s32s_AC4R | 180 |
| 7.8.1.78 | nppiConvert_8s32s_C1R | 180 |
| 7.8.1.79 | nppiConvert_8s32s_C3R | 180 |
| 7.8.1.80 | nppiConvert_8s32s_C4R | 181 |
| 7.8.1.81 | nppiConvert_8s32u_C1Rs | 181 |
| 7.8.1.82 | nppiConvert_8s8u_C1Rs | 181 |
| 7.8.1.83 | nppiConvert_8u16s_AC4R | 182 |
| 7.8.1.84 | nppiConvert_8u16s_C1R | 182 |
| 7.8.1.85 | nppiConvert_8u16s_C3R | 182 |
| 7.8.1.86 | nppiConvert_8u16s_C4R | 183 |
| 7.8.1.87 | nppiConvert_8u16u_AC4R | 183 |
| 7.8.1.88 | nppiConvert_8u16u_C1R | 183 |
| 7.8.1.89 | nppiConvert_8u16u_C3R | 184 |
| 7.8.1.90 | nppiConvert_8u16u_C4R | 184 |
| 7.8.1.91 | nppiConvert_8u32f_AC4R | 184 |
| 7.8.1.92 | nppiConvert_8u32f_C1R | 185 |
| 7.8.1.93 | nppiConvert_8u32f_C3R | 185 |

| | | |
|----------|---|-----|
| 7.8.1.94 | nppiConvert_8u32f_C4R | 185 |
| 7.8.1.95 | nppiConvert_8u32s_AC4R | 186 |
| 7.8.1.96 | nppiConvert_8u32s_C1R | 186 |
| 7.8.1.97 | nppiConvert_8u32s_C3R | 186 |
| 7.8.1.98 | nppiConvert_8u32s_C4R | 187 |
| 7.8.1.99 | nppiConvert_8u8s_C1RSfs | 187 |
| 7.9 | Scale | 188 |
| 7.9.1 | Function Documentation | 191 |
| 7.9.1.1 | nppiScale_16s8u_AC4R | 191 |
| 7.9.1.2 | nppiScale_16s8u_C1R | 191 |
| 7.9.1.3 | nppiScale_16s8u_C3R | 191 |
| 7.9.1.4 | nppiScale_16s8u_C4R | 192 |
| 7.9.1.5 | nppiScale_16u8u_AC4R | 192 |
| 7.9.1.6 | nppiScale_16u8u_C1R | 193 |
| 7.9.1.7 | nppiScale_16u8u_C3R | 193 |
| 7.9.1.8 | nppiScale_16u8u_C4R | 193 |
| 7.9.1.9 | nppiScale_32f8u_AC4R | 194 |
| 7.9.1.10 | nppiScale_32f8u_C1R | 194 |
| 7.9.1.11 | nppiScale_32f8u_C3R | 194 |
| 7.9.1.12 | nppiScale_32f8u_C4R | 195 |
| 7.9.1.13 | nppiScale_32s8u_AC4R | 195 |
| 7.9.1.14 | nppiScale_32s8u_C1R | 196 |
| 7.9.1.15 | nppiScale_32s8u_C3R | 196 |
| 7.9.1.16 | nppiScale_32s8u_C4R | 196 |
| 7.9.1.17 | nppiScale_8u16s_AC4R | 197 |
| 7.9.1.18 | nppiScale_8u16s_C1R | 197 |
| 7.9.1.19 | nppiScale_8u16s_C3R | 197 |
| 7.9.1.20 | nppiScale_8u16s_C4R | 198 |
| 7.9.1.21 | nppiScale_8u16u_AC4R | 198 |
| 7.9.1.22 | nppiScale_8u16u_C1R | 198 |
| 7.9.1.23 | nppiScale_8u16u_C3R | 199 |
| 7.9.1.24 | nppiScale_8u16u_C4R | 199 |
| 7.9.1.25 | nppiScale_8u32f_AC4R | 199 |
| 7.9.1.26 | nppiScale_8u32f_C1R | 200 |
| 7.9.1.27 | nppiScale_8u32f_C3R | 200 |
| 7.9.1.28 | nppiScale_8u32f_C4R | 200 |

| | | |
|-----------|----------------------------------|-----|
| 7.9.1.29 | nppiScale_8u32s_AC4R | 201 |
| 7.9.1.30 | nppiScale_8u32s_C1R | 201 |
| 7.9.1.31 | nppiScale_8u32s_C3R | 202 |
| 7.9.1.32 | nppiScale_8u32s_C4R | 202 |
| 7.10 | Copy Constant Border | 203 |
| 7.10.1 | Function Documentation | 205 |
| 7.10.1.1 | nppiCopyConstBorder_16s_AC4R | 205 |
| 7.10.1.2 | nppiCopyConstBorder_16s_C1R | 205 |
| 7.10.1.3 | nppiCopyConstBorder_16s_C3R | 206 |
| 7.10.1.4 | nppiCopyConstBorder_16s_C4R | 206 |
| 7.10.1.5 | nppiCopyConstBorder_16u_AC4R | 207 |
| 7.10.1.6 | nppiCopyConstBorder_16u_C1R | 207 |
| 7.10.1.7 | nppiCopyConstBorder_16u_C3R | 208 |
| 7.10.1.8 | nppiCopyConstBorder_16u_C4R | 208 |
| 7.10.1.9 | nppiCopyConstBorder_32f_AC4R | 209 |
| 7.10.1.10 | nppiCopyConstBorder_32f_C1R | 209 |
| 7.10.1.11 | nppiCopyConstBorder_32f_C3R | 210 |
| 7.10.1.12 | nppiCopyConstBorder_32f_C4R | 210 |
| 7.10.1.13 | nppiCopyConstBorder_32s_AC4R | 211 |
| 7.10.1.14 | nppiCopyConstBorder_32s_C1R | 211 |
| 7.10.1.15 | nppiCopyConstBorder_32s_C3R | 212 |
| 7.10.1.16 | nppiCopyConstBorder_32s_C4R | 212 |
| 7.10.1.17 | nppiCopyConstBorder_8u_AC4R | 213 |
| 7.10.1.18 | nppiCopyConstBorder_8u_C1R | 213 |
| 7.10.1.19 | nppiCopyConstBorder_8u_C3R | 214 |
| 7.10.1.20 | nppiCopyConstBorder_8u_C4R | 214 |
| 7.11 | Copy Replicate Border | 216 |
| 7.11.1 | Function Documentation | 218 |
| 7.11.1.1 | nppiCopyReplicateBorder_16s_AC4R | 218 |
| 7.11.1.2 | nppiCopyReplicateBorder_16s_C1R | 218 |
| 7.11.1.3 | nppiCopyReplicateBorder_16s_C3R | 219 |
| 7.11.1.4 | nppiCopyReplicateBorder_16s_C4R | 219 |
| 7.11.1.5 | nppiCopyReplicateBorder_16u_AC4R | 220 |
| 7.11.1.6 | nppiCopyReplicateBorder_16u_C1R | 220 |
| 7.11.1.7 | nppiCopyReplicateBorder_16u_C3R | 221 |
| 7.11.1.8 | nppiCopyReplicateBorder_16u_C4R | 221 |

| | | |
|-----------|----------------------------------|-----|
| 7.11.1.9 | nppiCopyReplicateBorder_32f_AC4R | 222 |
| 7.11.1.10 | nppiCopyReplicateBorder_32f_C1R | 222 |
| 7.11.1.11 | nppiCopyReplicateBorder_32f_C3R | 223 |
| 7.11.1.12 | nppiCopyReplicateBorder_32f_C4R | 223 |
| 7.11.1.13 | nppiCopyReplicateBorder_32s_AC4R | 224 |
| 7.11.1.14 | nppiCopyReplicateBorder_32s_C1R | 224 |
| 7.11.1.15 | nppiCopyReplicateBorder_32s_C3R | 225 |
| 7.11.1.16 | nppiCopyReplicateBorder_32s_C4R | 225 |
| 7.11.1.17 | nppiCopyReplicateBorder_8u_AC4R | 226 |
| 7.11.1.18 | nppiCopyReplicateBorder_8u_C1R | 226 |
| 7.11.1.19 | nppiCopyReplicateBorder_8u_C3R | 227 |
| 7.11.1.20 | nppiCopyReplicateBorder_8u_C4R | 227 |
| 7.12 | Copy Wrap Border | 228 |
| 7.12.1 | Function Documentation | 230 |
| 7.12.1.1 | nppiCopyWrapBorder_16s_AC4R | 230 |
| 7.12.1.2 | nppiCopyWrapBorder_16s_C1R | 231 |
| 7.12.1.3 | nppiCopyWrapBorder_16s_C3R | 231 |
| 7.12.1.4 | nppiCopyWrapBorder_16s_C4R | 232 |
| 7.12.1.5 | nppiCopyWrapBorder_16u_AC4R | 232 |
| 7.12.1.6 | nppiCopyWrapBorder_16u_C1R | 233 |
| 7.12.1.7 | nppiCopyWrapBorder_16u_C3R | 233 |
| 7.12.1.8 | nppiCopyWrapBorder_16u_C4R | 234 |
| 7.12.1.9 | nppiCopyWrapBorder_32f_AC4R | 234 |
| 7.12.1.10 | nppiCopyWrapBorder_32f_C1R | 235 |
| 7.12.1.11 | nppiCopyWrapBorder_32f_C3R | 235 |
| 7.12.1.12 | nppiCopyWrapBorder_32f_C4R | 236 |
| 7.12.1.13 | nppiCopyWrapBorder_32s_AC4R | 236 |
| 7.12.1.14 | nppiCopyWrapBorder_32s_C1R | 237 |
| 7.12.1.15 | nppiCopyWrapBorder_32s_C3R | 237 |
| 7.12.1.16 | nppiCopyWrapBorder_32s_C4R | 238 |
| 7.12.1.17 | nppiCopyWrapBorder_8u_AC4R | 238 |
| 7.12.1.18 | nppiCopyWrapBorder_8u_C1R | 239 |
| 7.12.1.19 | nppiCopyWrapBorder_8u_C3R | 239 |
| 7.12.1.20 | nppiCopyWrapBorder_8u_C4R | 240 |
| 7.13 | Copy Sub-Pixel | 241 |
| 7.13.1 | Function Documentation | 242 |

| | | |
|-----------|-------------------------|-----|
| 7.13.1.1 | nppiCopySubpix_16s_AC4R | 242 |
| 7.13.1.2 | nppiCopySubpix_16s_C1R | 243 |
| 7.13.1.3 | nppiCopySubpix_16s_C3R | 243 |
| 7.13.1.4 | nppiCopySubpix_16s_C4R | 244 |
| 7.13.1.5 | nppiCopySubpix_16u_AC4R | 244 |
| 7.13.1.6 | nppiCopySubpix_16u_C1R | 245 |
| 7.13.1.7 | nppiCopySubpix_16u_C3R | 245 |
| 7.13.1.8 | nppiCopySubpix_16u_C4R | 245 |
| 7.13.1.9 | nppiCopySubpix_32f_AC4R | 246 |
| 7.13.1.10 | nppiCopySubpix_32f_C1R | 246 |
| 7.13.1.11 | nppiCopySubpix_32f_C3R | 247 |
| 7.13.1.12 | nppiCopySubpix_32f_C4R | 247 |
| 7.13.1.13 | nppiCopySubpix_32s_AC4R | 247 |
| 7.13.1.14 | nppiCopySubpix_32s_C1R | 248 |
| 7.13.1.15 | nppiCopySubpix_32s_C3R | 248 |
| 7.13.1.16 | nppiCopySubpix_32s_C4R | 249 |
| 7.13.1.17 | nppiCopySubpix_8u_AC4R | 249 |
| 7.13.1.18 | nppiCopySubpix_8u_C1R | 250 |
| 7.13.1.19 | nppiCopySubpix_8u_C3R | 250 |
| 7.13.1.20 | nppiCopySubpix_8u_C4R | 250 |
| 7.14 | Duplicate Channel | 252 |
| 7.14.1 | Function Documentation | 253 |
| 7.14.1.1 | nppiDup_16s_C1AC4R | 253 |
| 7.14.1.2 | nppiDup_16s_C1C3R | 253 |
| 7.14.1.3 | nppiDup_16s_C1C4R | 254 |
| 7.14.1.4 | nppiDup_16u_C1AC4R | 254 |
| 7.14.1.5 | nppiDup_16u_C1C3R | 255 |
| 7.14.1.6 | nppiDup_16u_C1C4R | 255 |
| 7.14.1.7 | nppiDup_32f_C1AC4R | 255 |
| 7.14.1.8 | nppiDup_32f_C1C3R | 256 |
| 7.14.1.9 | nppiDup_32f_C1C4R | 256 |
| 7.14.1.10 | nppiDup_32s_C1AC4R | 256 |
| 7.14.1.11 | nppiDup_32s_C1C3R | 257 |
| 7.14.1.12 | nppiDup_32s_C1C4R | 257 |
| 7.14.1.13 | nppiDup_8u_C1AC4R | 257 |
| 7.14.1.14 | nppiDup_8u_C1C3R | 258 |

| | | |
|-----------|----------------------------|-----|
| 7.14.1.15 | nppiDup_8u_C1C4R | 258 |
| 7.15 | Transpose | 259 |
| 7.15.1 | Function Documentation | 260 |
| 7.15.1.1 | nppiTranspose_16s_C1R | 260 |
| 7.15.1.2 | nppiTranspose_16s_C3R | 260 |
| 7.15.1.3 | nppiTranspose_16s_C4R | 261 |
| 7.15.1.4 | nppiTranspose_16u_C1R | 261 |
| 7.15.1.5 | nppiTranspose_16u_C3R | 261 |
| 7.15.1.6 | nppiTranspose_16u_C4R | 262 |
| 7.15.1.7 | nppiTranspose_32f_C1R | 262 |
| 7.15.1.8 | nppiTranspose_32f_C3R | 262 |
| 7.15.1.9 | nppiTranspose_32f_C4R | 263 |
| 7.15.1.10 | nppiTranspose_32s_C1R | 263 |
| 7.15.1.11 | nppiTranspose_32s_C3R | 264 |
| 7.15.1.12 | nppiTranspose_32s_C4R | 264 |
| 7.15.1.13 | nppiTranspose_8u_C1R | 264 |
| 7.15.1.14 | nppiTranspose_8u_C3R | 265 |
| 7.15.1.15 | nppiTranspose_8u_C4R | 265 |
| 7.16 | Swap Channels | 266 |
| 7.16.1 | Function Documentation | 269 |
| 7.16.1.1 | nppiSwapChannels_16s_AC4R | 269 |
| 7.16.1.2 | nppiSwapChannels_16s_C3C4R | 269 |
| 7.16.1.3 | nppiSwapChannels_16s_C3IR | 270 |
| 7.16.1.4 | nppiSwapChannels_16s_C3R | 270 |
| 7.16.1.5 | nppiSwapChannels_16s_C4C3R | 270 |
| 7.16.1.6 | nppiSwapChannels_16s_C4IR | 271 |
| 7.16.1.7 | nppiSwapChannels_16s_C4R | 271 |
| 7.16.1.8 | nppiSwapChannels_16u_AC4R | 272 |
| 7.16.1.9 | nppiSwapChannels_16u_C3C4R | 272 |
| 7.16.1.10 | nppiSwapChannels_16u_C3IR | 273 |
| 7.16.1.11 | nppiSwapChannels_16u_C3R | 273 |
| 7.16.1.12 | nppiSwapChannels_16u_C4C3R | 273 |
| 7.16.1.13 | nppiSwapChannels_16u_C4IR | 274 |
| 7.16.1.14 | nppiSwapChannels_16u_C4R | 274 |
| 7.16.1.15 | nppiSwapChannels_32f_AC4R | 275 |
| 7.16.1.16 | nppiSwapChannels_32f_C3C4R | 275 |

| | | |
|-----------|-------------------------------------|------------|
| 7.16.1.17 | nppiSwapChannels_32f_C3IR | 276 |
| 7.16.1.18 | nppiSwapChannels_32f_C3R | 276 |
| 7.16.1.19 | nppiSwapChannels_32f_C4C3R | 276 |
| 7.16.1.20 | nppiSwapChannels_32f_C4IR | 277 |
| 7.16.1.21 | nppiSwapChannels_32f_C4R | 277 |
| 7.16.1.22 | nppiSwapChannels_32s_AC4R | 278 |
| 7.16.1.23 | nppiSwapChannels_32s_C3C4R | 278 |
| 7.16.1.24 | nppiSwapChannels_32s_C3IR | 279 |
| 7.16.1.25 | nppiSwapChannels_32s_C3R | 279 |
| 7.16.1.26 | nppiSwapChannels_32s_C4C3R | 279 |
| 7.16.1.27 | nppiSwapChannels_32s_C4IR | 280 |
| 7.16.1.28 | nppiSwapChannels_32s_C4R | 280 |
| 7.16.1.29 | nppiSwapChannels_8u_AC4R | 281 |
| 7.16.1.30 | nppiSwapChannels_8u_C3C4R | 281 |
| 7.16.1.31 | nppiSwapChannels_8u_C3IR | 282 |
| 7.16.1.32 | nppiSwapChannels_8u_C3R | 282 |
| 7.16.1.33 | nppiSwapChannels_8u_C4C3R | 282 |
| 7.16.1.34 | nppiSwapChannels_8u_C4IR | 283 |
| 7.16.1.35 | nppiSwapChannels_8u_C4R | 283 |
| 8 | Data Structure Documentation | 285 |
| 8.1 | NPP_ALIGN_16 Struct Reference | 285 |
| 8.1.1 | Detailed Description | 285 |
| 8.1.2 | Field Documentation | 285 |
| 8.1.2.1 | im | 285 |
| 8.1.2.2 | im | 286 |
| 8.1.2.3 | re | 286 |
| 8.1.2.4 | re | 286 |
| 8.2 | NPP_ALIGN_8 Struct Reference | 287 |
| 8.2.1 | Detailed Description | 287 |
| 8.2.2 | Field Documentation | 287 |
| 8.2.2.1 | im | 287 |
| 8.2.2.2 | im | 287 |
| 8.2.2.3 | im | 287 |
| 8.2.2.4 | re | 288 |
| 8.2.2.5 | re | 288 |
| 8.2.2.6 | re | 288 |

| | | |
|---------|---|-----|
| 8.3 | NppiHaarBuffer Struct Reference | 289 |
| 8.3.1 | Field Documentation | 289 |
| 8.3.1.1 | haarBuffer | 289 |
| 8.3.1.2 | haarBufferSize | 289 |
| 8.4 | NppiHaarClassifier_32f Struct Reference | 290 |
| 8.4.1 | Field Documentation | 290 |
| 8.4.1.1 | classifiers | 290 |
| 8.4.1.2 | classifierSize | 290 |
| 8.4.1.3 | classifierStep | 290 |
| 8.4.1.4 | counterDevice | 290 |
| 8.4.1.5 | numClassifiers | 290 |
| 8.5 | NppiPoint Struct Reference | 291 |
| 8.5.1 | Detailed Description | 291 |
| 8.5.2 | Field Documentation | 291 |
| 8.5.2.1 | x | 291 |
| 8.5.2.2 | y | 291 |
| 8.6 | NppiRect Struct Reference | 292 |
| 8.6.1 | Detailed Description | 292 |
| 8.6.2 | Field Documentation | 292 |
| 8.6.2.1 | height | 292 |
| 8.6.2.2 | width | 292 |
| 8.6.2.3 | x | 292 |
| 8.6.2.4 | y | 292 |
| 8.7 | NppiSize Struct Reference | 293 |
| 8.7.1 | Detailed Description | 293 |
| 8.7.2 | Field Documentation | 293 |
| 8.7.2.1 | height | 293 |
| 8.7.2.2 | width | 293 |
| 8.8 | NppLibraryVersion Struct Reference | 294 |
| 8.8.1 | Field Documentation | 294 |
| 8.8.1.1 | build | 294 |
| 8.8.1.2 | major | 294 |
| 8.8.1.3 | minor | 294 |

Chapter 1

NVIDIA Performance Primitives

Note: Starting with release 6.5, NPP is also provided as a static library (`libnppc_static.a`, `libnppi_static.a`, and `libnpps_static.a`) on Linux, Android, and Mac OSes in addition to being provided as a shared library. The static NPP libraries depend on a common thread abstraction layer library called `cuLIBOS` (`libculibos.a`) that is now distributed as part of the toolkit. Consequently, `cuLIBOS` must be provided to the linker when the static library is being linked against. The `libnppi` library is becoming quite large so to minimize library loading and CUDA runtime startup times it is recommended to use the static library(s) whenever possible. To improve loading and runtime performance when using dynamic libraries NPP 8.0 now includes the full set of `nppi` sub-libraries in addition to the full sized `nppi` library itself. Linking to only the sub-libraries that contain functions that your application uses can significantly improve load time and runtime startup performance. Some `nppi` functions make calls to other `nppi` and/or `npps` functions internally so you may need to link to a few extra libraries depending on what function calls your application makes. The `nppi` sub-libraries are split into sections corresponding to the way that `nppi` header files are split. There are also static versions of each of the new sub-libraries. The full sized `nppi` library will be deprecated in the next CUDA release. This list of sub-libraries is as follows:

```
nppial arithmetic and logical operation functions in nppi_arithmetic_and_logical_operations.h
nppicc color conversion and sampling functions in nppi_color_conversion.h
nppicom JPEG compression and decompression functions in nppi_compression_functions.h
nppidei data exchange and initialization functions in nppi_data_exchange_and_initialization.h
nppif filtering and computer vision functions in nppi_filter_functions.h
nppig geometry transformation functions found in nppi_geometry_transforms.h
nppim morphological operation functions found in nppi_morphological_operations.h
nppist statistics and linear transform in nppi_statistics_functions.h and nppi_linear_transforms.h
nppisu memory support functions in nppi_support_functions.h
nppitc threshold and compare operation functions in nppi_threshold_and_compare_operations.h
```

For example, on Linux, to compile a small application `foo` using NPP against the dynamic library, the following command can be used:

```
nvcc foo.c -lnppi -o foo
```

Whereas to compile against the static NPP library, the following command has to be used:

```
nvcc foo.c -lnppi_static -lculibos -o foo
```

It is also possible to use the native host C++ compiler. Depending on the host operating system, some additional libraries like `pthread` or `dl` might be needed on the linking line. The following command on Linux is suggested:

```
g++ foo.c -lnppi_static -lculibos -lcudart_static -lpthread -ldl
-I <cuda-toolkit-path>/include -L <cuda-toolkit-path>/lib64 -o foo
```

NPP is a stateless API, as of NPP 6.5 the ONLY state that NPP remembers between function calls is the current stream ID, i.e. the stream ID that was set in the most recent `nppSetStream` call. The default stream ID is 0. If an application intends to use NPP with multiple streams then it is the responsibility of the application to call `nppSetStream` whenever it wishes to change stream IDs. Several NPP functions may call other NPP functions internally to complete their functionality. For this reason it is recommended that `cudaDeviceSynchronize` be called before making an `nppSetStream` call to change to a new stream ID. This will insure that any internal function calls that have not yet occurred will be completed using the current stream ID before it changes to a new ID. Calling `cudaDeviceSynchronize` frequently call kill performance so minimizing the frequency of these calls is critical for good performance. It is not necessary to call `cudaDeviceSynchronize` for stream management while the same stream ID is used for multiple NPP calls. All NPP functions should be thread safe except for the following functions:

```
nppiGraphcut_32s8u - this function has been deprecated in NPP 8.0
nppiGraphcut_32f8u - this function has been deprecated in NPP 8.0
nppiGraphcut8_32s8u - this function has been deprecated in NPP 8.0
nppiGraphcut8_32f8u - this function has been deprecated in NPP 8.0
nppiDCTQuantFwd8x8LS_JPEG_8u16s_C1R
nppiDCTQuantInv8x8LS_JPEG_16s8u_C1R
```

As of NPP version 5.0 and beyond a few parameters for a few pre-5.0 existing image LUT functions have changed from host memory pointers to device memory pointers. Your application will fail (crash or report an error) if you use these functions with host memory pointers. The functions are the `nppiLUT_Linear_8u_xxx` functions.

Also, pre-5.0 function `nppiMeanStdDev8uC1RGetBufferHostSize` has been renamed `nppiMeanStdDevGetBufferHostSize_8u_C1R`.

1.1 What is NPP?

NVIDIA NPP is a library of functions for performing CUDA accelerated processing. The initial set of functionality in the library focuses on imaging and video processing and is widely applicable for developers in these areas. NPP will evolve over time to encompass more of the compute heavy tasks in a variety of problem domains. The NPP library is written to maximize flexibility, while maintaining high performance.

NPP can be used in one of two ways:

- A stand-alone library for adding GPU acceleration to an application with minimal effort. Using this route allows developers to add GPU acceleration to their applications in a matter of hours.
- A cooperative library for interoperating with a developer's GPU code efficiently.

Either route allows developers to harness the massive compute resources of NVIDIA GPUs, while simultaneously reducing development times.

1.2 Documentation

- [General API Conventions](#)

- [Signal-Processing Specific API Conventions](#)
- [Imaging-Processing Specific API Conventions](#)

1.3 Technical Specifications

Supported Platforms:

- Microsoft Windows 7, 8, and 10 (64-bit and 32-bit)
- Microsoft Windows Vista (64-bit and 32-bit)
- Linux (Centos, Ubuntu, and several others) (64-bit and 32-bit)
- Mac OS X (64-bit)
- Android on Arm (32-bit and 64-bit)

1.4 Files

NPP is comprised of the following files:

1.4.1 Header Files

- [nppdefs.h](#)
- [nppcore.h](#)
- `nppi.h`
- `npps.h`
- [nppversion.h](#)
- `npp.h`

All those header files are located in the CUDA Toolkit's

```
/include/
```

directory.

1.4.2 Library Files

Starting with Version 5.5 NPP's functionality is now split up into 3 distinct libraries:

- A core library (NPPC) containing basic functionality from the `npp.h` header files as well as functionality shared by the other two libraries.
- The image processing library NPPI. Any functions from the `nppi.h` header file (or the various header files named "nppi_XXX.h") are bundled into the NPPI library.

- The signal processing library NPPS. Any function from the npps.h header file (or the various header files named "npps_xxx.h" are bundled into the NPPS library.

On the Windows platform the NPP stub libraries are found in the CUDA Toolkit's library directory:

```
/lib/nppc.lib
```

```
/lib/nppi.lib
```

```
/lib/npps.lib
```

The matching DLLs are located in the CUDA Toolkit's binary directory. Example

```
/bin/nppi64_55_<build_no>.dll // Dynamic image-processing library for 64-bit Windows.
```

On Linux and Mac platforms the dynamic libraries are located in the lib directory

```
/lib/libnppc32.so.5.5.<build_no> // NPP 32-bit dynamic core library for Linux
```

```
/lib/libnpps32.5.5.dylib // NPP 32-bit dynamic signal processing library for Mac
```

1.5 Supported NVIDIA Hardware

NPP runs on all CUDA capable NVIDIA hardware. For details please see http://www.nvidia.com/object/cuda_learn_products.html

Chapter 2

General API Conventions

2.1 Memory Management

The design of all the NPP functions follows the same guidelines as other NVIDIA CUDA libraries like cuFFT and cuBLAS. That is that all pointer arguments in those APIs are device pointers.

This convention enables the individual developer to make smart choices about memory management that minimize the number of memory transfers. It also allows the user the maximum flexibility regarding which of the various memory transfer mechanisms offered by the CUDA runtime is used, e.g. synchronous or asynchronous memory transfers, zero-copy and pinned memory, etc.

The most basic steps involved in using NPP for processing data is as follows:

1. Transfer input data from the host to device using

```
cudaMemcpy(...)
```

2. Process data using one or several NPP functions or custom CUDA kernels

3. Transfer the result data from the device to the host using

```
cudaMemcpy(...)
```

2.1.1 Scratch Buffer and Host Pointer

Some primitives of NPP require additional device memory buffers (scratch buffers) for calculations, e.g. signal and image reductions (Sum, Max, Min, MinMax, etc.). In order to give the NPP user maximum control regarding memory allocations and performance, it is the user's responsibility to allocate and delete those temporary buffers. For one this has the benefit that the library will not allocate memory unbeknownst to the user. It also allows developers who invoke the same primitive repeatedly to allocate the scratch only once, improving performance and potential device-memory fragmentation.

Scratch-buffer memory is unstructured and may be passed to the primitive in uninitialized form. This allows for reuse of the same scratch buffers with any primitive require scratch memory, as long as it is sufficiently sized.

The minimum scratch-buffer size for a given primitive (e.g. `nppsSum_32f()`) can be obtained by a companion function (e.g. `nppsSumGetBufferSize_32f()`). The buffer size is returned via a host pointer as allocation of the scratch-buffer is performed via CUDA runtime host code.

An example to invoke signal sum primitive and allocate and free the necessary scratch memory:

```
// pSrc, pSum, pDeviceBuffer are all device pointers.
Npp32f * pSrc;
Npp32f * pSum;
Npp8u * pDeviceBuffer;
int nLength = 1024;

// Allocate the device memroy.
cudaMalloc((void **)(&pSrc), sizeof(Npp32f) * nLength);
nppsSet_32f(1.0f, pSrc, nLength);
cudaMalloc((void **)(&pSum), sizeof(Npp32f) * 1);

// Compute the appropriate size of the scratch-memory buffer
int nBufferSize;
nppsSumGetBufferSize_32f(nLength, &nBufferSize);
// Allocate the scratch buffer
cudaMalloc((void **)(&pDeviceBuffer), nBufferSize);

// Call the primitive with the scratch buffer
```

```
nppsSum_32f(pSrc, nLength, pSum, pDeviceBuffer);
Npp32f nSumHost;
cudaMemcpy(&nSumHost, pSum, sizeof(Npp32f) * 1, cudaMemcpyDeviceToHost);
printf("sum = %f\n", nSumHost); // nSumHost = 1024.0f;

// Free the device memory
cudaFree(pSrc);
cudaFree(pDeviceBuffer);
cudaFree(pSum);
```

2.2 Function Naming

Since NPP is a C API and therefore does not allow for function overloading for different data-types the NPP naming convention addresses the need to differentiate between different flavors of the same algorithm or primitive function but for various data types. This disambiguation of different flavors of a primitive is done via a suffix containing data type and other disambiguating information.

In addition to the flavor suffix, all NPP functions are prefixed with by the letters "npp". Primitives belonging to NPP's image-processing module add the letter "i" to the npp prefix, i.e. are prefixed by "nppi". Similarly signal-processing primitives are prefixed with "npps".

The general naming scheme is:

```
npp<module info><PrimitiveName>_<data-type info>[_<additional flavor info>](<parameter list>)
```

The data-type information uses the same names as the [Basic NPP Data Types](#). For example the data-type information "8u" would imply that the primitive operates on [Npp8u](#) data.

If a primitive consumes different type data from what it produces, both types will be listed in the order of consumed to produced data type.

Details about the "additional flavor information" is provided for each of the NPP modules, since each problem domain uses different flavor information suffixes.

2.3 Integer Result Scaling

NPP signal processing and imaging primitives often operate on integer data. This integer data is usually a fixed point fractional representation of some physical magnitue (e.g. luminance). Because of this fixed-point nature of the representation many numerical operations (e.g. addition or multiplication) tend to produce results exceeding the original fixed-point range if treated as regular integers.

In cases where the results exceed the original range, these functions clamp the result values back to the valid range. E.g. the maximum positive value for a 16-bit unsigned integer is 32767. A multiplication operation of $4 * 10000 = 40000$ would exceed this range. The result would be clamped to be 32767.

To avoid the level of lost information due to clamping most integer primitives allow for result scaling. Primitives with result scaling have the "Sfs" suffix in their name and provide a parameter "nScaleFactor" that controls the amount of scaling. Before the results of an operation are clamped to the valid output-data range by multiplying them with $2^{-nScaleFactor}$.

Example: The primitive `nppsSqr_8u_Sfs()` computes the square of 8-bit unsigned sample values in a signal (1D array of values). The maximum value of a 8-bit value is 255. The square of $255^2 = 65025$ which would be clamped to 255 if no result scaling is performed. In order to map the maximum value of 255 to 255 in the result, one would specify an integer result scaling factor of 8, i.e. multiply each result with $2^{-8} = \frac{1}{2^8} = \frac{1}{256}$. The final result for a signal value of 255 being squared and scaled would be:

$$255^2 \cdot 2^{-8} = 254.00390625$$

which would be rounded to a final result of 254.

A medium gray value of 128 would result in

$$128^2 * 2^{-8} = 64$$

2.4 Rounding Modes

Many NPP functions require converting floating-point values to integers. The [NppRoundMode](#) enum lists NPP's supported rounding modes. Not all primitives in NPP that perform rounding as part of their functionality allow the user to specify the round-mode used. Instead they use NPP's default rounding mode, which is [NPP_RND_FINANCIAL](#).

2.4.1 Rounding Mode Parameter

A subset of NPP functions performing rounding as part of their functionality do allow the user to specify which rounding mode is used through a parameter of the [NppRoundMode](#) type.

Chapter 3

Signal-Processing Specific API Conventions

3.1 Signal Data

Signal data is passed to and from NPPS primitives via a pointer to the signal's data type.

The general idea behind this fairly low-level way of passing signal data is ease-of-adoption into existing software projects:

- Passing the data pointer rather than a higher-level signal struct allows for easy adoption by not requiring a specific signal representation (that could include total signal size offset, or other additional information). This avoids awkward packing and unpacking of signal data from the host application to an NPP specific signal representation.

3.1.1 Parameter Names for Signal Data

There are three general cases of image-data passing throughout NPP detailed in the following sections.

Those are signals consumed by the algorithm.

3.1.1.1 Source Signal Pointer

The source signal data is generally passed via a pointer named

```
pSrc
```

The source signal pointer is generally defined constant, enforcing that the primitive does not change any image data pointed to by that pointer. E.g.

```
nppsPrimitive_32s(const Npp32s * pSrc, ...)
```

In case the primitive consumes multiple signals as inputs the source pointers are numbered like this:

```
pSrc1, pSrc2, ...
```

3.1.1.2 Destination Signal Pointer

The destination signal data is generally passed via a pointer named

```
pDst
```

In case the primitive consumes multiple signals as inputs the source pointers are numbered like this:

```
pDst1, pDst2, ...
```

3.1.1.3 In-Place Signal Pointer

In the case of in-place processing, source and destination are served by the same pointer and thus pointers to in-place signal data are called:

```
pSrcDst
```


3.1.2 Signal Data Alignment Requirements

NPP requires signal sample data to be naturally aligned, i.e. any pointer

```
NppType * p;
```

to a sample in a signal needs to fulfill:

```
assert(p % sizeof(p) == 0);
```

3.1.3 Signal Data Related Error Codes

All NPPI primitives operating on signal data validate the signal-data pointer for proper alignment and test that the point is not null.

Failed validation results in one of the following error codes being returned and the primitive not being executed:

- [NPP_NULL_POINTER_ERROR](#) is returned if the image-data pointer is 0 (NULL).
- [NPP_ALIGNMENT_ERROR](#) if the signal-data pointer address is not a multiple of the signal's data-type size.

3.2 Signal Length

The vast majority of NPPS functions take a

```
nLength
```

parameter that tells the primitive how many of the signal's samples starting from the given data pointer are to be processed.

3.2.1 Length Related Error Codes

All NPPS primitives taking a length parameter validate this input.

Failed validation results in the following error code being returned and the primitive not being executed:

- [NPP_SIZE_ERROR](#) is returned if the length is negative.

Chapter 4

Imaging-Processing Specific API Conventions

4.1 Function Naming

Image processing related functions use a number of suffixes to indicate various different flavors of a primitive beyond just different data types. The flavor suffix uses the following abbreviations:

- "A" if the image is a 4 channel image this indicates the result alpha channel is not affected by the primitive.
- "Cn" the image consists of n channel packed pixels, where n can be 1, 2, 3 or 4.
- "Pn" the image consists of n separate image planes, where n can be 1, 2, 3 or 4.
- "C" (following the channel information) indicates that the primitive only operates on one of the color channels, the "channel-of-interest". All other output channels are not affected by the primitive.
- "I" indicates that the primitive works "in-place". In this case the image-data pointer is usually named "pSrcDst" to indicate that the image data serves as source and destination at the same time.
- "M" indicates "masked operation". These types of primitives have an additional "mask image" as input. Each pixel in the destination image corresponds to a pixel in the mask image. Only pixels with a corresponding non-zero mask pixel are being processed.
- "R" indicates the primitive operates only on a rectangular "region-of-interest" or "ROI". All ROI primitives take an additional input parameter of type [NppiSize](#), which specifies the width and height of the rectangular region that the primitive should process. For details on how primitives operate on ROIs see: [Region-of-Interest \(ROI\)](#).
- "Sfs" indicates the result values are processed by fixed scaling and saturation before they're written out.

The suffixes above always appear in alphabetical order. E.g. a 4 channel primitive not affecting the alpha channel with masked operation, in place and with scaling/saturation and ROI would have the postfix: "AC4IMRSfs".

4.2 Image Data

Image data is passed to and from NPPI primitives via a pair of parameters:

1. A pointer to the image's underlying data type.
2. A line step in bytes (also sometimes called line stride).

The general idea behind this fairly low-level way of passing image data is ease-of-adoption into existing software projects:

- Passing a raw pointer to the underlying pixel data type, rather than structured (by color) channel pixel data allows usage of the function in a wide variety of situations avoiding risky type cast or expensive image data copies.
- Passing the data pointer and line step individually rather than a higher- level image struct again allows for easy adoption by not requiring a specific image representation and thus avoiding awkward packing and unpacking of image data from the host application to an NPP specific image representation.

4.2.1 Line Step

The line step (also called "line stride" or "row step") allows lines of oddly sized images to start on well-aligned addresses by adding a number of unused bytes at the ends of the lines. This type of line padding has been common practice in digital image processing for a long time and is not particular to GPU image processing.

The line step is the number of bytes in a line **including the padding**. An other way to interpret this number is to say that it is the number of bytes between the first pixel of successive rows in the image, or generally the number of bytes between two neighboring pixels in any column of pixels.

The general reason for the existence of the line step it is that uniformly aligned rows of pixel enable optimizations of memory-access patterns.

Even though all functions in NPP will work with arbitrarily aligned images, best performance can only be achieved with well aligned image data. Any image data allocated with the NPP image allocators or the 2D memory allocators in the CUDA runtime, is well aligned.

Particularly on older CUDA capable GPUs it is likely that the performance decrease for misaligned data is substantial (orders of magnitude).

All image data passed to NPPI primitives requires a line step to be provided. It is important to keep in mind that this line step is always specified in terms of bytes, not pixels.

4.2.2 Parameter Names for Image Data

There are three general cases of image-data passing throughout NPP detailed in the following sections.

4.2.2.1 Passing Source-Image Data

Those are images consumed by the algorithm.

4.2.2.1.1 Source-Image Pointer

The source image data is generally passed via a pointer named

```
pSrc
```

The source image pointer is generally defined constant, enforcing that the primitive does not change any image data pointed to by that pointer. E.g.

```
nppiPrimitive_32s_C1R(const Npp32s * pSrc, ...)
```

In case the primitive consumes multiple images as inputs the source pointers are numbered like this:

```
pSrc1, pSrc2, ...
```

4.2.2.1.2 Source-Planar-Image Pointer Array

The planar source image data is generally passed via an array of pointers named

```
pSrc[]
```

The planar source image pointer array is generally defined a constant array of constant pointers, enforcing that the primitive does not change any image data pointed to by those pointers. E.g.

```
nppiPrimitive_8u_P3R(const Npp8u * const pSrc[3], ...)
```

Each pointer in the array points to a different image plane.

4.2.2.1.3 Source-Planar-Image Pointer

The multiple plane source image data is passed via a set of pointers named

```
pSrc1, pSrc2, ...
```

The planar source image pointer is generally defined as one of a set of constant pointers with each pointer pointing to a different input image plane.

4.2.2.1.4 Source-Image Line Step

The source image line step is the number of bytes between successive rows in the image. The source image line step parameter is

```
nSrcStep
```

or in the case of multiple source images

```
nSrcStep1, nSrcStep2, ...
```

4.2.2.1.5 Source-Planar-Image Line Step Array

The source planar image line step array is an array where each element of the array contains the number of bytes between successive rows for a particular plane in the input image. The source planar image line step array parameter is

```
rSrcStep[]
```

4.2.2.1.6 Source-Planar-Image Line Step

The source planar image line step is the number of bytes between successive rows in a particular plane of the multiplane input image. The source planar image line step parameter is

```
nSrcStep1, nSrcStep2, ...
```

4.2.2.2 Passing Destination-Image Data

Those are images produced by the algorithm.

4.2.2.2.1 Destination-Image Pointer

The destination image data is generally passed via a pointer named

```
pDst
```

In case the primitive generates multiple images as outputs the destination pointers are numbered like this:

```
pDst1, pDst2, ...
```

4.2.2.2.2 Destination-Planar-Image Pointer Array

The planar destination image data pointers are generally passed via an array of pointers named

```
pDst[]
```

Each pointer in the array points to a different image plane.

4.2.2.2.3 Destination-Planar-Image Pointer

The destination planar image data is generally passed via a pointer to each plane of a multiplane output image named

```
pDst1, pDst2, ...
```

4.2.2.2.4 Destination-Image Line Step

The destination image line step parameter is

```
nDstStep
```

or in the case of multiple destination images

```
nDstStep1, nDstStep2, ...
```

4.2.2.2.5 Destination-Planar-Image Line Step Array

The destination planar image line step array is an array where each element of the array contains the number of bytes between successive rows for a particular plane in the output image. The destination planar image line step array parameter is

```
rDstStep[]
```

4.2.2.2.6 Destination-Planar-Image Line Step

The destination planar image line step is the number of bytes between successive rows for a particular plane in a multiplane output image. The destination planar image line step parameter is

```
nDstStep1, nDstStep2, ...
```

4.2.2.3 Passing In-Place Image Data

4.2.2.3.1 In-Place Image Pointer

In the case of in-place processing, source and destination are served by the same pointer and thus pointers to in-place image data are called:

```
pSrcDst
```

4.2.2.3.2 In-Place-Image Line Step

The in-place line step parameter is

```
nSrcDstStep
```

4.2.2.4 Passing Mask-Image Data

Some image processing primitives have variants supporting [Masked Operation](#).

4.2.2.4.1 Mask-Image Pointer

The mask-image data is generally passed via a pointer named

```
pMask
```

4.2.2.4.2 Mask-Image Line Step

The mask-image line step parameter is

```
nMaskStep
```

4.2.2.5 Passing Channel-of-Interest Data

Some image processing primitives support [Channel-of-Interest API](#).

4.2.2.5.1 Channel_of_Interest Number

The channel-of-interest data is generally an integer (either 1, 2, or 3):

```
nCOI
```

4.2.3 Image Data Alignment Requirements

NPP requires pixel data to adhere to certain alignment constraints: For 2 and 4 channel images the following alignment requirement holds: `data_pointer % (#channels * sizeof(channel type)) == 0`. E.g. a 4 channel image with underlying type [Npp8u](#) (8-bit unsigned) would require all pixels to fall on addresses that are multiples of 4 (4 channels * 1 byte size).

As a logical consequence of all pixels being aligned to their natural size the image line steps of 2 and 4 channel images also need to be multiples of the pixel size.

1 and 3 channel images only require that pixel pointers are aligned to the underlying data type, i.e. `pData % sizeof(data type) == 0`. And consequentially line steps are also held to this requirement.

4.2.4 Image Data Related Error Codes

All NPPI primitives operating on image data validate the image-data pointer for proper alignment and test that the point is not null. They also validate the line stride for proper alignment and guard against the step being less or equal to 0. Failed validation results in one of the following error codes being returned and the primitive not being executed:

- `NPP_STEP_ERROR` is returned if the data step is 0 or negative.
- `NPP_NOT_EVEN_STEP_ERROR` is returned if the line step is not a multiple of the pixel size for 2 and 4 channel images.
- `NPP_NULL_POINTER_ERROR` is returned if the image-data pointer is 0 (NULL).
- `NPP_ALIGNMENT_ERROR` if the image-data pointer address is not a multiple of the pixel size for 2 and 4 channel images.

4.3 Region-of-Interest (ROI)

In practice processing a rectangular sub-region of an image is often more common than processing complete images. The vast majority of NPPI's image-processing primitives allow for processing of such sub regions also referred to as regions-of-interest or ROIs.

All primitives supporting ROI processing are marked by a "R" in their name suffix. In most cases the ROI is passed as a single `NppiSize` struct, which provides the width and height of the ROI. This raises the question how the primitive knows where in the image this rectangle of (width, height) is located. The "start pixel" of the ROI is implicitly given by the image-data pointer. I.e. instead of explicitly passing a pixel coordinate for the upper-left corner (lowest memory address), the user simply offsets the image-data pointers to point to the first pixel of the ROI.

In practice this means that for an image (`pSrc`, `nSrcStep`) and the start-pixel of the ROI being at location (`x`, `y`), one would pass

```
pSrcOffset = pSrc + y * nSrcStep + x * PixelSize;
```

as the image-data source to the primitive. `PixelSize` is typically computed as

```
PixelSize = NumberOfColorChannels * sizeof(PixelDataType).
```

E.g. for a primitive like `nppiSet_16s_C4R()` we would have

- `NumberOfColorChannels == 4;`
- `sizeof(Npp16s) == 2;`
- and thus `PixelSize = 4 * 2 = 8;`

4.3.1 ROI Related Error Codes

All NPPI primitives operating on ROIs of image data validate the ROI size and image's step size. Failed validation results in one of the following error codes being returned and the primitive not being executed:

- `NPP_SIZE_ERROR` is returned if either the ROI width or ROI height are negative.
- `NPP_STEP_ERROR` is returned if the ROI width exceeds the image's line step. In mathematical terms $(\text{widthROI} * \text{PixelSize}) > \text{nLinStep}$ indicates an error.

4.4 Masked Operation

Some primitive support masked operation. An "M" in the suffix of those variants indicates masked operation. Primitives supporting masked operation consume an additional input image provided via a [Mask-Image Pointer](#) and [Mask-Image Line Step](#). The mask image is interpreted by these primitives as a boolean image. The values of type `Npp8u` are interpreted as boolean values where a values of 0 indicates false, any non-zero values true.

Unless otherwise indicated the operation is only performed on pixels where its spatially corresponding mask pixel is true (non-zero). E.g. a masked copy operation would only copy those pixels in the ROI that have corresponding non-zero mask pixels.

4.5 Channel-of-Interest API

Some primitives allow restricting operations to a single channel of interest within a multi-channel image. These primitives are suffixed with the letter "C" (after the channel information, e.g. `nppiCopy_8u_C3CR(...)`). The channel-of-interest is generally selected by offsetting the image-data pointer to point directly to the channel- of-interest rather than the base of the first pixel in the ROI. Some primitives also explicitly specify the selected channel number and pass it via an integer, e.g. `nppiMean_StdDev_8u_C3CR(...)`.

4.5.1 Select-Channel Source-Image Pointer

This is a pointer to the channel-of-interest within the first pixel of the source image. E.g. if `pSrc` is the pointer to the first pixel inside the ROI of a three channel image. Using the appropriate select-channel copy primitive one could copy the second channel of this source image into the first channel of a destination image given by `pDst` by offsetting the pointer by one:

```
nppiCopy_8u_C3CR(pSrc + 1, nSrcStep, pDst, nDstStep, oSizeROI);
```

4.5.2 Select-Channel Source-Image

Some primitives allow the user to select the channel-of-interest by specifying the channel number (`nCOI`). This approach is typically used in the image statistical functions. For example,

```
nppiMean_StdDev_8u_C3CR(pSrc, nSrcStep, oSizeROI, nCOI, pDeviceBuffer, pMean, pStdDev );
```

The channel-of-interest number can be either 1, 2, or 3.

4.5.3 Select-Channel Destination-Image Pointer

This is a pointer to the channel-of-interest within the first pixel of the destination image. E.g. if `pDst` is the pointer to the first pixel inside the ROI of a three channel image. Using the appropriate select-channel

copy primitive one could copy data into the second channel of this destination image from the first channel of a source image given by pSrc by offsetting the destination pointer by one:

```
nppiCopy_8u_C3CR(pSrc, nSrcStep, pDst + 1, nDstStep, oSizeROI);
```

4.6 Source-Image Sampling

A large number of NPP image-processing functions consume at least one source image and produce an output image (e.g. `nppiAddC_8u_C1RSfs()` or `nppiFilterBox_8u_C1R()`). All NPP functions falling into this category also operate on ROIs (see [Region-of-Interest \(ROI\)](#)) which for these functions should be considered to describe the destination ROI. In other words the ROI describes a rectangular region in the destination image and all pixels inside of this region are being written by the function in question.

In order to use such functions successfully it is important to understand how the user defined destination ROI affects which pixels in the input image(s) are being read by the algorithms. To simplify the discussion of ROI propagation (i.e. given a destination ROI, what are the ROIs in the source(s)), it makes sense to distinguish two major cases:

1. Point-Wise Operations: These are primitives like `nppiAddC_8u_C1RSfs()`. Each output pixel requires exactly one input pixel to be read.
2. Neighborhood Operations: These are primitives like `nppiFilterBox_8u_C1R()`, which require a group of pixels from the source image(s) to be read in order to produce a single output.

4.6.1 Point-Wise Operations

As mentioned above, point-wise operations consume a single pixel from the input image (or a single pixel from each input image, if the operation in question has more than one input image) in order to produce a single output pixel.

4.6.2 Neighborhood Operations

In the case of neighborhood operations a number of input pixels (a "neighborhood" of pixels) is read in the input image (or images) in order to compute a single output pixel. All of the functions for `image_filtering_functions` and `image_morphological_operations` are neighborhood operations.

Most of these functions have parameters that affect the size and relative location of the neighborhood: a mask-size structure and an anchor-point structure. Both parameters are described in more detail in the next subsections.

4.6.2.1 Mask-Size Parameter

Many NPP neighborhood operations allow the user to specify the size of the neighborhood via a parameter usually named `oMaskSize` of type `NppiSize`. In those cases the neighborhood of pixels read from the source(s) is exactly the size of the mask. Assuming the mask is anchored at location (0, 0) (see [Anchor-Point Parameter](#) below) and has a size of (w, h), i.e.

```
assert(oMaskSize.w == w);  
assert(oMaskSize.h == h);  
assert(oAnchor.x == 0);  
assert(oAnchor.y == 0);
```

a neighborhood operation would read the following source pixels in order to compute destination pixel $D_{i,j}$:

$$\begin{array}{cccc} S_{i,j} & S_{i,j+1} & \cdots & S_{i,j+w-1} \\ S_{i+1,j} & S_{i+1,j+1} & \cdots & S_{i+1,j+w-1} \\ \vdots & \vdots & \ddots & \vdots \\ S_{i+h-1,j} & S_{i+h-1,j+1} & \cdots & S_{i+h-1,j+w-1} \end{array}$$

4.6.2.2 Anchor-Point Parameter

Many NPP primitives performing neighborhood operations allow the user to specify the relative location of the neighborhood via a parameter usually named `oAnchor` of type [NppiPoint](#). Using the anchor a developer can choose the position of the mask (see [Mask-Size Parameter](#)) relative to current pixel index.

Using the same example as in [Mask-Size Parameter](#), but this time with an anchor position of (a, b):

```
assert(oMaskSize.w == w);
assert(oMaskSize.h == h);
assert(oAnchor.x == a);
assert(oAnchor.y == b);
```

the following pixels from the source image would be read:

$$\begin{array}{cccc} S_{i-a,j-b} & S_{i-a,j-b+1} & \cdots & S_{i-a,j-b+w-1} \\ S_{i-a+1,j-b} & S_{i-a+1,j-b+1} & \cdots & S_{i-a+1,j-b+w-1} \\ \vdots & \vdots & \ddots & \vdots \\ S_{i-a+h-1,j-b} & S_{i-a+h-1,j-b+1} & \cdots & S_{i-a+h-1,j-b+w-1} \end{array}$$

4.6.2.3 Sampling Beyond Image Boundaries

NPP primitives in general and NPP neighborhood operations in particular require that all pixel locations read and written are valid and within the boundaries of the respective images. Sampling outside of the defined image data regions results in undefined behavior and may lead to system instability.

This poses a problem in practice: when processing full-size images one cannot choose the destination ROI to be the same size as the source image. Because neighborhood operations read pixels from an enlarged source ROI, the destination ROI must be shrunk so that the expanded source ROI does not exceed the source image's size.

For cases where this "shrinking" of the destination image size is unacceptable, NPP provides a set of border-expanding Copy primitives. E.g. [nppiCopyConstBorder_8u_C1R\(\)](#), [nppiCopyReplicateBorder_8u_C1R\(\)](#) and [nppiCopyWrapBorder_8u_C1R\(\)](#). The user can use these primitives to "expand" the source image's size using one of the three expansion modes. The expanded image can then be safely passed to a neighborhood operation producing a full-size result.

Chapter 5

Module Index

5.1 Modules

Here is a list of all modules:

| | |
|--|-----|
| NPP Core | 27 |
| NPP Type Definitions and Constants | 31 |
| Basic NPP Data Types | 46 |
| Memory Management | 50 |
| Data Exchange and Initialization | 62 |
| Set | 63 |
| Copy | 97 |
| Convert | 144 |
| Scale | 188 |
| Copy Constant Border | 203 |
| Copy Replicate Border | 216 |
| Copy Wrap Border | 228 |
| Copy Sub-Pixel | 241 |
| Duplicate Channel | 252 |
| Transpose | 259 |
| Swap Channels | 266 |

Chapter 6

Data Structure Index

6.1 Data Structures

Here are the data structures with brief descriptions:

| | |
|---|-----|
| NPP_ALIGN_16 (Complex Number This struct represents a long long complex number) | 285 |
| NPP_ALIGN_8 (Complex Number This struct represents an unsigned int complex number) . . | 287 |
| NppiHaarBuffer | 289 |
| NppiHaarClassifier_32f | 290 |
| NppiPoint (2D Point) | 291 |
| NppiRect (2D Rectangle This struct contains position and size information of a rectangle in two space) | 292 |
| NppiSize (2D Size This struct typically represents the size of a a rectangular region in two space) | 293 |
| NppLibraryVersion | 294 |

Chapter 7

Module Documentation

7.1 NPP Core

Basic functions for library management, in particular library version and device property query functions.

Functions

- const [NppLibraryVersion](#) * [nppGetLibVersion](#) (void)
Get the NPP library version.
- [NppGpuComputeCapability](#) [nppGetGpuComputeCapability](#) (void)
What CUDA compute model is supported by the active CUDA device?
- int [nppGetGpuNumSMs](#) (void)
Get the number of Streaming Multiprocessors (SM) on the active CUDA device.
- int [nppGetMaxThreadsPerBlock](#) (void)
Get the maximum number of threads per block on the active CUDA device.
- int [nppGetMaxThreadsPerSM](#) (void)
Get the maximum number of threads per SM for the active GPU.
- int [nppGetGpuDeviceProperties](#) (int *pMaxThreadsPerSM, int *pMaxThreadsPerBlock, int *pNumberOfSMs)
Get the maximum number of threads per SM, maximum threads per block, and number of SMs for the active GPU.
- const char * [nppGetGpuName](#) (void)
Get the name of the active CUDA device.
- [cudaStream_t](#) [nppGetStream](#) (void)
Get the NPP CUDA stream.
- unsigned int [nppGetStreamNumSMs](#) (void)
Get the number of SMs on the device associated with the current NPP CUDA stream.

- unsigned int `nppGetStreamMaxThreadsPerSM` (void)
Get the maximum number of threads per SM on the device associated with the current NPP CUDA stream.
- void `nppSetStream` (cudaStream_t hStream)
Set the NPP CUDA stream.

7.1.1 Detailed Description

Basic functions for library management, in particular library version and device property query functions.

7.1.2 Function Documentation

7.1.2.1 NppGpuComputeCapability nppGetGpuComputeCapability (void)

What CUDA compute model is supported by the active CUDA device?

Before trying to call any NPP functions, the user should make a call this function to ensure that the current machine has a CUDA capable device.

Returns:

An enum value representing if a CUDA capable device was found and what level of compute capabilities it supports.

7.1.2.2 int nppGetGpuDeviceProperties (int * pMaxThreadsPerSM, int * pMaxThreadsPerBlock, int * pNumberOfSMs)

Get the maximum number of threads per SM, maximum threads per block, and number of SMs for the active GPU.

Returns:

cudaSuccess for success, -1 for failure

7.1.2.3 const char* nppGetGpuName (void)

Get the name of the active CUDA device.

Returns:

Name string of the active graphics-card/compute device in a system.

7.1.2.4 int nppGetGpuNumSMs (void)

Get the number of Streaming Multiprocessors (SM) on the active CUDA device.

Returns:

Number of SMs of the default CUDA device.

7.1.2.5 const NppLibraryVersion* nppGetLibVersion (void)

Get the NPP library version.

Returns:

A struct containing separate values for major and minor revision and build number.

7.1.2.6 int nppGetMaxThreadsPerBlock (void)

Get the maximum number of threads per block on the active CUDA device.

Returns:

Maximum number of threads per block on the active CUDA device.

7.1.2.7 int nppGetMaxThreadsPerSM (void)

Get the maximum number of threads per SM for the active GPU.

Returns:

Maximum number of threads per SM for the active GPU

7.1.2.8 cudaStream_t nppGetStream (void)

Get the NPP CUDA stream.

NPP enables concurrent device tasks via a global stream state variable. The NPP stream by default is set to stream 0, i.e. non-concurrent mode. A user can set the NPP stream to any valid CUDA stream. All CUDA commands issued by NPP (e.g. kernels launched by the NPP library) are then issued to that NPP stream.

7.1.2.9 unsigned int nppGetStreamMaxThreadsPerSM (void)

Get the maximum number of threads per SM on the device associated with the current NPP CUDA stream.

NPP enables concurrent device tasks via a global stream state variable. The NPP stream by default is set to stream 0, i.e. non-concurrent mode. A user can set the NPP stream to any valid CUDA stream. All CUDA commands issued by NPP (e.g. kernels launched by the NPP library) are then issued to that NPP stream. This call avoids a cudaGetDeviceProperties() call.

7.1.2.10 unsigned int nppGetStreamNumSMs (void)

Get the number of SMs on the device associated with the current NPP CUDA stream.

NPP enables concurrent device tasks via a global stream state variable. The NPP stream by default is set to stream 0, i.e. non-concurrent mode. A user can set the NPP stream to any valid CUDA stream. All CUDA commands issued by NPP (e.g. kernels launched by the NPP library) are then issued to that NPP stream. This call avoids a cudaGetDeviceProperties() call.

7.1.2.11 void nppSetStream (cudaStream_t *hStream*)

Set the NPP CUDA stream.

See also:

[nppGetStream\(\)](#)

7.2 NPP Type Definitions and Constants

Data Structures

- struct [NppLibraryVersion](#)
- struct [NppiPoint](#)
2D Point
- struct [NppiSize](#)
2D Size This struct typically represents the size of a rectangular region in two space.
- struct [NppiRect](#)
2D Rectangle This struct contains position and size information of a rectangle in two space.
- struct [NppiHaarClassifier_32f](#)
- struct [NppiHaarBuffer](#)

Modules

- [Basic NPP Data Types](#)

Defines

- #define [NPP_MIN_8U](#) (0)
Minimum 8-bit unsigned integer.
- #define [NPP_MAX_8U](#) (255)
Maximum 8-bit unsigned integer.
- #define [NPP_MIN_16U](#) (0)
Minimum 16-bit unsigned integer.
- #define [NPP_MAX_16U](#) (65535)
Maximum 16-bit unsigned integer.
- #define [NPP_MIN_32U](#) (0)
Minimum 32-bit unsigned integer.
- #define [NPP_MAX_32U](#) (4294967295U)
Maximum 32-bit unsigned integer.
- #define [NPP_MIN_64U](#) (0)
Minimum 64-bit unsigned integer.
- #define [NPP_MAX_64U](#) (18446744073709551615ULL)
Maximum 64-bit unsigned integer.
- #define [NPP_MIN_8S](#) (-127 - 1)
Minimum 8-bit signed integer.

- #define `NPP_MAX_8S` (127)
Maximum 8-bit signed integer.
- #define `NPP_MIN_16S` (-32767 - 1)
Minimum 16-bit signed integer.
- #define `NPP_MAX_16S` (32767)
Maximum 16-bit signed integer.
- #define `NPP_MIN_32S` (-2147483647 - 1)
Minimum 32-bit signed integer.
- #define `NPP_MAX_32S` (2147483647)
Maximum 32-bit signed integer.
- #define `NPP_MAX_64S` (9223372036854775807LL)
Maximum 64-bit signed integer.
- #define `NPP_MIN_64S` (-9223372036854775807LL - 1)
Minimum 64-bit signed integer.
- #define `NPP_MINABS_32F` (1.175494351e-38f)
Smallest positive 32-bit floating point value.
- #define `NPP_MAXABS_32F` (3.402823466e+38f)
Largest positive 32-bit floating point value.
- #define `NPP_MINABS_64F` (2.2250738585072014e-308)
Smallest positive 64-bit floating point value.
- #define `NPP_MAXABS_64F` (1.7976931348623158e+308)
Largest positive 64-bit floating point value.

Enumerations

- enum `NppiInterpolationMode` {
`NPPI_INTER_UNDEFINED` = 0,
`NPPI_INTER_NN` = 1,
`NPPI_INTER_LINEAR` = 2,
`NPPI_INTER_CUBIC` = 4,
`NPPI_INTER_CUBIC2P_BSPLINE`,
`NPPI_INTER_CUBIC2P_CATMULLROM`,
`NPPI_INTER_CUBIC2P_B05C03`,
`NPPI_INTER_SUPER` = 8,
`NPPI_INTER_LANCZOS` = 16,
`NPPI_INTER_LANCZOS3_ADVANCED` = 17,
`NPPI_SMOOTH_EDGE` = (1 << 31) }

Filtering methods.

- enum `NppiBayerGridPosition` {
 `NPPI_BAYER_BGGR` = 0,
 `NPPI_BAYER_RGGB` = 1,
 `NPPI_BAYER_GBRG` = 2,
 `NPPI_BAYER_GRBG` = 3 }

Bayer Grid Position Registration.

- enum `NppiMaskSize` {
 `NPP_MASK_SIZE_1_X_3`,
 `NPP_MASK_SIZE_1_X_5`,
 `NPP_MASK_SIZE_3_X_1` = 100,
 `NPP_MASK_SIZE_5_X_1`,
 `NPP_MASK_SIZE_3_X_3` = 200,
 `NPP_MASK_SIZE_5_X_5`,
 `NPP_MASK_SIZE_7_X_7` = 400,
 `NPP_MASK_SIZE_9_X_9` = 500,
 `NPP_MASK_SIZE_11_X_11` = 600,
 `NPP_MASK_SIZE_13_X_13` = 700,
 `NPP_MASK_SIZE_15_X_15` = 800 }

Fixed filter-kernel sizes.

- enum `NppiDifferentialKernel` {
 `NPP_FILTER_SOBEL`,
 `NPP_FILTER_SCHARR` }

Differential Filter types.

- enum `NppStatus` {
 `NPP_NOT_SUPPORTED_MODE_ERROR` = -9999,
 `NPP_INVALID_HOST_POINTER_ERROR` = -1032,
 `NPP_INVALID_DEVICE_POINTER_ERROR` = -1031,
 `NPP_LUT_PALETTE_BITSIZE_ERROR` = -1030,
 `NPP_ZC_MODE_NOT_SUPPORTED_ERROR` = -1028,
 `NPP_NOT_SUFFICIENT_COMPUTE_CAPABILITY` = -1027,
 `NPP_TEXTURE_BIND_ERROR` = -1024,
 `NPP_WRONG_INTERSECTION_ROI_ERROR` = -1020,
 `NPP_HAAR_CLASSIFIER_PIXEL_MATCH_ERROR` = -1006,
 `NPP_MEMFREE_ERROR` = -1005,
 `NPP_MEMSET_ERROR` = -1004,
 `NPP_MEMCPY_ERROR` = -1003,
 `NPP_ALIGNMENT_ERROR` = -1002,
 `NPP_CUDA_KERNEL_EXECUTION_ERROR` = -1000,

NPP_ROUND_MODE_NOT_SUPPORTED_ERROR = -213,
NPP_QUALITY_INDEX_ERROR = -210,
NPP_RESIZE_NO_OPERATION_ERROR = -201,
NPP_OVERFLOW_ERROR = -109,
NPP_NOT_EVEN_STEP_ERROR = -108,
NPP_HISTOGRAM_NUMBER_OF_LEVELS_ERROR = -107,
NPP_LUT_NUMBER_OF_LEVELS_ERROR = -106,
NPP_CORRUPTED_DATA_ERROR = -61,
NPP_CHANNEL_ORDER_ERROR = -60,
NPP_ZERO_MASK_VALUE_ERROR = -59,
NPP_QUADRANGLE_ERROR = -58,
NPP_RECTANGLE_ERROR = -57,
NPP_COEFFICIENT_ERROR = -56,
NPP_NUMBER_OF_CHANNELS_ERROR = -53,
NPP_COI_ERROR = -52,
NPP_DIVISOR_ERROR = -51,
NPP_CHANNEL_ERROR = -47,
NPP_STRIDE_ERROR = -37,
NPP_ANCHOR_ERROR = -34,
NPP_MASK_SIZE_ERROR = -33,
NPP_RESIZE_FACTOR_ERROR = -23,
NPP_INTERPOLATION_ERROR = -22,
NPP_MIRROR_FLIP_ERROR = -21,
NPP_MOMENT_00_ZERO_ERROR = -20,
NPP_THRESHOLD_NEGATIVE_LEVEL_ERROR = -19,
NPP_THRESHOLD_ERROR = -18,
NPP_CONTEXT_MATCH_ERROR = -17,
NPP_FFT_FLAG_ERROR = -16,
NPP_FFT_ORDER_ERROR = -15,
NPP_STEP_ERROR = -14,
NPP_SCALE_RANGE_ERROR = -13,
NPP_DATA_TYPE_ERROR = -12,
NPP_OUT_OFF_RANGE_ERROR = -11,
NPP_DIVIDE_BY_ZERO_ERROR = -10,
NPP_MEMORY_ALLOCATION_ERR = -9,
NPP_NULL_POINTER_ERROR = -8,
NPP_RANGE_ERROR = -7,
NPP_SIZE_ERROR = -6,
NPP_BAD_ARGUMENT_ERROR = -5,
NPP_NO_MEMORY_ERROR = -4,
NPP_NOT_IMPLEMENTED_ERROR = -3,


```
NPP_ERROR = -2,  
NPP_ERROR_RESERVED = -1,  
NPP_NO_ERROR = 0,  
NPP_SUCCESS = NPP_NO_ERROR,  
NPP_NO_OPERATION_WARNING = 1,  
NPP_DIVIDE_BY_ZERO_WARNING = 6,  
NPP_AFFINE_QUAD_INCORRECT_WARNING = 28,  
NPP_WRONG_INTERSECTION_ROI_WARNING = 29,  
NPP_WRONG_INTERSECTION_QUAD_WARNING = 30,  
NPP_DOUBLE_SIZE_WARNING = 35,  
NPP_MISALIGNED_DST_ROI_WARNING = 10000 }
```

Error Status Codes.

- enum NppGpuComputeCapability {
NPP_CUDA_UNKNOWN_VERSION = -1,
NPP_CUDA_NOT_CAPABLE = 0,
NPP_CUDA_1_0 = 100,
NPP_CUDA_1_1 = 110,
NPP_CUDA_1_2 = 120,
NPP_CUDA_1_3 = 130,
NPP_CUDA_2_0 = 200,
NPP_CUDA_2_1 = 210,
NPP_CUDA_3_0 = 300,
NPP_CUDA_3_2 = 320,
NPP_CUDA_3_5 = 350,
NPP_CUDA_3_7 = 370,
NPP_CUDA_5_0 = 500,
NPP_CUDA_5_2 = 520,
NPP_CUDA_5_3 = 530,
NPP_CUDA_6_0 = 600 }
- enum NppiAxis {
NPP_HORIZONTAL_AXIS,
NPP_VERTICAL_AXIS,
NPP_BOTH_AXIS }
- enum NppCmpOp {
NPP_CMP_LESS,
NPP_CMP_LESS_EQ,
NPP_CMP_EQ,
NPP_CMP_GREATER_EQ,
NPP_CMP_GREATER }

- enum `NppRoundMode` {
`NPP_RND_NEAR`,
`NPP_ROUND_NEAREST_TIES_TO_EVEN` = `NPP_RND_NEAR`,
`NPP_RND_FINANCIAL`,
`NPP_ROUND_NEAREST_TIES_AWAY_FROM_ZERO` = `NPP_RND_FINANCIAL`,
`NPP_RND_ZERO`,
`NPP_ROUND_TOWARD_ZERO` = `NPP_RND_ZERO` }
Rounding Modes.
- enum `NppiBorderType` {
`NPP_BORDER_UNDEFINED` = 0,
`NPP_BORDER_NONE` = `NPP_BORDER_UNDEFINED`,
`NPP_BORDER_CONSTANT` = 1,
`NPP_BORDER_REPLICATE` = 2,
`NPP_BORDER_WRAP` = 3,
`NPP_BORDER_MIRROR` = 4 }
- enum `NppHintAlgorithm` {
`NPP_ALG_HINT_NONE`,
`NPP_ALG_HINT_FAST`,
`NPP_ALG_HINT_ACCURATE` }
- enum `NppiAlphaOp` {
`NPPI_OP_ALPHA_OVER`,
`NPPI_OP_ALPHA_IN`,
`NPPI_OP_ALPHA_OUT`,
`NPPI_OP_ALPHA_ATOP`,
`NPPI_OP_ALPHA_XOR`,
`NPPI_OP_ALPHA_PLUS`,
`NPPI_OP_ALPHA_OVER_PREMUL`,
`NPPI_OP_ALPHA_IN_PREMUL`,
`NPPI_OP_ALPHA_OUT_PREMUL`,
`NPPI_OP_ALPHA_ATOP_PREMUL`,
`NPPI_OP_ALPHA_XOR_PREMUL`,
`NPPI_OP_ALPHA_PLUS_PREMUL`,
`NPPI_OP_ALPHA_PREMUL` }
- enum `NppsZCType` {
`nppZCR`,
`nppZCXor`,
`nppZCC` }
- enum `NppiHuffmanTableType` {
`nppiDCTable`,
`nppiACTable` }

- enum `NppiNorm` {
 `nppiNormInf` = 0,
 `nppiNormL1` = 1,
 `nppiNormL2` = 2 }

7.2.1 Define Documentation

7.2.1.1 #define `NPP_MAX_16S` (32767)

Maximum 16-bit signed integer.

7.2.1.2 #define `NPP_MAX_16U` (65535)

Maximum 16-bit unsigned integer.

7.2.1.3 #define `NPP_MAX_32S` (2147483647)

Maximum 32-bit signed integer.

7.2.1.4 #define `NPP_MAX_32U` (4294967295U)

Maximum 32-bit unsigned integer.

7.2.1.5 #define `NPP_MAX_64S` (9223372036854775807LL)

Maximum 64-bit signed integer.

7.2.1.6 #define `NPP_MAX_64U` (18446744073709551615ULL)

Maximum 64-bit unsigned integer.

7.2.1.7 #define `NPP_MAX_8S` (127)

Maximum 8-bit signed integer.

7.2.1.8 #define `NPP_MAX_8U` (255)

Maximum 8-bit unsigned integer.

7.2.1.9 #define `NPP_MAXABS_32F` (3.402823466e+38f)

Largest positive 32-bit floating point value.

7.2.1.10 #define `NPP_MAXABS_64F` (1.7976931348623158e+308)

Largest positive 64-bit floating point value.

7.2.1.11 #define NPP_MIN_16S (-32767 - 1)

Minimum 16-bit signed integer.

7.2.1.12 #define NPP_MIN_16U (0)

Minimum 16-bit unsigned integer.

7.2.1.13 #define NPP_MIN_32S (-2147483647 - 1)

Minimum 32-bit signed integer.

7.2.1.14 #define NPP_MIN_32U (0)

Minimum 32-bit unsigned integer.

7.2.1.15 #define NPP_MIN_64S (-9223372036854775807LL - 1)

Minimum 64-bit signed integer.

7.2.1.16 #define NPP_MIN_64U (0)

Minimum 64-bit unsigned integer.

7.2.1.17 #define NPP_MIN_8S (-127 - 1)

Minimum 8-bit signed integer.

7.2.1.18 #define NPP_MIN_8U (0)

Minimum 8-bit unsigned integer.

7.2.1.19 #define NPP_MINABS_32F (1.175494351e-38f)

Smallest positive 32-bit floating point value.

7.2.1.20 #define NPP_MINABS_64F (2.2250738585072014e-308)

Smallest positive 64-bit floating point value.

7.2.2 Enumeration Type Documentation**7.2.2.1 enum NppCmpOp**

Enumerator:

NPP_CMP_LESS

NPP_CMP_LESS_EQ
NPP_CMP_EQ
NPP_CMP_GREATER_EQ
NPP_CMP_GREATER

7.2.2.2 enum NppGpuComputeCapability

Enumerator:

NPP_CUDA_UNKNOWN_VERSION Indicates that the compute-capability query failed.
NPP_CUDA_NOT_CAPABLE Indicates that no CUDA capable device was found.
NPP_CUDA_1_0 Indicates that CUDA 1.0 capable device is machine's default device.
NPP_CUDA_1_1 Indicates that CUDA 1.1 capable device is machine's default device.
NPP_CUDA_1_2 Indicates that CUDA 1.2 capable device is machine's default device.
NPP_CUDA_1_3 Indicates that CUDA 1.3 capable device is machine's default device.
NPP_CUDA_2_0 Indicates that CUDA 2.0 capable device is machine's default device.
NPP_CUDA_2_1 Indicates that CUDA 2.1 capable device is machine's default device.
NPP_CUDA_3_0 Indicates that CUDA 3.0 capable device is machine's default device.
NPP_CUDA_3_2 Indicates that CUDA 3.2 capable device is machine's default device.
NPP_CUDA_3_5 Indicates that CUDA 3.5 capable device is machine's default device.
NPP_CUDA_3_7 Indicates that CUDA 3.7 capable device is machine's default device.
NPP_CUDA_5_0 Indicates that CUDA 5.0 capable device is machine's default device.
NPP_CUDA_5_2 Indicates that CUDA 5.2 capable device is machine's default device.
NPP_CUDA_5_3 Indicates that CUDA 5.3 capable device is machine's default device.
NPP_CUDA_6_0 Indicates that CUDA 6.0 or better is machine's default device.

7.2.2.3 enum NppHintAlgorithm

Enumerator:

NPP_ALG_HINT_NONE
NPP_ALG_HINT_FAST
NPP_ALG_HINT_ACCURATE

7.2.2.4 enum NppiAlphaOp

Enumerator:

NPPI_OP_ALPHA_OVER
NPPI_OP_ALPHA_IN
NPPI_OP_ALPHA_OUT
NPPI_OP_ALPHA_ATOP
NPPI_OP_ALPHA_XOR

NPPI_OP_ALPHA_PLUS
NPPI_OP_ALPHA_OVER_PREMUL
NPPI_OP_ALPHA_IN_PREMUL
NPPI_OP_ALPHA_OUT_PREMUL
NPPI_OP_ALPHA_ATOP_PREMUL
NPPI_OP_ALPHA_XOR_PREMUL
NPPI_OP_ALPHA_PLUS_PREMUL
NPPI_OP_ALPHA_PREMUL

7.2.2.5 enum NppiAxis

Enumerator:

NPP_HORIZONTAL_AXIS
NPP_VERTICAL_AXIS
NPP_BOTH_AXIS

7.2.2.6 enum NppiBayerGridPosition

Bayer Grid Position Registration.

Enumerator:

NPPI_BAYER_BGGR Default registration position.
NPPI_BAYER_RGGB
NPPI_BAYER_GBRG
NPPI_BAYER_GRBG

7.2.2.7 enum NppiBorderType

Enumerator:

NPP_BORDER_UNDEFINED
NPP_BORDER_NONE
NPP_BORDER_CONSTANT
NPP_BORDER_REPLICATE
NPP_BORDER_WRAP
NPP_BORDER_MIRROR

7.2.2.8 enum NppiDifferentialKernel

Differential Filter types.

Enumerator:

NPP_FILTER_SOBEL
NPP_FILTER_SCHARR

7.2.2.9 enum NppiHuffmanTableType

Enumerator:

nppiDCTable DC Table.

nppiACTable AC Table.

7.2.2.10 enum NppiInterpolationMode

Filtering methods.

Enumerator:

NPPI_INTER_UNDEFINED

NPPI_INTER_NN Nearest neighbor filtering.

NPPI_INTER_LINEAR Linear interpolation.

NPPI_INTER_CUBIC Cubic interpolation.

NPPI_INTER_CUBIC2P_BSPLINE Two-parameter cubic filter (B=1, C=0).

NPPI_INTER_CUBIC2P_CATMULLROM Two-parameter cubic filter (B=0, C=1/2).

NPPI_INTER_CUBIC2P_B05C03 Two-parameter cubic filter (B=1/2, C=3/10).

NPPI_INTER_SUPER Super sampling.

NPPI_INTER_LANCZOS Lanczos filtering.

NPPI_INTER_LANCZOS3_ADVANCED Generic Lanczos filtering with order 3.

NPPI_SMOOTH_EDGE Smooth edge filtering.

7.2.2.11 enum NppiMaskSize

Fixed filter-kernel sizes.

Enumerator:

NPP_MASK_SIZE_1_X_3

NPP_MASK_SIZE_1_X_5

NPP_MASK_SIZE_3_X_1

NPP_MASK_SIZE_5_X_1

NPP_MASK_SIZE_3_X_3

NPP_MASK_SIZE_5_X_5

NPP_MASK_SIZE_7_X_7

NPP_MASK_SIZE_9_X_9

NPP_MASK_SIZE_11_X_11

NPP_MASK_SIZE_13_X_13

NPP_MASK_SIZE_15_X_15

7.2.2.12 enum NppiNorm

Enumerator:

- nppiNormInf* maximum
- nppiNormL1* sum
- nppiNormL2* square root of sum of squares

7.2.2.13 enum NppRoundMode

Rounding Modes.

The enumerated rounding modes are used by a large number of NPP primitives to allow the user to specify the method by which fractional values are converted to integer values. Also see [Rounding Modes](#).

For NPP release 5.5 new names for the three rounding modes are introduced that are based on the naming conventions for rounding modes set forth in the IEEE-754 floating-point standard. Developers are encouraged to use the new, longer names to be future proof as the legacy names will be deprecated in subsequent NPP releases.

Enumerator:

NPP_RND_NEAR Round to the nearest even integer.

All fractional numbers are rounded to their nearest integer. The ambiguous cases (i.e. $\langle \text{integer} \rangle .5$) are rounded to the closest even integer. E.g.

- `roundNear(0.5) = 0`
- `roundNear(0.6) = 1`
- `roundNear(1.5) = 2`
- `roundNear(-1.5) = -2`

NPP_ROUND_NEAREST_TIES_TO_EVEN Alias name for [NPP_RND_NEAR](#).

NPP_RND_FINANCIAL Round according to financial rule.

All fractional numbers are rounded to their nearest integer. The ambiguous cases (i.e. $\langle \text{integer} \rangle .5$) are rounded away from zero. E.g.

- `roundFinancial(0.4) = 0`
- `roundFinancial(0.5) = 1`
- `roundFinancial(-1.5) = -2`

NPP_ROUND_NEAREST_TIES_AWAY_FROM_ZERO Alias name for [NPP_RND_FINANCIAL](#).

NPP_RND_ZERO Round towards zero (truncation).

All fractional numbers of the form $\langle \text{integer} \rangle . \langle \text{decimals} \rangle$ are truncated to $\langle \text{integer} \rangle$.

- `roundZero(1.5) = 1`
- `roundZero(1.9) = 1`
- `roundZero(-2.5) = -2`

NPP_ROUND_TOWARD_ZERO Alias name for [NPP_RND_ZERO](#).

7.2.2.14 enum NppStatus

Error Status Codes.

Almost all NPP function return error-status information using these return codes. Negative return codes indicate errors, positive return codes indicate warnings, a return code of 0 indicates success.

Enumerator:

NPP_NOT_SUPPORTED_MODE_ERROR

NPP_INVALID_HOST_POINTER_ERROR

NPP_INVALID_DEVICE_POINTER_ERROR

NPP_LUT_PALETTE_BITSIZE_ERROR

NPP_ZC_MODE_NOT_SUPPORTED_ERROR ZeroCrossing mode not supported.

NPP_NOT_SUFFICIENT_COMPUTE_CAPABILITY

NPP_TEXTURE_BIND_ERROR

NPP_WRONG_INTERSECTION_ROI_ERROR

NPP_HAAR_CLASSIFIER_PIXEL_MATCH_ERROR

NPP_MEMFREE_ERROR

NPP_MEMSET_ERROR

NPP_MEMCPY_ERROR

NPP_ALIGNMENT_ERROR

NPP_CUDA_KERNEL_EXECUTION_ERROR

NPP_ROUND_MODE_NOT_SUPPORTED_ERROR Unsupported round mode.

NPP_QUALITY_INDEX_ERROR Image pixels are constant for quality index.

NPP_RESIZE_NO_OPERATION_ERROR One of the output image dimensions is less than 1 pixel.

NPP_OVERFLOW_ERROR Number overflows the upper or lower limit of the data type.

NPP_NOT_EVEN_STEP_ERROR Step value is not pixel multiple.

NPP_HISTOGRAM_NUMBER_OF_LEVELS_ERROR Number of levels for histogram is less than 2.

NPP_LUT_NUMBER_OF_LEVELS_ERROR Number of levels for LUT is less than 2.

NPP_CORRUPTED_DATA_ERROR Processed data is corrupted.

NPP_CHANNEL_ORDER_ERROR Wrong order of the destination channels.

NPP_ZERO_MASK_VALUE_ERROR All values of the mask are zero.

NPP_QUADRANGLE_ERROR The quadrangle is nonconvex or degenerates into triangle, line or point.

NPP_RECTANGLE_ERROR Size of the rectangle region is less than or equal to 1.

NPP_COEFFICIENT_ERROR Unallowable values of the transformation coefficients.

NPP_NUMBER_OF_CHANNELS_ERROR Bad or unsupported number of channels.

NPP_COI_ERROR Channel of interest is not 1, 2, or 3.

NPP_DIVISOR_ERROR Divisor is equal to zero.

NPP_CHANNEL_ERROR Illegal channel index.

NPP_STRIDE_ERROR Stride is less than the row length.

NPP_ANCHOR_ERROR Anchor point is outside mask.

NPP_MASK_SIZE_ERROR Lower bound is larger than upper bound.

NPP_RESIZE_FACTOR_ERROR

NPP_INTERPOLATION_ERROR

NPP_MIRROR_FLIP_ERROR

NPP_MOMENT_00_ZERO_ERROR

NPP_THRESHOLD_NEGATIVE_LEVEL_ERROR

NPP_THRESHOLD_ERROR

NPP_CONTEXT_MATCH_ERROR

NPP_FFT_FLAG_ERROR

NPP_FFT_ORDER_ERROR

NPP_STEP_ERROR Step is less or equal zero.

NPP_SCALE_RANGE_ERROR

NPP_DATA_TYPE_ERROR

NPP_OUT_OFF_RANGE_ERROR

NPP_DIVIDE_BY_ZERO_ERROR

NPP_MEMORY_ALLOCATION_ERR

NPP_NULL_POINTER_ERROR

NPP_RANGE_ERROR

NPP_SIZE_ERROR

NPP_BAD_ARGUMENT_ERROR

NPP_NO_MEMORY_ERROR

NPP_NOT_IMPLEMENTED_ERROR

NPP_ERROR

NPP_ERROR_RESERVED

NPP_NO_ERROR Error free operation.

NPP_SUCCESS Successful operation (same as ***NPP_NO_ERROR***).

NPP_NO_OPERATION_WARNING Indicates that no operation was performed.

NPP_DIVIDE_BY_ZERO_WARNING Divisor is zero however does not terminate the execution.

NPP_AFFINE_QUAD_INCORRECT_WARNING Indicates that the quadrangle passed to one of affine warping functions doesn't have necessary properties.
First 3 vertices are used, the fourth vertex discarded.

NPP_WRONG_INTERSECTION_ROI_WARNING The given ROI has no intersection with either the source or destination ROI.
Thus no operation was performed.

NPP_WRONG_INTERSECTION_QUAD_WARNING The given quadrangle has no intersection with either the source or destination ROI.
Thus no operation was performed.

NPP_DOUBLE_SIZE_WARNING Image size isn't multiple of two.
Indicates that in case of 422/411/420 sampling the ROI width/height was modified for proper processing.

NPP_MISALIGNED_DST_ROI_WARNING Speed reduction due to uncoalesced memory accesses warning.

7.2.2.15 enum NppsZCType**Enumerator:**

nppZCR sign change
nppZCXor sign change XOR
nppZCC sign change count_0

7.3 Basic NPP Data Types

Data Structures

- struct [NPP_ALIGN_8](#)
Complex Number This struct represents an unsigned int complex number.
- struct [NPP_ALIGN_16](#)
Complex Number This struct represents a long long complex number.

Typedefs

- typedef unsigned char [Npp8u](#)
8-bit unsigned chars
- typedef signed char [Npp8s](#)
8-bit signed chars
- typedef unsigned short [Npp16u](#)
16-bit unsigned integers
- typedef short [Npp16s](#)
16-bit signed integers
- typedef unsigned int [Npp32u](#)
32-bit unsigned integers
- typedef int [Npp32s](#)
32-bit signed integers
- typedef unsigned long long [Npp64u](#)
64-bit unsigned integers
- typedef long long [Npp64s](#)
64-bit signed integers
- typedef float [Npp32f](#)
32-bit (IEEE) floating-point numbers
- typedef double [Npp64f](#)
64-bit floating-point numbers
- typedef struct [NPP_ALIGN_8](#) [Npp32uc](#)
Complex Number This struct represents an unsigned int complex number.
- typedef struct [NPP_ALIGN_8](#) [Npp32sc](#)
Complex Number This struct represents a signed int complex number.

- typedef struct [NPP_ALIGN_8 Npp32fc](#)
Complex Number This struct represents a single floating-point complex number.
- typedef struct [NPP_ALIGN_16 Npp64sc](#)
Complex Number This struct represents a long long complex number.
- typedef struct [NPP_ALIGN_16 Npp64fc](#)
Complex Number This struct represents a double floating-point complex number.

Functions

- struct [__align__](#) (2)
Complex Number This struct represents an unsigned char complex number.
- struct [__align__](#) (4)
Complex Number This struct represents an unsigned short complex number.

Variables

- [Npp8uc](#)
- [Npp16uc](#)
- [Npp16sc](#)

7.3.1 Typedef Documentation

7.3.1.1 typedef short Npp16s

16-bit signed integers

7.3.1.2 typedef unsigned short Npp16u

16-bit unsigned integers

7.3.1.3 typedef float Npp32f

32-bit (IEEE) floating-point numbers

7.3.1.4 typedef struct NPP_ALIGN_8 Npp32fc

Complex Number This struct represents a single floating-point complex number.

7.3.1.5 typedef int Npp32s

32-bit signed integers

7.3.1.6 typedef struct NPP_ALIGN_8 Npp32sc

Complex Number This struct represents a signed int complex number.

7.3.1.7 typedef unsigned int Npp32u

32-bit unsigned integers

7.3.1.8 typedef struct NPP_ALIGN_8 Npp32uc

Complex Number This struct represents an unsigned int complex number.

7.3.1.9 typedef double Npp64f

64-bit floating-point numbers

7.3.1.10 typedef struct NPP_ALIGN_16 Npp64fc

Complex Number This struct represents a double floating-point complex number.

7.3.1.11 typedef long long Npp64s

64-bit signed integers

7.3.1.12 typedef struct NPP_ALIGN_16 Npp64sc

Complex Number This struct represents a long long complex number.

7.3.1.13 typedef unsigned long long Npp64u

64-bit unsigned integers

7.3.1.14 typedef signed char Npp8s

8-bit signed chars

7.3.1.15 typedef unsigned char Npp8u

8-bit unsigned chars

7.3.2 Function Documentation**7.3.2.1 struct __align__ (4) [read]**

Complex Number This struct represents an unsigned short complex number.

Complex Number This struct represents a short complex number.

< Real part
< Imaginary part
< Real part
< Imaginary part

7.3.2.2 `struct __align__(2)` [read]

Complex Number This struct represents an unsigned char complex number.

< Real part
< Imaginary part

7.3.3 Variable Documentation

7.3.3.1 `Npp16sc`

7.3.3.2 `Npp16uc`

7.3.3.3 `Npp8uc`

7.4 Memory Management

Routines for allocating and deallocating pitched image storage.

Functions

- void `nppiFree` (void *pData)
Free method for any 2D allocated memory.

Image Memory Allocation

ImageAllocator methods for 2D arrays of data.

The allocators have width and height parameters to specify the size of the image data being allocated. They return a pointer to the newly created memory and return the numbers of bytes between successive lines.

If the memory allocation failed due to lack of free device memory or device memory fragmentation the routine returns 0.

All allocators return memory with line strides that are beneficial for performance. It is not mandatory to use these allocators. Any valid CUDA device-memory pointers can be used by the NPP primitives and there are no restrictions on line strides.

- `Npp8u * nppiMalloc_8u_C1` (int nWidthPixels, int nHeightPixels, int *pStepBytes)
8-bit unsigned image memory allocator.
- `Npp8u * nppiMalloc_8u_C2` (int nWidthPixels, int nHeightPixels, int *pStepBytes)
2 channel 8-bit unsigned image memory allocator.
- `Npp8u * nppiMalloc_8u_C3` (int nWidthPixels, int nHeightPixels, int *pStepBytes)
3 channel 8-bit unsigned image memory allocator.
- `Npp8u * nppiMalloc_8u_C4` (int nWidthPixels, int nHeightPixels, int *pStepBytes)
4 channel 8-bit unsigned image memory allocator.
- `Npp16u * nppiMalloc_16u_C1` (int nWidthPixels, int nHeightPixels, int *pStepBytes)
16-bit unsigned image memory allocator.
- `Npp16u * nppiMalloc_16u_C2` (int nWidthPixels, int nHeightPixels, int *pStepBytes)
2 channel 16-bit unsigned image memory allocator.
- `Npp16u * nppiMalloc_16u_C3` (int nWidthPixels, int nHeightPixels, int *pStepBytes)
3 channel 16-bit unsigned image memory allocator.
- `Npp16u * nppiMalloc_16u_C4` (int nWidthPixels, int nHeightPixels, int *pStepBytes)
4 channel 16-bit unsigned image memory allocator.
- `Npp16s * nppiMalloc_16s_C1` (int nWidthPixels, int nHeightPixels, int *pStepBytes)
16-bit signed image memory allocator.

- `Npp16s * nppiMalloc_16s_C2` (int nWidthPixels, int nHeightPixels, int *pStepBytes)
2 channel 16-bit signed image memory allocator.
- `Npp16s * nppiMalloc_16s_C4` (int nWidthPixels, int nHeightPixels, int *pStepBytes)
4 channel 16-bit signed image memory allocator.
- `Npp16sc * nppiMalloc_16sc_C1` (int nWidthPixels, int nHeightPixels, int *pStepBytes)
1 channel 16-bit signed complex image memory allocator.
- `Npp16sc * nppiMalloc_16sc_C2` (int nWidthPixels, int nHeightPixels, int *pStepBytes)
2 channel 16-bit signed complex image memory allocator.
- `Npp16sc * nppiMalloc_16sc_C3` (int nWidthPixels, int nHeightPixels, int *pStepBytes)
3 channel 16-bit signed complex image memory allocator.
- `Npp16sc * nppiMalloc_16sc_C4` (int nWidthPixels, int nHeightPixels, int *pStepBytes)
4 channel 16-bit signed complex image memory allocator.
- `Npp32s * nppiMalloc_32s_C1` (int nWidthPixels, int nHeightPixels, int *pStepBytes)
32-bit signed image memory allocator.
- `Npp32s * nppiMalloc_32s_C3` (int nWidthPixels, int nHeightPixels, int *pStepBytes)
3 channel 32-bit signed image memory allocator.
- `Npp32s * nppiMalloc_32s_C4` (int nWidthPixels, int nHeightPixels, int *pStepBytes)
4 channel 32-bit signed image memory allocator.
- `Npp32sc * nppiMalloc_32sc_C1` (int nWidthPixels, int nHeightPixels, int *pStepBytes)
32-bit integer complex image memory allocator.
- `Npp32sc * nppiMalloc_32sc_C2` (int nWidthPixels, int nHeightPixels, int *pStepBytes)
2 channel 32-bit integer complex image memory allocator.
- `Npp32sc * nppiMalloc_32sc_C3` (int nWidthPixels, int nHeightPixels, int *pStepBytes)
3 channel 32-bit integer complex image memory allocator.
- `Npp32sc * nppiMalloc_32sc_C4` (int nWidthPixels, int nHeightPixels, int *pStepBytes)
4 channel 32-bit integer complex image memory allocator.
- `Npp32f * nppiMalloc_32f_C1` (int nWidthPixels, int nHeightPixels, int *pStepBytes)
32-bit floating point image memory allocator.
- `Npp32f * nppiMalloc_32f_C2` (int nWidthPixels, int nHeightPixels, int *pStepBytes)
2 channel 32-bit floating point image memory allocator.
- `Npp32f * nppiMalloc_32f_C3` (int nWidthPixels, int nHeightPixels, int *pStepBytes)
3 channel 32-bit floating point image memory allocator.
- `Npp32f * nppiMalloc_32f_C4` (int nWidthPixels, int nHeightPixels, int *pStepBytes)
4 channel 32-bit floating point image memory allocator.

- `Npp32fc * nppiMalloc_32fc_C1` (int *nWidthPixels*, int *nHeightPixels*, int **pStepBytes*)
32-bit float complex image memory allocator.
- `Npp32fc * nppiMalloc_32fc_C2` (int *nWidthPixels*, int *nHeightPixels*, int **pStepBytes*)
2 channel 32-bit float complex image memory allocator.
- `Npp32fc * nppiMalloc_32fc_C3` (int *nWidthPixels*, int *nHeightPixels*, int **pStepBytes*)
3 channel 32-bit float complex image memory allocator.
- `Npp32fc * nppiMalloc_32fc_C4` (int *nWidthPixels*, int *nHeightPixels*, int **pStepBytes*)
4 channel 32-bit float complex image memory allocator.

7.4.1 Detailed Description

Routines for allocating and deallocating pitched image storage.

These methods are provided for convenience. They allocate memory that may contain additional padding bytes at the end of each line of pixels. Though padding is not necessary for any of the NPP image-processing primitives to work correctly, its absence may cause severe performance degradation compared to properly padded images.

These functions can be found in either the `nppi` or `nppisu` libraries. Linking to only the sub-libraries that you use can significantly save link time, application load time, and CUDA runtime startup time when using dynamic libraries.

7.4.2 Function Documentation

7.4.2.1 void nppiFree (void * *pData*)

Free method for any 2D allocated memory.

This method should be used to free memory allocated with any of the `nppiMalloc_<modifier>` methods.

Parameters:

pData A pointer to memory allocated using `nppiMalloc_<modifier>`.

7.4.2.2 Npp16s* nppiMalloc_16s_C1 (int *nWidthPixels*, int *nHeightPixels*, int **pStepBytes*)

16-bit signed image memory allocator.

Parameters:

nWidthPixels Image width.
nHeightPixels Image height.
pStepBytes [Line Step](#).

Returns:

Pointer to new image data.

7.4.2.3 Npp16s* nppiMalloc_16s_C2 (int *nWidthPixels*, int *nHeightPixels*, int * *pStepBytes*)

2 channel 16-bit signed image memory allocator.

Parameters:

nWidthPixels Image width.
nHeightPixels Image height.
pStepBytes [Line Step](#).

Returns:

Pointer to new image data.

7.4.2.4 Npp16s* nppiMalloc_16s_C4 (int *nWidthPixels*, int *nHeightPixels*, int * *pStepBytes*)

4 channel 16-bit signed image memory allocator.

Parameters:

nWidthPixels Image width.
nHeightPixels Image height.
pStepBytes [Line Step](#).

Returns:

Pointer to new image data.

7.4.2.5 Npp16sc* nppiMalloc_16sc_C1 (int *nWidthPixels*, int *nHeightPixels*, int * *pStepBytes*)

1 channel 16-bit signed complex image memory allocator.

Parameters:

nWidthPixels Image width.
nHeightPixels Image height.
pStepBytes [Line Step](#).

Returns:

Pointer to new image data.

7.4.2.6 Npp16sc* nppiMalloc_16sc_C2 (int *nWidthPixels*, int *nHeightPixels*, int * *pStepBytes*)

2 channel 16-bit signed complex image memory allocator.

Parameters:

nWidthPixels Image width.
nHeightPixels Image height.

pStepBytes [Line Step](#).

Returns:

Pointer to new image data.

7.4.2.7 Npp16sc* nppiMalloc_16sc_C3 (int nWidthPixels, int nHeightPixels, int * pStepBytes)

3 channel 16-bit signed complex image memory allocator.

Parameters:

nWidthPixels Image width.

nHeightPixels Image height.

pStepBytes [Line Step](#).

Returns:

Pointer to new image data.

7.4.2.8 Npp16sc* nppiMalloc_16sc_C4 (int nWidthPixels, int nHeightPixels, int * pStepBytes)

4 channel 16-bit signed complex image memory allocator.

Parameters:

nWidthPixels Image width.

nHeightPixels Image height.

pStepBytes [Line Step](#).

Returns:

Pointer to new image data.

7.4.2.9 Npp16u* nppiMalloc_16u_C1 (int nWidthPixels, int nHeightPixels, int * pStepBytes)

16-bit unsigned image memory allocator.

Parameters:

nWidthPixels Image width.

nHeightPixels Image height.

pStepBytes [Line Step](#).

Returns:

Pointer to new image data.

7.4.2.10 Npp16u* nppiMalloc_16u_C2 (int nWidthPixels, int nHeightPixels, int * pStepBytes)

2 channel 16-bit unsigned image memory allocator.

Parameters:

nWidthPixels Image width.
nHeightPixels Image height.
pStepBytes [Line Step](#).

Returns:

Pointer to new image data.

7.4.2.11 Npp16u* nppiMalloc_16u_C3 (int nWidthPixels, int nHeightPixels, int * pStepBytes)

3 channel 16-bit unsigned image memory allocator.

Parameters:

nWidthPixels Image width.
nHeightPixels Image height.
pStepBytes [Line Step](#).

Returns:

Pointer to new image data.

7.4.2.12 Npp16u* nppiMalloc_16u_C4 (int nWidthPixels, int nHeightPixels, int * pStepBytes)

4 channel 16-bit unsigned image memory allocator.

Parameters:

nWidthPixels Image width.
nHeightPixels Image height.
pStepBytes [Line Step](#).

Returns:

Pointer to new image data.

7.4.2.13 Npp32f* nppiMalloc_32f_C1 (int nWidthPixels, int nHeightPixels, int * pStepBytes)

32-bit floating point image memory allocator.

Parameters:

nWidthPixels Image width.
nHeightPixels Image height.

pStepBytes [Line Step](#).

Returns:

Pointer to new image data.

7.4.2.14 Npp32f* nppiMalloc_32f_C2 (int nWidthPixels, int nHeightPixels, int * pStepBytes)

2 channel 32-bit floating point image memory allocator.

Parameters:

nWidthPixels Image width.

nHeightPixels Image height.

pStepBytes [Line Step](#).

Returns:

Pointer to new image data.

7.4.2.15 Npp32f* nppiMalloc_32f_C3 (int nWidthPixels, int nHeightPixels, int * pStepBytes)

3 channel 32-bit floating point image memory allocator.

Parameters:

nWidthPixels Image width.

nHeightPixels Image height.

pStepBytes [Line Step](#).

Returns:

Pointer to new image data.

7.4.2.16 Npp32f* nppiMalloc_32f_C4 (int nWidthPixels, int nHeightPixels, int * pStepBytes)

4 channel 32-bit floating point image memory allocator.

Parameters:

nWidthPixels Image width.

nHeightPixels Image height.

pStepBytes [Line Step](#).

Returns:

Pointer to new image data.

7.4.2.17 Npp32fc* nppiMalloc_32fc_C1 (int nWidthPixels, int nHeightPixels, int * pStepBytes)

32-bit float complex image memory allocator.

Parameters:

nWidthPixels Image width.
nHeightPixels Image height.
pStepBytes [Line Step](#).

Returns:

Pointer to new image data.

7.4.2.18 Npp32fc* nppiMalloc_32fc_C2 (int nWidthPixels, int nHeightPixels, int * pStepBytes)

2 channel 32-bit float complex image memory allocator.

Parameters:

nWidthPixels Image width.
nHeightPixels Image height.
pStepBytes [Line Step](#).

Returns:

Pointer to new image data.

7.4.2.19 Npp32fc* nppiMalloc_32fc_C3 (int nWidthPixels, int nHeightPixels, int * pStepBytes)

3 channel 32-bit float complex image memory allocator.

Parameters:

nWidthPixels Image width.
nHeightPixels Image height.
pStepBytes [Line Step](#).

Returns:

Pointer to new image data.

7.4.2.20 Npp32fc* nppiMalloc_32fc_C4 (int nWidthPixels, int nHeightPixels, int * pStepBytes)

4 channel 32-bit float complex image memory allocator.

Parameters:

nWidthPixels Image width.
nHeightPixels Image height.

pStepBytes [Line Step](#).

Returns:

Pointer to new image data.

7.4.2.21 `Npp32s* nppiMalloc_32s_C1 (int nWidthPixels, int nHeightPixels, int * pStepBytes)`

32-bit signed image memory allocator.

Parameters:

nWidthPixels Image width.

nHeightPixels Image height.

pStepBytes [Line Step](#).

Returns:

Pointer to new image data.

7.4.2.22 `Npp32s* nppiMalloc_32s_C3 (int nWidthPixels, int nHeightPixels, int * pStepBytes)`

3 channel 32-bit signed image memory allocator.

Parameters:

nWidthPixels Image width.

nHeightPixels Image height.

pStepBytes [Line Step](#).

Returns:

Pointer to new image data.

7.4.2.23 `Npp32s* nppiMalloc_32s_C4 (int nWidthPixels, int nHeightPixels, int * pStepBytes)`

4 channel 32-bit signed image memory allocator.

Parameters:

nWidthPixels Image width.

nHeightPixels Image height.

pStepBytes [Line Step](#).

Returns:

Pointer to new image data.

7.4.2.24 Npp32sc* nppiMalloc_32sc_C1 (int nWidthPixels, int nHeightPixels, int * pStepBytes)

32-bit integer complex image memory allocator.

Parameters:

nWidthPixels Image width.
nHeightPixels Image height.
pStepBytes [Line Step](#).

Returns:

Pointer to new image data.

7.4.2.25 Npp32sc* nppiMalloc_32sc_C2 (int nWidthPixels, int nHeightPixels, int * pStepBytes)

2 channel 32-bit integer complex image memory allocator.

Parameters:

nWidthPixels Image width.
nHeightPixels Image height.
pStepBytes [Line Step](#).

Returns:

Pointer to new image data.

7.4.2.26 Npp32sc* nppiMalloc_32sc_C3 (int nWidthPixels, int nHeightPixels, int * pStepBytes)

3 channel 32-bit integer complex image memory allocator.

Parameters:

nWidthPixels Image width.
nHeightPixels Image height.
pStepBytes [Line Step](#).

Returns:

Pointer to new image data.

7.4.2.27 Npp32sc* nppiMalloc_32sc_C4 (int nWidthPixels, int nHeightPixels, int * pStepBytes)

4 channel 32-bit integer complex image memory allocator.

Parameters:

nWidthPixels Image width.
nHeightPixels Image height.

pStepBytes [Line Step](#).

Returns:

Pointer to new image data.

7.4.2.28 `Npp8u* nppiMalloc_8u_C1 (int nWidthPixels, int nHeightPixels, int * pStepBytes)`

8-bit unsigned image memory allocator.

Parameters:

nWidthPixels Image width.

nHeightPixels Image height.

pStepBytes [Line Step](#).

Returns:

Pointer to new image data.

7.4.2.29 `Npp8u* nppiMalloc_8u_C2 (int nWidthPixels, int nHeightPixels, int * pStepBytes)`

2 channel 8-bit unsigned image memory allocator.

Parameters:

nWidthPixels Image width.

nHeightPixels Image height.

pStepBytes [Line Step](#).

Returns:

Pointer to new image data.

7.4.2.30 `Npp8u* nppiMalloc_8u_C3 (int nWidthPixels, int nHeightPixels, int * pStepBytes)`

3 channel 8-bit unsigned image memory allocator.

Parameters:

nWidthPixels Image width.

nHeightPixels Image height.

pStepBytes [Line Step](#).

Returns:

Pointer to new image data.

7.4.2.31 Npp8u* nppiMalloc_8u_C4 (int *nWidthPixels*, int *nHeightPixels*, int * *pStepBytes*)

4 channel 8-bit unsigned image memory allocator.

Parameters:

nWidthPixels Image width.

nHeightPixels Image height.

pStepBytes [Line Step](#).

Returns:

Pointer to new image data.

7.5 Data Exchange and Initialization

Primitives for initializing, copying and converting image data.

Modules

- [Set](#)

Primitives for setting pixels to a specific value.

- [Copy](#)
- [Convert](#)
- [Scale](#)
- [Copy Constant Border](#)
- [Copy Replicate Border](#)
- [Copy Wrap Border](#)
- [Copy Sub-Pixel](#)
- [Duplicate Channel](#)
- [Transpose](#)
- [Swap Channels](#)

7.5.1 Detailed Description

Primitives for initializing, copying and converting image data.

These functions can be found in either the nppi or nppidei libraries. Linking to only the sub-libraries that you use can significantly save link time, application load time, and CUDA runtime startup time when using dynamic libraries.

7.6 Set

Primitives for setting pixels to a specific value.

Set

Set all pixels within the ROI to a specific value.

- `NppStatus nppiSet_8s_C1R` (const `Npp8s` nValue, `Npp8s` *pDst, int nDstStep, `NppiSize` oSizeROI)
8-bit image set.
- `NppStatus nppiSet_8s_C2R` (const `Npp8s` aValue[2], `Npp8s` *pDst, int nDstStep, `NppiSize` oSizeROI)
8-bit two-channel image set.
- `NppStatus nppiSet_8s_C3R` (const `Npp8s` aValue[3], `Npp8s` *pDst, int nDstStep, `NppiSize` oSizeROI)
8-bit three-channel image set.
- `NppStatus nppiSet_8s_C4R` (const `Npp8s` aValue[4], `Npp8s` *pDst, int nDstStep, `NppiSize` oSizeROI)
8-bit four-channel image set.
- `NppStatus nppiSet_8s_AC4R` (const `Npp8s` aValue[3], `Npp8s` *pDst, int nDstStep, `NppiSize` oSizeROI)
8-bit four-channel image set ignoring alpha channel.
- `NppStatus nppiSet_8u_C1R` (const `Npp8u` nValue, `Npp8u` *pDst, int nDstStep, `NppiSize` oSizeROI)
8-bit unsigned image set.
- `NppStatus nppiSet_8u_C2R` (const `Npp8u` aValue[2], `Npp8u` *pDst, int nDstStep, `NppiSize` oSizeROI)
2 channel 8-bit unsigned image set.
- `NppStatus nppiSet_8u_C3R` (const `Npp8u` aValue[3], `Npp8u` *pDst, int nDstStep, `NppiSize` oSizeROI)
3 channel 8-bit unsigned image set.
- `NppStatus nppiSet_8u_C4R` (const `Npp8u` aValue[4], `Npp8u` *pDst, int nDstStep, `NppiSize` oSizeROI)
4 channel 8-bit unsigned image set.
- `NppStatus nppiSet_8u_AC4R` (const `Npp8u` aValue[3], `Npp8u` *pDst, int nDstStep, `NppiSize` oSizeROI)
4 channel 8-bit unsigned image set method, not affecting Alpha channel.
- `NppStatus nppiSet_16u_C1R` (const `Npp16u` nValue, `Npp16u` *pDst, int nDstStep, `NppiSize` oSizeROI)
16-bit unsigned image set.

- [NppStatus nppiSet_16u_C2R](#) (const [Npp16u](#) aValue[2], [Npp16u](#) *pDst, int nDstStep, [NppiSize](#) oSizeROI)
2 channel 16-bit unsigned image set.
- [NppStatus nppiSet_16u_C3R](#) (const [Npp16u](#) aValue[3], [Npp16u](#) *pDst, int nDstStep, [NppiSize](#) oSizeROI)
3 channel 16-bit unsigned image set.
- [NppStatus nppiSet_16u_C4R](#) (const [Npp16u](#) aValue[4], [Npp16u](#) *pDst, int nDstStep, [NppiSize](#) oSizeROI)
4 channel 16-bit unsigned image set.
- [NppStatus nppiSet_16u_AC4R](#) (const [Npp16u](#) aValue[3], [Npp16u](#) *pDst, int nDstStep, [NppiSize](#) oSizeROI)
4 channel 16-bit unsigned image set method, not affecting Alpha channel.
- [NppStatus nppiSet_16s_C1R](#) (const [Npp16s](#) nValue, [Npp16s](#) *pDst, int nDstStep, [NppiSize](#) oSizeROI)
16-bit image set.
- [NppStatus nppiSet_16s_C2R](#) (const [Npp16s](#) aValue[2], [Npp16s](#) *pDst, int nDstStep, [NppiSize](#) oSizeROI)
2 channel 16-bit image set.
- [NppStatus nppiSet_16s_C3R](#) (const [Npp16s](#) aValue[3], [Npp16s](#) *pDst, int nDstStep, [NppiSize](#) oSizeROI)
3 channel 16-bit image set.
- [NppStatus nppiSet_16s_C4R](#) (const [Npp16s](#) aValue[4], [Npp16s](#) *pDst, int nDstStep, [NppiSize](#) oSizeROI)
4 channel 16-bit image set.
- [NppStatus nppiSet_16s_AC4R](#) (const [Npp16s](#) aValue[3], [Npp16s](#) *pDst, int nDstStep, [NppiSize](#) oSizeROI)
4 channel 16-bit image set method, not affecting Alpha channel.
- [NppStatus nppiSet_16sc_C1R](#) (const [Npp16sc](#) oValue, [Npp16sc](#) *pDst, int nDstStep, [NppiSize](#) oSizeROI)
16-bit complex integer image set.
- [NppStatus nppiSet_16sc_C2R](#) (const [Npp16sc](#) aValue[2], [Npp16sc](#) *pDst, int nDstStep, [NppiSize](#) oSizeROI)
16-bit complex integer two-channel image set.
- [NppStatus nppiSet_16sc_C3R](#) (const [Npp16sc](#) aValue[3], [Npp16sc](#) *pDst, int nDstStep, [NppiSize](#) oSizeROI)
16-bit complex integer three-channel image set.
- [NppStatus nppiSet_16sc_C4R](#) (const [Npp16sc](#) aValue[4], [Npp16sc](#) *pDst, int nDstStep, [NppiSize](#) oSizeROI)

16-bit complex integer four-channel image set.

- `NppStatus nppiSet_16sc_AC4R` (const `Npp16sc` aValue[3], `Npp16sc` *pDst, int nDstStep, `NppiSize` oSizeROI)

16-bit complex integer four-channel image set ignoring alpha.

- `NppStatus nppiSet_32s_C1R` (const `Npp32s` nValue, `Npp32s` *pDst, int nDstStep, `NppiSize` oSizeROI)

32-bit image set.

- `NppStatus nppiSet_32s_C2R` (const `Npp32s` aValue[2], `Npp32s` *pDst, int nDstStep, `NppiSize` oSizeROI)

2 channel 32-bit image set.

- `NppStatus nppiSet_32s_C3R` (const `Npp32s` aValue[3], `Npp32s` *pDst, int nDstStep, `NppiSize` oSizeROI)

3 channel 32-bit image set.

- `NppStatus nppiSet_32s_C4R` (const `Npp32s` aValue[4], `Npp32s` *pDst, int nDstStep, `NppiSize` oSizeROI)

4 channel 32-bit image set.

- `NppStatus nppiSet_32s_AC4R` (const `Npp32s` aValue[3], `Npp32s` *pDst, int nDstStep, `NppiSize` oSizeROI)

4 channel 32-bit image set method, not affecting Alpha channel.

- `NppStatus nppiSet_32u_C1R` (const `Npp32u` nValue, `Npp32u` *pDst, int nDstStep, `NppiSize` oSizeROI)

32-bit unsigned image set.

- `NppStatus nppiSet_32u_C2R` (const `Npp32u` aValue[2], `Npp32u` *pDst, int nDstStep, `NppiSize` oSizeROI)

2 channel 32-bit unsigned image set.

- `NppStatus nppiSet_32u_C3R` (const `Npp32u` aValue[3], `Npp32u` *pDst, int nDstStep, `NppiSize` oSizeROI)

3 channel 32-bit unsigned image set.

- `NppStatus nppiSet_32u_C4R` (const `Npp32u` aValue[4], `Npp32u` *pDst, int nDstStep, `NppiSize` oSizeROI)

4 channel 32-bit unsigned image set.

- `NppStatus nppiSet_32u_AC4R` (const `Npp32u` aValue[3], `Npp32u` *pDst, int nDstStep, `NppiSize` oSizeROI)

4 channel 32-bit unsigned image set method, not affecting Alpha channel.

- `NppStatus nppiSet_32sc_C1R` (const `Npp32sc` oValue, `Npp32sc` *pDst, int nDstStep, `NppiSize` oSizeROI)

Single channel 32-bit complex integer image set.

- [NppStatus nppiSet_32sc_C2R](#) (const [Npp32sc](#) aValue[2], [Npp32sc](#) *pDst, int nDstStep, [NppiSize](#) oSizeROI)
Two channel 32-bit complex integer image set.
- [NppStatus nppiSet_32sc_C3R](#) (const [Npp32sc](#) aValue[3], [Npp32sc](#) *pDst, int nDstStep, [NppiSize](#) oSizeROI)
Three channel 32-bit complex integer image set.
- [NppStatus nppiSet_32sc_C4R](#) (const [Npp32sc](#) aValue[4], [Npp32sc](#) *pDst, int nDstStep, [NppiSize](#) oSizeROI)
Four channel 32-bit complex integer image set.
- [NppStatus nppiSet_32sc_AC4R](#) (const [Npp32sc](#) aValue[3], [Npp32sc](#) *pDst, int nDstStep, [NppiSize](#) oSizeROI)
32-bit complex integer four-channel image set ignoring alpha.
- [NppStatus nppiSet_32f_C1R](#) (const [Npp32f](#) nValue, [Npp32f](#) *pDst, int nDstStep, [NppiSize](#) oSizeROI)
32-bit floating point image set.
- [NppStatus nppiSet_32f_C2R](#) (const [Npp32f](#) aValue[2], [Npp32f](#) *pDst, int nDstStep, [NppiSize](#) oSizeROI)
2 channel 32-bit floating point image set.
- [NppStatus nppiSet_32f_C3R](#) (const [Npp32f](#) aValue[3], [Npp32f](#) *pDst, int nDstStep, [NppiSize](#) oSizeROI)
3 channel 32-bit floating point image set.
- [NppStatus nppiSet_32f_C4R](#) (const [Npp32f](#) aValue[4], [Npp32f](#) *pDst, int nDstStep, [NppiSize](#) oSizeROI)
4 channel 32-bit floating point image set.
- [NppStatus nppiSet_32f_AC4R](#) (const [Npp32f](#) aValue[3], [Npp32f](#) *pDst, int nDstStep, [NppiSize](#) oSizeROI)
4 channel 32-bit floating point image set method, not affecting Alpha channel.
- [NppStatus nppiSet_32fc_C1R](#) (const [Npp32fc](#) oValue, [Npp32fc](#) *pDst, int nDstStep, [NppiSize](#) oSizeROI)
Single channel 32-bit complex image set.
- [NppStatus nppiSet_32fc_C2R](#) (const [Npp32fc](#) aValue[2], [Npp32fc](#) *pDst, int nDstStep, [NppiSize](#) oSizeROI)
Two channel 32-bit complex image set.
- [NppStatus nppiSet_32fc_C3R](#) (const [Npp32fc](#) aValue[3], [Npp32fc](#) *pDst, int nDstStep, [NppiSize](#) oSizeROI)
Three channel 32-bit complex image set.
- [NppStatus nppiSet_32fc_C4R](#) (const [Npp32fc](#) aValue[4], [Npp32fc](#) *pDst, int nDstStep, [NppiSize](#) oSizeROI)
Four channel 32-bit complex image set.

- `NppStatus nppiSet_32fc_AC4R` (const `Npp32fc` aValue[3], `Npp32fc` *pDst, int nDstStep, `NppiSize` oSizeROI)

32-bit complex four-channel image set ignoring alpha.

Masked Set

The masked set primitives have an additional "mask image" input.

The mask controls which pixels within the ROI are set. For details see [Masked Operation](#).

- `NppStatus nppiSet_8u_C1MR` (`Npp8u` nValue, `Npp8u` *pDst, int nDstStep, `NppiSize` oSizeROI, const `Npp8u` *pMask, int nMaskStep)

Masked 8-bit unsigned image set.

- `NppStatus nppiSet_8u_C3MR` (const `Npp8u` aValue[3], `Npp8u` *pDst, int nDstStep, `NppiSize` oSizeROI, const `Npp8u` *pMask, int nMaskStep)

Masked 3 channel 8-bit unsigned image set.

- `NppStatus nppiSet_8u_C4MR` (const `Npp8u` aValue[4], `Npp8u` *pDst, int nDstStep, `NppiSize` oSizeROI, const `Npp8u` *pMask, int nMaskStep)

Masked 4 channel 8-bit unsigned image set.

- `NppStatus nppiSet_8u_AC4MR` (const `Npp8u` aValue[3], `Npp8u` *pDst, int nDstStep, `NppiSize` oSizeROI, const `Npp8u` *pMask, int nMaskStep)

Masked 4 channel 8-bit unsigned image set method, not affecting Alpha channel.

- `NppStatus nppiSet_16u_C1MR` (`Npp16u` nValue, `Npp16u` *pDst, int nDstStep, `NppiSize` oSizeROI, const `Npp8u` *pMask, int nMaskStep)

Masked 16-bit unsigned image set.

- `NppStatus nppiSet_16u_C3MR` (const `Npp16u` aValue[3], `Npp16u` *pDst, int nDstStep, `NppiSize` oSizeROI, const `Npp8u` *pMask, int nMaskStep)

Masked 3 channel 16-bit unsigned image set.

- `NppStatus nppiSet_16u_C4MR` (const `Npp16u` aValue[4], `Npp16u` *pDst, int nDstStep, `NppiSize` oSizeROI, const `Npp8u` *pMask, int nMaskStep)

Masked 4 channel 16-bit unsigned image set.

- `NppStatus nppiSet_16u_AC4MR` (const `Npp16u` aValue[3], `Npp16u` *pDst, int nDstStep, `NppiSize` oSizeROI, const `Npp8u` *pMask, int nMaskStep)

Masked 4 channel 16-bit unsigned image set method, not affecting Alpha channel.

- `NppStatus nppiSet_16s_C1MR` (`Npp16s` nValue, `Npp16s` *pDst, int nDstStep, `NppiSize` oSizeROI, const `Npp8u` *pMask, int nMaskStep)

Masked 16-bit image set.

- `NppStatus nppiSet_16s_C3MR` (const `Npp16s` aValue[3], `Npp16s` *pDst, int nDstStep, `NppiSize` oSizeROI, const `Npp8u` *pMask, int nMaskStep)

Masked 3 channel 16-bit image set.

- `NppStatus nppiSet_16s_C4MR` (const `Npp16s` aValue[4], `Npp16s` *pDst, int nDstStep, `NppiSize` oSizeROI, const `Npp8u` *pMask, int nMaskStep)
Masked 4 channel 16-bit image set.
- `NppStatus nppiSet_16s_AC4MR` (const `Npp16s` aValue[3], `Npp16s` *pDst, int nDstStep, `NppiSize` oSizeROI, const `Npp8u` *pMask, int nMaskStep)
Masked 4 channel 16-bit image set method, not affecting Alpha channel.
- `NppStatus nppiSet_32s_C1MR` (`Npp32s` nValue, `Npp32s` *pDst, int nDstStep, `NppiSize` oSizeROI, const `Npp8u` *pMask, int nMaskStep)
Masked 32-bit image set.
- `NppStatus nppiSet_32s_C3MR` (const `Npp32s` aValue[3], `Npp32s` *pDst, int nDstStep, `NppiSize` oSizeROI, const `Npp8u` *pMask, int nMaskStep)
Masked 3 channel 32-bit image set.
- `NppStatus nppiSet_32s_C4MR` (const `Npp32s` aValue[4], `Npp32s` *pDst, int nDstStep, `NppiSize` oSizeROI, const `Npp8u` *pMask, int nMaskStep)
Masked 4 channel 32-bit image set.
- `NppStatus nppiSet_32s_AC4MR` (const `Npp32s` aValue[3], `Npp32s` *pDst, int nDstStep, `NppiSize` oSizeROI, const `Npp8u` *pMask, int nMaskStep)
Masked 4 channel 16-bit image set method, not affecting Alpha channel.
- `NppStatus nppiSet_32f_C1MR` (`Npp32f` nValue, `Npp32f` *pDst, int nDstStep, `NppiSize` oSizeROI, const `Npp8u` *pMask, int nMaskStep)
Masked 32-bit floating point image set.
- `NppStatus nppiSet_32f_C3MR` (const `Npp32f` aValue[3], `Npp32f` *pDst, int nDstStep, `NppiSize` oSizeROI, const `Npp8u` *pMask, int nMaskStep)
Masked 3 channel 32-bit floating point image set.
- `NppStatus nppiSet_32f_C4MR` (const `Npp32f` aValue[4], `Npp32f` *pDst, int nDstStep, `NppiSize` oSizeROI, const `Npp8u` *pMask, int nMaskStep)
Masked 4 channel 32-bit floating point image set.
- `NppStatus nppiSet_32f_AC4MR` (const `Npp32f` aValue[3], `Npp32f` *pDst, int nDstStep, `NppiSize` oSizeROI, const `Npp8u` *pMask, int nMaskStep)
Masked 4 channel 32-bit floating point image set method, not affecting Alpha channel.

Channel Set

The select-channel set primitives set a single color channel in multi-channel images to a given value.

The channel is selected by adjusting the pDst pointer to point to the desired color channel (see [Channel-of-Interest API](#)).

- `NppStatus nppiSet_8u_C3CR` (`Npp8u` nValue, `Npp8u` *pDst, int nDstStep, `NppiSize` oSizeROI)
3 channel 8-bit unsigned image set affecting only single channel.

- `NppStatus nppiSet_8u_C4CR` (`Npp8u` `nValue`, `Npp8u` `*pDst`, `int` `nDstStep`, `NppiSize` `oSizeROI`)
4 channel 8-bit unsigned image set affecting only single channel.
- `NppStatus nppiSet_16u_C3CR` (`Npp16u` `nValue`, `Npp16u` `*pDst`, `int` `nDstStep`, `NppiSize` `oSizeROI`)
3 channel 16-bit unsigned image set affecting only single channel.
- `NppStatus nppiSet_16u_C4CR` (`Npp16u` `nValue`, `Npp16u` `*pDst`, `int` `nDstStep`, `NppiSize` `oSizeROI`)
4 channel 16-bit unsigned image set affecting only single channel.
- `NppStatus nppiSet_16s_C3CR` (`Npp16s` `nValue`, `Npp16s` `*pDst`, `int` `nDstStep`, `NppiSize` `oSizeROI`)
3 channel 16-bit signed image set affecting only single channel.
- `NppStatus nppiSet_16s_C4CR` (`Npp16s` `nValue`, `Npp16s` `*pDst`, `int` `nDstStep`, `NppiSize` `oSizeROI`)
4 channel 16-bit signed image set affecting only single channel.
- `NppStatus nppiSet_32s_C3CR` (`Npp32s` `nValue`, `Npp32s` `*pDst`, `int` `nDstStep`, `NppiSize` `oSizeROI`)
3 channel 32-bit signed image set affecting only single channel.
- `NppStatus nppiSet_32s_C4CR` (`Npp32s` `nValue`, `Npp32s` `*pDst`, `int` `nDstStep`, `NppiSize` `oSizeROI`)
4 channel 32-bit signed image set affecting only single channel.
- `NppStatus nppiSet_32f_C3CR` (`Npp32f` `nValue`, `Npp32f` `*pDst`, `int` `nDstStep`, `NppiSize` `oSizeROI`)
3 channel 32-bit floating point image set affecting only single channel.
- `NppStatus nppiSet_32f_C4CR` (`Npp32f` `nValue`, `Npp32f` `*pDst`, `int` `nDstStep`, `NppiSize` `oSizeROI`)
4 channel 32-bit floating point image set affecting only single channel.

7.6.1 Detailed Description

Primitives for setting pixels to a specific value.

7.6.2 Function Documentation

7.6.2.1 `NppStatus nppiSet_16s_AC4MR` (`const Npp16s aValue[3]`, `Npp16s *pDst`, `int nDstStep`, `NppiSize oSizeROI`, `const Npp8u *pMask`, `int nMaskStep`)

Masked 4 channel 16-bit image set method, not affecting Alpha channel.

Parameters:

aValue The pixel-value to be set.

pDst Destination-Image Pointer.

nDstStep Destination-Image Line Step.

oSizeROI Region-of-Interest (ROI).

pMask Mask-Image Pointer.

nMaskStep Mask-Image Line Step.

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.6.2.2 NppStatus nppiSet_16s_AC4R (const Npp16s aValue[3], Npp16s * pDst, int nDstStep, NppiSize oSizeROI)

4 channel 16-bit image set method, not affecting Alpha channel.

Parameters:

aValue The pixel-value to be set.

pDst Destination-Image Pointer.

nDstStep Destination-Image Line Step.

oSizeROI Region-of-Interest (ROI).

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.6.2.3 NppStatus nppiSet_16s_C1MR (Npp16s nValue, Npp16s * pDst, int nDstStep, NppiSize oSizeROI, const Npp8u * pMask, int nMaskStep)

Masked 16-bit image set.

Parameters:

nValue The pixel-value to be set.

pDst Destination-Image Pointer.

nDstStep Destination-Image Line Step.

oSizeROI Region-of-Interest (ROI).

pMask Mask-Image Pointer.

nMaskStep Mask-Image Line Step.

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.6.2.4 NppStatus nppiSet_16s_C1R (const Npp16s nValue, Npp16s * pDst, int nDstStep, NppiSize oSizeROI)

16-bit image set.

Parameters:

nValue The pixel-value to be set.

pDst Destination-Image Pointer.

nDstStep Destination-Image Line Step.

oSizeROI Region-of-Interest (ROI).

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.6.2.5 NppStatus nppiSet_16s_C2R (const Npp16s aValue[2], Npp16s * pDst, int nDstStep, NppiSize oSizeROI)

2 channel 16-bit image set.

Parameters:

aValue The pixel-value to be set.

pDst Destination-Image Pointer.

nDstStep Destination-Image Line Step.

oSizeROI Region-of-Interest (ROI).

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.6.2.6 NppStatus nppiSet_16s_C3CR (Npp16s nValue, Npp16s * pDst, int nDstStep, NppiSize oSizeROI)

3 channel 16-bit signed image set affecting only single channel.

Parameters:

nValue The pixel-value to be set.

pDst Select-Channel Destination-Image Pointer.

nDstStep Destination-Image Line Step.

oSizeROI Region-of-Interest (ROI).

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.6.2.7 NppStatus nppiSet_16s_C3MR (const Npp16s aValue[3], Npp16s * pDst, int nDstStep, NppiSize oSizeROI, const Npp8u * pMask, int nMaskStep)

Masked 3 channel 16-bit image set.

Parameters:

aValue The pixel-value to be set.

pDst Destination-Image Pointer.

nDstStep Destination-Image Line Step.

oSizeROI Region-of-Interest (ROI).

pMask Mask-Image Pointer.

nMaskStep Mask-Image Line Step.

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.6.2.8 NppStatus nppiSet_16s_C3R (const Npp16s aValue[3], Npp16s * pDst, int nDstStep, NppiSize oSizeROI)

3 channel 16-bit image set.

Parameters:

aValue The pixel-value to be set.

pDst Destination-Image Pointer.

nDstStep Destination-Image Line Step.

oSizeROI Region-of-Interest (ROI).

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.6.2.9 NppStatus nppiSet_16s_C4CR (Npp16s nValue, Npp16s * pDst, int nDstStep, NppiSize oSizeROI)

4 channel 16-bit signed image set affecting only single channel.

Parameters:

nValue The pixel-value to be set.

pDst Select-Channel Destination-Image Pointer.

nDstStep Destination-Image Line Step.

oSizeROI Region-of-Interest (ROI).

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.6.2.10 NppStatus nppiSet_16s_C4MR (const Npp16s aValue[4], Npp16s * pDst, int nDstStep, NppiSize oSizeROI, const Npp8u * pMask, int nMaskStep)

Masked 4 channel 16-bit image set.

Parameters:

aValue The pixel-value to be set.

pDst Destination-Image Pointer.

nDstStep Destination-Image Line Step.

oSizeROI Region-of-Interest (ROI).

pMask Mask-Image Pointer.

nMaskStep Mask-Image Line Step.

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.6.2.11 NppStatus nppiSet_16s_C4R (const Npp16s aValue[4], Npp16s * pDst, int nDstStep, NppiSize oSizeROI)

4 channel 16-bit image set.

Parameters:

aValue The pixel-value to be set.

pDst Destination-Image Pointer.

nDstStep Destination-Image Line Step.

oSizeROI Region-of-Interest (ROI).

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.6.2.12 NppStatus nppiSet_16sc_AC4R (const Npp16sc aValue[3], Npp16sc * pDst, int nDstStep, NppiSize oSizeROI)

16-bit complex integer four-channel image set ignoring alpha.

Parameters:

aValue The pixel-value to be set.

pDst Destination-Image Pointer.

nDstStep Destination-Image Line Step.

oSizeROI Region-of-Interest (ROI).

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.6.2.13 NppStatus nppiSet_16sc_C1R (const Npp16sc oValue, Npp16sc * pDst, int nDstStep, NppiSize oSizeROI)

16-bit complex integer image set.

Parameters:

oValue The pixel-value to be set.

pDst Destination-Image Pointer.
nDstStep Destination-Image Line Step.
oSizeROI Region-of-Interest (ROI).

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.6.2.14 NppStatus nppiSet_16sc_C2R (const Npp16sc aValue[2], Npp16sc * pDst, int nDstStep, NppiSize oSizeROI)

16-bit complex integer two-channel image set.

Parameters:

aValue The pixel-value to be set.
pDst Destination-Image Pointer.
nDstStep Destination-Image Line Step.
oSizeROI Region-of-Interest (ROI).

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.6.2.15 NppStatus nppiSet_16sc_C3R (const Npp16sc aValue[3], Npp16sc * pDst, int nDstStep, NppiSize oSizeROI)

16-bit complex integer three-channel image set.

Parameters:

aValue The pixel-value to be set.
pDst Destination-Image Pointer.
nDstStep Destination-Image Line Step.
oSizeROI Region-of-Interest (ROI).

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.6.2.16 NppStatus nppiSet_16sc_C4R (const Npp16sc aValue[4], Npp16sc * pDst, int nDstStep, NppiSize oSizeROI)

16-bit complex integer four-channel image set.

Parameters:

aValue The pixel-value to be set.
pDst Destination-Image Pointer.

nDstStep Destination-Image Line Step.

oSizeROI Region-of-Interest (ROI).

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.6.2.17 `NppStatus nppiSet_16u_AC4MR (const Npp16u aValue[3], Npp16u * pDst, int nDstStep, NppiSize oSizeROI, const Npp8u * pMask, int nMaskStep)`

Masked 4 channel 16-bit unsigned image set method, not affecting Alpha channel.

Parameters:

aValue The pixel-value to be set.

pDst Destination-Image Pointer.

nDstStep Destination-Image Line Step.

oSizeROI Region-of-Interest (ROI).

pMask Mask-Image Pointer.

nMaskStep Mask-Image Line Step.

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.6.2.18 `NppStatus nppiSet_16u_AC4R (const Npp16u aValue[3], Npp16u * pDst, int nDstStep, NppiSize oSizeROI)`

4 channel 16-bit unsigned image set method, not affecting Alpha channel.

Parameters:

aValue The pixel-value to be set.

pDst Destination-Image Pointer.

nDstStep Destination-Image Line Step.

oSizeROI Region-of-Interest (ROI).

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.6.2.19 `NppStatus nppiSet_16u_C1MR (Npp16u nValue, Npp16u * pDst, int nDstStep, NppiSize oSizeROI, const Npp8u * pMask, int nMaskStep)`

Masked 16-bit unsigned image set.

Parameters:

nValue The pixel-value to be set.

pDst Destination-Image Pointer.
nDstStep Destination-Image Line Step.
oSizeROI Region-of-Interest (ROI).
pMask Mask-Image Pointer.
nMaskStep Mask-Image Line Step.

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.6.2.20 NppStatus nppiSet_16u_C1R (const Npp16u nValue, Npp16u * pDst, int nDstStep, NppiSize oSizeROI)

16-bit unsigned image set.

Parameters:

nValue The pixel-value to be set.
pDst Destination-Image Pointer.
nDstStep Destination-Image Line Step.
oSizeROI Region-of-Interest (ROI).

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.6.2.21 NppStatus nppiSet_16u_C2R (const Npp16u aValue[2], Npp16u * pDst, int nDstStep, NppiSize oSizeROI)

2 channel 16-bit unsigned image set.

Parameters:

aValue The pixel-value to be set.
pDst Destination-Image Pointer.
nDstStep Destination-Image Line Step.
oSizeROI Region-of-Interest (ROI).

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.6.2.22 NppStatus nppiSet_16u_C3CR (Npp16u nValue, Npp16u * pDst, int nDstStep, NppiSize oSizeROI)

3 channel 16-bit unsigned image set affecting only single channel.

Parameters:

nValue The pixel-value to be set.

pDst Select-Channel Destination-Image Pointer.
nDstStep Destination-Image Line Step.
oSizeROI Region-of-Interest (ROI).

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.6.2.23 NppStatus nppiSet_16u_C3MR (const Npp16u aValue[3], Npp16u * pDst, int nDstStep, NppiSize oSizeROI, const Npp8u * pMask, int nMaskStep)

Masked 3 channel 16-bit unsigned image set.

Parameters:

aValue The pixel-value to be set.
pDst Destination-Image Pointer.
nDstStep Destination-Image Line Step.
oSizeROI Region-of-Interest (ROI).
pMask Mask-Image Pointer.
nMaskStep Mask-Image Line Step.

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.6.2.24 NppStatus nppiSet_16u_C3R (const Npp16u aValue[3], Npp16u * pDst, int nDstStep, NppiSize oSizeROI)

3 channel 16-bit unsigned image set.

Parameters:

aValue The pixel-value to be set.
pDst Destination-Image Pointer.
nDstStep Destination-Image Line Step.
oSizeROI Region-of-Interest (ROI).

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.6.2.25 NppStatus nppiSet_16u_C4CR (Npp16u nValue, Npp16u * pDst, int nDstStep, NppiSize oSizeROI)

4 channel 16-bit unsigned image set affecting only single channel.

Parameters:

nValue The pixel-value to be set.

pDst Select-Channel Destination-Image Pointer.
nDstStep Destination-Image Line Step.
oSizeROI Region-of-Interest (ROI).

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.6.2.26 `NppStatus nppiSet_16u_C4MR (const Npp16u aValue[4], Npp16u * pDst, int nDstStep, NppiSize oSizeROI, const Npp8u * pMask, int nMaskStep)`

Masked 4 channel 16-bit unsigned image set.

Parameters:

aValue The pixel-value to be set.
pDst Destination-Image Pointer.
nDstStep Destination-Image Line Step.
oSizeROI Region-of-Interest (ROI).
pMask Mask-Image Pointer.
nMaskStep Mask-Image Line Step.

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.6.2.27 `NppStatus nppiSet_16u_C4R (const Npp16u aValue[4], Npp16u * pDst, int nDstStep, NppiSize oSizeROI)`

4 channel 16-bit unsigned image set.

Parameters:

aValue The pixel-value to be set.
pDst Destination-Image Pointer.
nDstStep Destination-Image Line Step.
oSizeROI Region-of-Interest (ROI).

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.6.2.28 `NppStatus nppiSet_32f_AC4MR (const Npp32f aValue[3], Npp32f * pDst, int nDstStep, NppiSize oSizeROI, const Npp8u * pMask, int nMaskStep)`

Masked 4 channel 32-bit floating point image set method, not affecting Alpha channel.

Parameters:

aValue The pixel-value to be set.

pDst Destination-Image Pointer.

nDstStep Destination-Image Line Step.

oSizeROI Region-of-Interest (ROI).

pMask Mask-Image Pointer.

nMaskStep Mask-Image Line Step.

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.6.2.29 NppStatus nppiSet_32f_AC4R (const Npp32f aValue[3], Npp32f * pDst, int nDstStep, NppiSize oSizeROI)

4 channel 32-bit floating point image set method, not affecting Alpha channel.

Parameters:

aValue The pixel-value to be set.

pDst Destination-Image Pointer.

nDstStep Destination-Image Line Step.

oSizeROI Region-of-Interest (ROI).

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.6.2.30 NppStatus nppiSet_32f_C1MR (Npp32f nValue, Npp32f * pDst, int nDstStep, NppiSize oSizeROI, const Npp8u * pMask, int nMaskStep)

Masked 32-bit floating point image set.

Parameters:

nValue The pixel-value to be set.

pDst Destination-Image Pointer.

nDstStep Destination-Image Line Step.

oSizeROI Region-of-Interest (ROI).

pMask Mask-Image Pointer.

nMaskStep Mask-Image Line Step.

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.6.2.31 NppStatus nppiSet_32f_C1R (const Npp32f nValue, Npp32f * pDst, int nDstStep, NppiSize oSizeROI)

32-bit floating point image set.

Parameters:

nValue The pixel-value to be set.

pDst Destination-Image Pointer.

nDstStep Destination-Image Line Step.

oSizeROI Region-of-Interest (ROI).

Returns:

[Image Data Related Error Codes](#), [ROI Related Error Codes](#)

7.6.2.32 NppStatus nppiSet_32f_C2R (const Npp32f aValue[2], Npp32f * pDst, int nDstStep, NppiSize oSizeROI)

2 channel 32-bit floating point image set.

Parameters:

aValue The pixel-value to be set.

pDst Destination-Image Pointer.

nDstStep Destination-Image Line Step.

oSizeROI Region-of-Interest (ROI).

Returns:

[Image Data Related Error Codes](#), [ROI Related Error Codes](#)

7.6.2.33 NppStatus nppiSet_32f_C3CR (Npp32f nValue, Npp32f * pDst, int nDstStep, NppiSize oSizeROI)

3 channel 32-bit floating point image set affecting only single channel.

Parameters:

nValue The pixel-value to be set.

pDst Select-Channel Destination-Image Pointer.

nDstStep Destination-Image Line Step.

oSizeROI Region-of-Interest (ROI).

Returns:

[Image Data Related Error Codes](#), [ROI Related Error Codes](#)

7.6.2.34 NppStatus nppiSet_32f_C3MR (const Npp32f *aValue*[3], Npp32f * *pDst*, int *nDstStep*, NppiSize *oSizeROI*, const Npp8u * *pMask*, int *nMaskStep*)

Masked 3 channel 32-bit floating point image set.

Parameters:

aValue The pixel-value to be set.

pDst Destination-Image Pointer.

nDstStep Destination-Image Line Step.

oSizeROI Region-of-Interest (ROI).

pMask Mask-Image Pointer.

nMaskStep Mask-Image Line Step.

Returns:

[Image Data Related Error Codes](#), [ROI Related Error Codes](#)

7.6.2.35 NppStatus nppiSet_32f_C3R (const Npp32f *aValue*[3], Npp32f * *pDst*, int *nDstStep*, NppiSize *oSizeROI*)

3 channel 32-bit floating point image set.

Parameters:

aValue The pixel-value to be set.

pDst Destination-Image Pointer.

nDstStep Destination-Image Line Step.

oSizeROI Region-of-Interest (ROI).

Returns:

[Image Data Related Error Codes](#), [ROI Related Error Codes](#)

7.6.2.36 NppStatus nppiSet_32f_C4CR (Npp32f *nValue*, Npp32f * *pDst*, int *nDstStep*, NppiSize *oSizeROI*)

4 channel 32-bit floating point image set affecting only single channel.

Parameters:

nValue The pixel-value to be set.

pDst Select-Channel Destination-Image Pointer.

nDstStep Destination-Image Line Step.

oSizeROI Region-of-Interest (ROI).

Returns:

[Image Data Related Error Codes](#), [ROI Related Error Codes](#)

7.6.2.37 NppStatus nppiSet_32f_C4MR (const Npp32f aValue[4], Npp32f * pDst, int nDstStep, NppiSize oSizeROI, const Npp8u * pMask, int nMaskStep)

Masked 4 channel 32-bit floating point image set.

Parameters:

aValue The pixel-value to be set.

pDst Destination-Image Pointer.

nDstStep Destination-Image Line Step.

oSizeROI Region-of-Interest (ROI).

pMask Mask-Image Pointer.

nMaskStep Mask-Image Line Step.

Returns:

[Image Data Related Error Codes](#), [ROI Related Error Codes](#)

7.6.2.38 NppStatus nppiSet_32f_C4R (const Npp32f aValue[4], Npp32f * pDst, int nDstStep, NppiSize oSizeROI)

4 channel 32-bit floating point image set.

Parameters:

aValue The pixel-value to be set.

pDst Destination-Image Pointer.

nDstStep Destination-Image Line Step.

oSizeROI Region-of-Interest (ROI).

Returns:

[Image Data Related Error Codes](#), [ROI Related Error Codes](#)

7.6.2.39 NppStatus nppiSet_32fc_AC4R (const Npp32fc aValue[3], Npp32fc * pDst, int nDstStep, NppiSize oSizeROI)

32-bit complex four-channel image set ignoring alpha.

Parameters:

aValue The pixel-value to be set.

pDst Destination-Image Pointer.

nDstStep Destination-Image Line Step.

oSizeROI Region-of-Interest (ROI).

Returns:

[Image Data Related Error Codes](#), [ROI Related Error Codes](#)

7.6.2.40 NppStatus nppiSet_32fc_C1R (const Npp32fc oValue, Npp32fc * pDst, int nDstStep, NppiSize oSizeROI)

Single channel 32-bit complex image set.

Parameters:

oValue The pixel-value to be set.

pDst Destination-Image Pointer.

nDstStep Destination-Image Line Step.

oSizeROI Region-of-Interest (ROI).

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.6.2.41 NppStatus nppiSet_32fc_C2R (const Npp32fc aValue[2], Npp32fc * pDst, int nDstStep, NppiSize oSizeROI)

Two channel 32-bit complex image set.

Parameters:

aValue The pixel-value to be set.

pDst Destination-Image Pointer.

nDstStep Destination-Image Line Step.

oSizeROI Region-of-Interest (ROI).

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.6.2.42 NppStatus nppiSet_32fc_C3R (const Npp32fc aValue[3], Npp32fc * pDst, int nDstStep, NppiSize oSizeROI)

Three channel 32-bit complex image set.

Parameters:

aValue The pixel-value to be set.

pDst Destination-Image Pointer.

nDstStep Destination-Image Line Step.

oSizeROI Region-of-Interest (ROI).

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.6.2.43 **NppStatus nppiSet_32fc_C4R** (const Npp32fc *aValue*[4], Npp32fc * *pDst*, int *nDstStep*, NppiSize *oSizeROI*)

Four channel 32-bit complex image set.

Parameters:

aValue The pixel-value to be set.

pDst Destination-Image Pointer.

nDstStep Destination-Image Line Step.

oSizeROI Region-of-Interest (ROI).

Returns:

[Image Data Related Error Codes](#), [ROI Related Error Codes](#)

7.6.2.44 **NppStatus nppiSet_32s_AC4MR** (const Npp32s *aValue*[3], Npp32s * *pDst*, int *nDstStep*, NppiSize *oSizeROI*, const Npp8u * *pMask*, int *nMaskStep*)

Masked 4 channel 16-bit image set method, not affecting Alpha channel.

Parameters:

aValue The pixel-value to be set.

pDst Destination-Image Pointer.

nDstStep Destination-Image Line Step.

oSizeROI Region-of-Interest (ROI).

pMask Mask-Image Pointer.

nMaskStep Mask-Image Line Step.

Returns:

[Image Data Related Error Codes](#), [ROI Related Error Codes](#)

7.6.2.45 **NppStatus nppiSet_32s_AC4R** (const Npp32s *aValue*[3], Npp32s * *pDst*, int *nDstStep*, NppiSize *oSizeROI*)

4 channel 32-bit image set method, not affecting Alpha channel.

Parameters:

aValue The pixel-value to be set.

pDst Destination-Image Pointer.

nDstStep Destination-Image Line Step.

oSizeROI Region-of-Interest (ROI).

Returns:

[Image Data Related Error Codes](#), [ROI Related Error Codes](#)

7.6.2.46 NppStatus nppiSet_32s_C1MR (Npp32s nValue, Npp32s * pDst, int nDstStep, NppiSize oSizeROI, const Npp8u * pMask, int nMaskStep)

Masked 32-bit image set.

Parameters:

nValue The pixel-value to be set.

pDst Destination-Image Pointer.

nDstStep Destination-Image Line Step.

oSizeROI Region-of-Interest (ROI).

pMask Mask-Image Pointer.

nMaskStep Mask-Image Line Step.

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.6.2.47 NppStatus nppiSet_32s_C1R (const Npp32s nValue, Npp32s * pDst, int nDstStep, NppiSize oSizeROI)

32-bit image set.

Parameters:

nValue The pixel-value to be set.

pDst Destination-Image Pointer.

nDstStep Destination-Image Line Step.

oSizeROI Region-of-Interest (ROI).

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.6.2.48 NppStatus nppiSet_32s_C2R (const Npp32s aValue[2], Npp32s * pDst, int nDstStep, NppiSize oSizeROI)

2 channel 32-bit image set.

Parameters:

aValue The pixel-value to be set.

pDst Destination-Image Pointer.

nDstStep Destination-Image Line Step.

oSizeROI Region-of-Interest (ROI).

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.6.2.49 NppStatus nppiSet_32s_C3CR (Npp32s *nValue*, Npp32s * *pDst*, int *nDstStep*, NppiSize *oSizeROI*)

3 channel 32-bit unsigned image set affecting only single channel.

Parameters:

- nValue* The pixel-value to be set.
- pDst* [Select-Channel Destination-Image Pointer](#).
- nDstStep* [Destination-Image Line Step](#).
- oSizeROI* [Region-of-Interest \(ROI\)](#).

Returns:

[Image Data Related Error Codes](#), [ROI Related Error Codes](#)

7.6.2.50 NppStatus nppiSet_32s_C3MR (const Npp32s *aValue*[3], Npp32s * *pDst*, int *nDstStep*, NppiSize *oSizeROI*, const Npp8u * *pMask*, int *nMaskStep*)

Masked 3 channel 32-bit image set.

Parameters:

- aValue* The pixel-value to be set.
- pDst* [Destination-Image Pointer](#).
- nDstStep* [Destination-Image Line Step](#).
- oSizeROI* [Region-of-Interest \(ROI\)](#).
- pMask* [Mask-Image Pointer](#).
- nMaskStep* [Mask-Image Line Step](#).

Returns:

[Image Data Related Error Codes](#), [ROI Related Error Codes](#)

7.6.2.51 NppStatus nppiSet_32s_C3R (const Npp32s *aValue*[3], Npp32s * *pDst*, int *nDstStep*, NppiSize *oSizeROI*)

3 channel 32-bit image set.

Parameters:

- aValue* The pixel-value to be set.
- pDst* [Destination-Image Pointer](#).
- nDstStep* [Destination-Image Line Step](#).
- oSizeROI* [Region-of-Interest \(ROI\)](#).

Returns:

[Image Data Related Error Codes](#), [ROI Related Error Codes](#)

7.6.2.52 NppStatus nppiSet_32s_C4CR (Npp32s *nValue*, Npp32s * *pDst*, int *nDstStep*, NppiSize *oSizeROI*)

4 channel 32-bit unsigned image set affecting only single channel.

Parameters:

- nValue* The pixel-value to be set.
- pDst* [Select-Channel Destination-Image Pointer](#).
- nDstStep* [Destination-Image Line Step](#).
- oSizeROI* [Region-of-Interest \(ROI\)](#).

Returns:

[Image Data Related Error Codes](#), [ROI Related Error Codes](#)

7.6.2.53 NppStatus nppiSet_32s_C4MR (const Npp32s *aValue*[4], Npp32s * *pDst*, int *nDstStep*, NppiSize *oSizeROI*, const Npp8u * *pMask*, int *nMaskStep*)

Masked 4 channel 32-bit image set.

Parameters:

- aValue* The pixel-value to be set.
- pDst* [Destination-Image Pointer](#).
- nDstStep* [Destination-Image Line Step](#).
- oSizeROI* [Region-of-Interest \(ROI\)](#).
- pMask* [Mask-Image Pointer](#).
- nMaskStep* [Mask-Image Line Step](#).

Returns:

[Image Data Related Error Codes](#), [ROI Related Error Codes](#)

7.6.2.54 NppStatus nppiSet_32s_C4R (const Npp32s *aValue*[4], Npp32s * *pDst*, int *nDstStep*, NppiSize *oSizeROI*)

4 channel 32-bit image set.

Parameters:

- aValue* The pixel-value to be set.
- pDst* [Destination-Image Pointer](#).
- nDstStep* [Destination-Image Line Step](#).
- oSizeROI* [Region-of-Interest \(ROI\)](#).

Returns:

[Image Data Related Error Codes](#), [ROI Related Error Codes](#)

7.6.2.55 NppStatus nppiSet_32sc_AC4R (const Npp32sc *aValue*[3], Npp32sc * *pDst*, int *nDstStep*, NppiSize *oSizeROI*)

32-bit complex integer four-channel image set ignoring alpha.

Parameters:

aValue The pixel-value to be set.

pDst Destination-Image Pointer.

nDstStep Destination-Image Line Step.

oSizeROI Region-of-Interest (ROI).

Returns:

[Image Data Related Error Codes](#), [ROI Related Error Codes](#)

7.6.2.56 NppStatus nppiSet_32sc_C1R (const Npp32sc *oValue*, Npp32sc * *pDst*, int *nDstStep*, NppiSize *oSizeROI*)

Single channel 32-bit complex integer image set.

Parameters:

oValue The pixel-value to be set.

pDst Destination-Image Pointer.

nDstStep Destination-Image Line Step.

oSizeROI Region-of-Interest (ROI).

Returns:

[Image Data Related Error Codes](#), [ROI Related Error Codes](#)

7.6.2.57 NppStatus nppiSet_32sc_C2R (const Npp32sc *aValue*[2], Npp32sc * *pDst*, int *nDstStep*, NppiSize *oSizeROI*)

Two channel 32-bit complex integer image set.

Parameters:

aValue The pixel-value to be set.

pDst Destination-Image Pointer.

nDstStep Destination-Image Line Step.

oSizeROI Region-of-Interest (ROI).

Returns:

[Image Data Related Error Codes](#), [ROI Related Error Codes](#)

7.6.2.58 NppStatus nppiSet_32sc_C3R (const Npp32sc aValue[3], Npp32sc * pDst, int nDstStep, NppiSize oSizeROI)

Three channel 32-bit complex integer image set.

Parameters:

aValue The pixel-value to be set.

pDst Destination-Image Pointer.

nDstStep Destination-Image Line Step.

oSizeROI Region-of-Interest (ROI).

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.6.2.59 NppStatus nppiSet_32sc_C4R (const Npp32sc aValue[4], Npp32sc * pDst, int nDstStep, NppiSize oSizeROI)

Four channel 32-bit complex integer image set.

Parameters:

aValue The pixel-value to be set.

pDst Destination-Image Pointer.

nDstStep Destination-Image Line Step.

oSizeROI Region-of-Interest (ROI).

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.6.2.60 NppStatus nppiSet_32u_AC4R (const Npp32u aValue[3], Npp32u * pDst, int nDstStep, NppiSize oSizeROI)

4 channel 32-bit unsigned image set method, not affecting Alpha channel.

Parameters:

aValue The pixel-value to be set.

pDst Destination-Image Pointer.

nDstStep Destination-Image Line Step.

oSizeROI Region-of-Interest (ROI).

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.6.2.61 NppStatus nppiSet_32u_C1R (const Npp32u *nValue*, Npp32u * *pDst*, int *nDstStep*, NppiSize *oSizeROI*)

32-bit unsigned image set.

Parameters:

nValue The pixel-value to be set.

pDst Destination-Image Pointer.

nDstStep Destination-Image Line Step.

oSizeROI Region-of-Interest (ROI).

Returns:

[Image Data Related Error Codes](#), [ROI Related Error Codes](#)

7.6.2.62 NppStatus nppiSet_32u_C2R (const Npp32u *aValue*[2], Npp32u * *pDst*, int *nDstStep*, NppiSize *oSizeROI*)

2 channel 32-bit unsigned image set.

Parameters:

aValue The pixel-value to be set.

pDst Destination-Image Pointer.

nDstStep Destination-Image Line Step.

oSizeROI Region-of-Interest (ROI).

Returns:

[Image Data Related Error Codes](#), [ROI Related Error Codes](#)

7.6.2.63 NppStatus nppiSet_32u_C3R (const Npp32u *aValue*[3], Npp32u * *pDst*, int *nDstStep*, NppiSize *oSizeROI*)

3 channel 32-bit unsigned image set.

Parameters:

aValue The pixel-value to be set.

pDst Destination-Image Pointer.

nDstStep Destination-Image Line Step.

oSizeROI Region-of-Interest (ROI).

Returns:

[Image Data Related Error Codes](#), [ROI Related Error Codes](#)

7.6.2.64 NppStatus nppiSet_32u_C4R (const Npp32u aValue[4], Npp32u * pDst, int nDstStep, NppiSize oSizeROI)

4 channel 32-bit unsigned image set.

Parameters:

aValue The pixel-value to be set.

pDst Destination-Image Pointer.

nDstStep Destination-Image Line Step.

oSizeROI Region-of-Interest (ROI).

Returns:

[Image Data Related Error Codes](#), [ROI Related Error Codes](#)

7.6.2.65 NppStatus nppiSet_8s_AC4R (const Npp8s aValue[3], Npp8s * pDst, int nDstStep, NppiSize oSizeROI)

8-bit four-channel image set ignoring alpha channel.

Parameters:

aValue The pixel value to be set.

pDst Destination-Image Pointer.

nDstStep Destination-Image Line Step.

oSizeROI Region-of-Interest (ROI).

Returns:

[Image Data Related Error Codes](#), [ROI Related Error Codes](#)

7.6.2.66 NppStatus nppiSet_8s_C1R (const Npp8s nValue, Npp8s * pDst, int nDstStep, NppiSize oSizeROI)

8-bit image set.

Parameters:

nValue The pixel value to be set.

pDst Destination-Image Pointer.

nDstStep Destination-Image Line Step.

oSizeROI Region-of-Interest (ROI).

Returns:

[Image Data Related Error Codes](#), [ROI Related Error Codes](#)

7.6.2.67 NppStatus nppiSet_8s_C2R (const Npp8s *aValue*[2], Npp8s * *pDst*, int *nDstStep*, NppiSize *oSizeROI*)

8-bit two-channel image set.

Parameters:

aValue The pixel value to be set.

pDst Destination-Image Pointer.

nDstStep Destination-Image Line Step.

oSizeROI Region-of-Interest (ROI).

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.6.2.68 NppStatus nppiSet_8s_C3R (const Npp8s *aValue*[3], Npp8s * *pDst*, int *nDstStep*, NppiSize *oSizeROI*)

8-bit three-channel image set.

Parameters:

aValue The pixel value to be set.

pDst Destination-Image Pointer.

nDstStep Destination-Image Line Step.

oSizeROI Region-of-Interest (ROI).

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.6.2.69 NppStatus nppiSet_8s_C4R (const Npp8s *aValue*[4], Npp8s * *pDst*, int *nDstStep*, NppiSize *oSizeROI*)

8-bit four-channel image set.

Parameters:

aValue The pixel value to be set.

pDst Destination-Image Pointer.

nDstStep Destination-Image Line Step.

oSizeROI Region-of-Interest (ROI).

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.6.2.70 NppStatus nppiSet_8u_AC4MR (const Npp8u aValue[3], Npp8u * pDst, int nDstStep, NppiSize oSizeROI, const Npp8u * pMask, int nMaskStep)

Masked 4 channel 8-bit unsigned image set method, not affecting Alpha channel.

Parameters:

aValue The pixel-value to be set.
pDst Destination-Image Pointer.
nDstStep Destination-Image Line Step.
oSizeROI Region-of-Interest (ROI).
pMask Mask-Image Pointer.
nMaskStep Mask-Image Line Step.

Returns:

[Image Data Related Error Codes](#), [ROI Related Error Codes](#)

7.6.2.71 NppStatus nppiSet_8u_AC4R (const Npp8u aValue[3], Npp8u * pDst, int nDstStep, NppiSize oSizeROI)

4 channel 8-bit unsigned image set method, not affecting Alpha channel.

Parameters:

aValue The pixel-value to be set.
pDst Destination-Image Pointer.
nDstStep Destination-Image Line Step.
oSizeROI Region-of-Interest (ROI).

Returns:

[Image Data Related Error Codes](#), [ROI Related Error Codes](#)

7.6.2.72 NppStatus nppiSet_8u_C1MR (Npp8u nValue, Npp8u * pDst, int nDstStep, NppiSize oSizeROI, const Npp8u * pMask, int nMaskStep)

Masked 8-bit unsigned image set.

Parameters:

nValue The pixel value to be set.
pDst Pointer Destination-Image Pointer.
nDstStep Destination-Image Line Step.
oSizeROI Region-of-Interest (ROI).
pMask Mask-Image Pointer.
nMaskStep Mask-Image Line Step.

Returns:

[Image Data Related Error Codes](#), [ROI Related Error Codes](#)

7.6.2.73 NppStatus nppiSet_8u_C1R (const Npp8u *nValue*, Npp8u * *pDst*, int *nDstStep*, NppiSize *oSizeROI*)

8-bit unsigned image set.

Parameters:

nValue The pixel value to be set.

pDst Destination-Image Pointer.

nDstStep Destination-Image Line Step.

oSizeROI Region-of-Interest (ROI).

Returns:

[Image Data Related Error Codes](#), [ROI Related Error Codes](#)

7.6.2.74 NppStatus nppiSet_8u_C2R (const Npp8u *aValue*[2], Npp8u * *pDst*, int *nDstStep*, NppiSize *oSizeROI*)

2 channel 8-bit unsigned image set.

Parameters:

aValue The pixel value to be set.

pDst Destination-Image Pointer.

nDstStep Destination-Image Line Step.

oSizeROI Region-of-Interest (ROI).

Returns:

[Image Data Related Error Codes](#), [ROI Related Error Codes](#)

7.6.2.75 NppStatus nppiSet_8u_C3CR (Npp8u *nValue*, Npp8u * *pDst*, int *nDstStep*, NppiSize *oSizeROI*)

3 channel 8-bit unsigned image set affecting only single channel.

Parameters:

nValue The pixel-value to be set.

pDst Select-Channel Destination-Image Pointer.

nDstStep Destination-Image Line Step.

oSizeROI Region-of-Interest (ROI).

Returns:

[Image Data Related Error Codes](#), [ROI Related Error Codes](#)

7.6.2.76 NppStatus nppiSet_8u_C3MR (const Npp8u aValue[3], Npp8u * pDst, int nDstStep, NppiSize oSizeROI, const Npp8u * pMask, int nMaskStep)

Masked 3 channel 8-bit unsigned image set.

Parameters:

aValue The pixel-value to be set.

pDst Destination-Image Pointer.

nDstStep Destination-Image Line Step.

oSizeROI Region-of-Interest (ROI).

pMask Mask-Image Pointer.

nMaskStep Mask-Image Line Step.

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.6.2.77 NppStatus nppiSet_8u_C3R (const Npp8u aValue[3], Npp8u * pDst, int nDstStep, NppiSize oSizeROI)

3 channel 8-bit unsigned image set.

Parameters:

aValue The pixel value to be set.

pDst Destination-Image Pointer.

nDstStep Destination-Image Line Step.

oSizeROI Region-of-Interest (ROI).

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.6.2.78 NppStatus nppiSet_8u_C4CR (Npp8u nValue, Npp8u * pDst, int nDstStep, NppiSize oSizeROI)

4 channel 8-bit unsigned image set affecting only single channel.

Parameters:

nValue The pixel-value to be set.

pDst Select-Channel Destination-Image Pointer.

nDstStep Destination-Image Line Step.

oSizeROI Region-of-Interest (ROI).

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.6.2.79 NppStatus nppiSet_8u_C4MR (const Npp8u aValue[4], Npp8u * pDst, int nDstStep, NppiSize oSizeROI, const Npp8u * pMask, int nMaskStep)

Masked 4 channel 8-bit unsigned image set.

Parameters:

aValue The pixel-value to be set.

pDst Destination-Image Pointer.

nDstStep Destination-Image Line Step.

oSizeROI Region-of-Interest (ROI).

pMask Mask-Image Pointer.

nMaskStep Mask-Image Line Step.

Returns:

[Image Data Related Error Codes](#), [ROI Related Error Codes](#)

7.6.2.80 NppStatus nppiSet_8u_C4R (const Npp8u aValue[4], Npp8u * pDst, int nDstStep, NppiSize oSizeROI)

4 channel 8-bit unsigned image set.

Parameters:

aValue The pixel-value to be set.

pDst Destination-Image Pointer.

nDstStep Destination-Image Line Step.

oSizeROI Region-of-Interest (ROI).

Returns:

[Image Data Related Error Codes](#), [ROI Related Error Codes](#)

7.7 Copy

Copy

Copy pixels from one image to another.

- **NppStatus nppiCopy_8s_C1R** (const **Npp8s** *pSrc, int nSrcStep, **Npp8s** *pDst, int nDstStep, **NppiSize** oSizeROI)
8-bit image copy.
- **NppStatus nppiCopy_8s_C2R** (const **Npp8s** *pSrc, int nSrcStep, **Npp8s** *pDst, int nDstStep, **NppiSize** oSizeROI)
Two-channel 8-bit image copy.
- **NppStatus nppiCopy_8s_C3R** (const **Npp8s** *pSrc, int nSrcStep, **Npp8s** *pDst, int nDstStep, **NppiSize** oSizeROI)
Three-channel 8-bit image copy.
- **NppStatus nppiCopy_8s_C4R** (const **Npp8s** *pSrc, int nSrcStep, **Npp8s** *pDst, int nDstStep, **NppiSize** oSizeROI)
Four-channel 8-bit image copy.
- **NppStatus nppiCopy_8s_AC4R** (const **Npp8s** *pSrc, int nSrcStep, **Npp8s** *pDst, int nDstStep, **NppiSize** oSizeROI)
Four-channel 8-bit image copy, ignoring alpha channel.
- **NppStatus nppiCopy_8u_C1R** (const **Npp8u** *pSrc, int nSrcStep, **Npp8u** *pDst, int nDstStep, **NppiSize** oSizeROI)
8-bit unsigned image copy.
- **NppStatus nppiCopy_8u_C3R** (const **Npp8u** *pSrc, int nSrcStep, **Npp8u** *pDst, int nDstStep, **NppiSize** oSizeROI)
Three channel 8-bit unsigned image copy.
- **NppStatus nppiCopy_8u_C4R** (const **Npp8u** *pSrc, int nSrcStep, **Npp8u** *pDst, int nDstStep, **NppiSize** oSizeROI)
4 channel 8-bit unsigned image copy.
- **NppStatus nppiCopy_8u_AC4R** (const **Npp8u** *pSrc, int nSrcStep, **Npp8u** *pDst, int nDstStep, **NppiSize** oSizeROI)
4 channel 8-bit unsigned image copy, not affecting Alpha channel.
- **NppStatus nppiCopy_16u_C1R** (const **Npp16u** *pSrc, int nSrcStep, **Npp16u** *pDst, int nDstStep, **NppiSize** oSizeROI)
16-bit unsigned image copy.
- **NppStatus nppiCopy_16u_C3R** (const **Npp16u** *pSrc, int nSrcStep, **Npp16u** *pDst, int nDstStep, **NppiSize** oSizeROI)
Three channel 16-bit unsigned image copy.

- `NppStatus nppiCopy_16u_C4R` (const `Npp16u *pSrc`, int `nSrcStep`, `Npp16u *pDst`, int `nDstStep`, `NppiSize oSizeROI`)
4 channel 16-bit unsigned image copy.
- `NppStatus nppiCopy_16u_AC4R` (const `Npp16u *pSrc`, int `nSrcStep`, `Npp16u *pDst`, int `nDstStep`, `NppiSize oSizeROI`)
4 channel 16-bit unsigned image copy, not affecting Alpha channel.
- `NppStatus nppiCopy_16s_C1R` (const `Npp16s *pSrc`, int `nSrcStep`, `Npp16s *pDst`, int `nDstStep`, `NppiSize oSizeROI`)
16-bit image copy.
- `NppStatus nppiCopy_16s_C3R` (const `Npp16s *pSrc`, int `nSrcStep`, `Npp16s *pDst`, int `nDstStep`, `NppiSize oSizeROI`)
Three channel 16-bit image copy.
- `NppStatus nppiCopy_16s_C4R` (const `Npp16s *pSrc`, int `nSrcStep`, `Npp16s *pDst`, int `nDstStep`, `NppiSize oSizeROI`)
4 channel 16-bit image copy.
- `NppStatus nppiCopy_16s_AC4R` (const `Npp16s *pSrc`, int `nSrcStep`, `Npp16s *pDst`, int `nDstStep`, `NppiSize oSizeROI`)
4 channel 16-bit image copy, not affecting Alpha.
- `NppStatus nppiCopy_16sc_C1R` (const `Npp16sc *pSrc`, int `nSrcStep`, `Npp16sc *pDst`, int `nDstStep`, `NppiSize oSizeROI`)
16-bit complex image copy.
- `NppStatus nppiCopy_16sc_C2R` (const `Npp16sc *pSrc`, int `nSrcStep`, `Npp16sc *pDst`, int `nDstStep`, `NppiSize oSizeROI`)
Two-channel 16-bit complex image copy.
- `NppStatus nppiCopy_16sc_C3R` (const `Npp16sc *pSrc`, int `nSrcStep`, `Npp16sc *pDst`, int `nDstStep`, `NppiSize oSizeROI`)
Three-channel 16-bit complex image copy.
- `NppStatus nppiCopy_16sc_C4R` (const `Npp16sc *pSrc`, int `nSrcStep`, `Npp16sc *pDst`, int `nDstStep`, `NppiSize oSizeROI`)
Four-channel 16-bit complex image copy.
- `NppStatus nppiCopy_16sc_AC4R` (const `Npp16sc *pSrc`, int `nSrcStep`, `Npp16sc *pDst`, int `nDstStep`, `NppiSize oSizeROI`)
Four-channel 16-bit complex image copy, ignoring alpha.
- `NppStatus nppiCopy_32s_C1R` (const `Npp32s *pSrc`, int `nSrcStep`, `Npp32s *pDst`, int `nDstStep`, `NppiSize oSizeROI`)
32-bit image copy.
- `NppStatus nppiCopy_32s_C3R` (const `Npp32s *pSrc`, int `nSrcStep`, `Npp32s *pDst`, int `nDstStep`, `NppiSize oSizeROI`)
Three channel 32-bit image copy.

- `NppStatus nppiCopy_32s_C4R` (const `Npp32s` *pSrc, int nSrcStep, `Npp32s` *pDst, int nDstStep, `NppiSize` oSizeROI)
4 channel 32-bit image copy.
- `NppStatus nppiCopy_32s_AC4R` (const `Npp32s` *pSrc, int nSrcStep, `Npp32s` *pDst, int nDstStep, `NppiSize` oSizeROI)
4 channel 32-bit image copy, not affecting Alpha.
- `NppStatus nppiCopy_32sc_C1R` (const `Npp32sc` *pSrc, int nSrcStep, `Npp32sc` *pDst, int nDstStep, `NppiSize` oSizeROI)
32-bit complex image copy.
- `NppStatus nppiCopy_32sc_C2R` (const `Npp32sc` *pSrc, int nSrcStep, `Npp32sc` *pDst, int nDstStep, `NppiSize` oSizeROI)
Two-channel 32-bit complex image copy.
- `NppStatus nppiCopy_32sc_C3R` (const `Npp32sc` *pSrc, int nSrcStep, `Npp32sc` *pDst, int nDstStep, `NppiSize` oSizeROI)
Three-channel 32-bit complex image copy.
- `NppStatus nppiCopy_32sc_C4R` (const `Npp32sc` *pSrc, int nSrcStep, `Npp32sc` *pDst, int nDstStep, `NppiSize` oSizeROI)
Four-channel 32-bit complex image copy.
- `NppStatus nppiCopy_32sc_AC4R` (const `Npp32sc` *pSrc, int nSrcStep, `Npp32sc` *pDst, int nDstStep, `NppiSize` oSizeROI)
Four-channel 32-bit complex image copy, ignoring alpha.
- `NppStatus nppiCopy_32f_C1R` (const `Npp32f` *pSrc, int nSrcStep, `Npp32f` *pDst, int nDstStep, `NppiSize` oSizeROI)
32-bit floating point image copy.
- `NppStatus nppiCopy_32f_C3R` (const `Npp32f` *pSrc, int nSrcStep, `Npp32f` *pDst, int nDstStep, `NppiSize` oSizeROI)
Three channel 32-bit floating point image copy.
- `NppStatus nppiCopy_32f_C4R` (const `Npp32f` *pSrc, int nSrcStep, `Npp32f` *pDst, int nDstStep, `NppiSize` oSizeROI)
4 channel 32-bit floating point image copy.
- `NppStatus nppiCopy_32f_AC4R` (const `Npp32f` *pSrc, int nSrcStep, `Npp32f` *pDst, int nDstStep, `NppiSize` oSizeROI)
4 channel 32-bit floating point image copy, not affecting Alpha.
- `NppStatus nppiCopy_32fc_C1R` (const `Npp32fc` *pSrc, int nSrcStep, `Npp32fc` *pDst, int nDstStep, `NppiSize` oSizeROI)
32-bit floating-point complex image copy.
- `NppStatus nppiCopy_32fc_C2R` (const `Npp32fc` *pSrc, int nSrcStep, `Npp32fc` *pDst, int nDstStep, `NppiSize` oSizeROI)

Two-channel 32-bit floating-point complex image copy.

- `NppStatus nppiCopy_32fc_C3R` (const `Npp32fc` *pSrc, int nSrcStep, `Npp32fc` *pDst, int nDstStep, `NppiSize` oSizeROI)

Three-channel 32-bit floating-point complex image copy.

- `NppStatus nppiCopy_32fc_C4R` (const `Npp32fc` *pSrc, int nSrcStep, `Npp32fc` *pDst, int nDstStep, `NppiSize` oSizeROI)

Four-channel 32-bit floating-point complex image copy.

- `NppStatus nppiCopy_32fc_AC4R` (const `Npp32fc` *pSrc, int nSrcStep, `Npp32fc` *pDst, int nDstStep, `NppiSize` oSizeROI)

Four-channel 32-bit floating-point complex image copy, ignoring alpha.

Masked Copy

The masked copy primitives have an additional "mask image" input.

The mask controls which pixels within the ROI are copied. For details see [Masked Operation](#).

- `NppStatus nppiCopy_8u_C1MR` (const `Npp8u` *pSrc, int nSrcStep, `Npp8u` *pDst, int nDstStep, `NppiSize` oSizeROI, const `Npp8u` *pMask, int nMaskStep)

Masked Operation 8-bit unsigned image copy.

- `NppStatus nppiCopy_8u_C3MR` (const `Npp8u` *pSrc, int nSrcStep, `Npp8u` *pDst, int nDstStep, `NppiSize` oSizeROI, const `Npp8u` *pMask, int nMaskStep)

Masked Operation three channel 8-bit unsigned image copy.

- `NppStatus nppiCopy_8u_C4MR` (const `Npp8u` *pSrc, int nSrcStep, `Npp8u` *pDst, int nDstStep, `NppiSize` oSizeROI, const `Npp8u` *pMask, int nMaskStep)

Masked Operation four channel 8-bit unsigned image copy.

- `NppStatus nppiCopy_8u_AC4MR` (const `Npp8u` *pSrc, int nSrcStep, `Npp8u` *pDst, int nDstStep, `NppiSize` oSizeROI, const `Npp8u` *pMask, int nMaskStep)

Masked Operation four channel 8-bit unsigned image copy, ignoring alpha.

- `NppStatus nppiCopy_16u_C1MR` (const `Npp16u` *pSrc, int nSrcStep, `Npp16u` *pDst, int nDstStep, `NppiSize` oSizeROI, const `Npp8u` *pMask, int nMaskStep)

Masked Operation 16-bit unsigned image copy.

- `NppStatus nppiCopy_16u_C3MR` (const `Npp16u` *pSrc, int nSrcStep, `Npp16u` *pDst, int nDstStep, `NppiSize` oSizeROI, const `Npp8u` *pMask, int nMaskStep)

Masked Operation three channel 16-bit unsigned image copy.

- `NppStatus nppiCopy_16u_C4MR` (const `Npp16u` *pSrc, int nSrcStep, `Npp16u` *pDst, int nDstStep, `NppiSize` oSizeROI, const `Npp8u` *pMask, int nMaskStep)

Masked Operation four channel 16-bit unsigned image copy.

- `NppStatus nppiCopy_16u_AC4MR` (const `Npp16u` *pSrc, int nSrcStep, `Npp16u` *pDst, int nDstStep, `NppiSize` oSizeROI, const `Npp8u` *pMask, int nMaskStep)

Masked Operation four channel 16-bit unsigned image copy, ignoring alpha.

- `NppStatus nppiCopy_16s_C1MR` (const `Npp16s` *pSrc, int nSrcStep, `Npp16s` *pDst, int nDstStep, `NppiSize` oSizeROI, const `Npp8u` *pMask, int nMaskStep)

Masked Operation 16-bit signed image copy.

- `NppStatus nppiCopy_16s_C3MR` (const `Npp16s` *pSrc, int nSrcStep, `Npp16s` *pDst, int nDstStep, `NppiSize` oSizeROI, const `Npp8u` *pMask, int nMaskStep)

Masked Operation three channel 16-bit signed image copy.

- `NppStatus nppiCopy_16s_C4MR` (const `Npp16s` *pSrc, int nSrcStep, `Npp16s` *pDst, int nDstStep, `NppiSize` oSizeROI, const `Npp8u` *pMask, int nMaskStep)

Masked Operation four channel 16-bit signed image copy.

- `NppStatus nppiCopy_16s_AC4MR` (const `Npp16s` *pSrc, int nSrcStep, `Npp16s` *pDst, int nDstStep, `NppiSize` oSizeROI, const `Npp8u` *pMask, int nMaskStep)

Masked Operation four channel 16-bit signed image copy, ignoring alpha.

- `NppStatus nppiCopy_32s_C1MR` (const `Npp32s` *pSrc, int nSrcStep, `Npp32s` *pDst, int nDstStep, `NppiSize` oSizeROI, const `Npp8u` *pMask, int nMaskStep)

Masked Operation 32-bit signed image copy.

- `NppStatus nppiCopy_32s_C3MR` (const `Npp32s` *pSrc, int nSrcStep, `Npp32s` *pDst, int nDstStep, `NppiSize` oSizeROI, const `Npp8u` *pMask, int nMaskStep)

Masked Operation three channel 32-bit signed image copy.

- `NppStatus nppiCopy_32s_C4MR` (const `Npp32s` *pSrc, int nSrcStep, `Npp32s` *pDst, int nDstStep, `NppiSize` oSizeROI, const `Npp8u` *pMask, int nMaskStep)

Masked Operation four channel 32-bit signed image copy.

- `NppStatus nppiCopy_32s_AC4MR` (const `Npp32s` *pSrc, int nSrcStep, `Npp32s` *pDst, int nDstStep, `NppiSize` oSizeROI, const `Npp8u` *pMask, int nMaskStep)

Masked Operation four channel 32-bit signed image copy, ignoring alpha.

- `NppStatus nppiCopy_32f_C1MR` (const `Npp32f` *pSrc, int nSrcStep, `Npp32f` *pDst, int nDstStep, `NppiSize` oSizeROI, const `Npp8u` *pMask, int nMaskStep)

Masked Operation 32-bit float image copy.

- `NppStatus nppiCopy_32f_C3MR` (const `Npp32f` *pSrc, int nSrcStep, `Npp32f` *pDst, int nDstStep, `NppiSize` oSizeROI, const `Npp8u` *pMask, int nMaskStep)

Masked Operation three channel 32-bit float image copy.

- `NppStatus nppiCopy_32f_C4MR` (const `Npp32f` *pSrc, int nSrcStep, `Npp32f` *pDst, int nDstStep, `NppiSize` oSizeROI, const `Npp8u` *pMask, int nMaskStep)

Masked Operation four channel 32-bit float image copy.

- `NppStatus nppiCopy_32f_AC4MR` (const `Npp32f` *pSrc, int nSrcStep, `Npp32f` *pDst, int nDstStep, `NppiSize` oSizeROI, const `Npp8u` *pMask, int nMaskStep)

Masked Operation four channel 32-bit float image copy, ignoring alpha.

Channel Copy

The channel copy primitives copy a single color channel from a multi-channel source image to any other color channel in a multi-channel destination image.

The channel is selected by adjusting the respective image pointers to point to the desired color channel (see [Channel-of-Interest API](#)).

- `NppStatus nppiCopy_8u_C3CR` (const `Npp8u *pSrc`, int `nSrcStep`, `Npp8u *pDst`, int `nDstStep`, `NppiSize oSizeROI`)

Select-channel 8-bit unsigned image copy for three-channel images.

- `NppStatus nppiCopy_8u_C4CR` (const `Npp8u *pSrc`, int `nSrcStep`, `Npp8u *pDst`, int `nDstStep`, `NppiSize oSizeROI`)

Select-channel 8-bit unsigned image copy for four-channel images.

- `NppStatus nppiCopy_16s_C3CR` (const `Npp16s *pSrc`, int `nSrcStep`, `Npp16s *pDst`, int `nDstStep`, `NppiSize oSizeROI`)

Select-channel 16-bit signed image copy for three-channel images.

- `NppStatus nppiCopy_16s_C4CR` (const `Npp16s *pSrc`, int `nSrcStep`, `Npp16s *pDst`, int `nDstStep`, `NppiSize oSizeROI`)

Select-channel 16-bit signed image copy for four-channel images.

- `NppStatus nppiCopy_16u_C3CR` (const `Npp16u *pSrc`, int `nSrcStep`, `Npp16u *pDst`, int `nDstStep`, `NppiSize oSizeROI`)

Select-channel 16-bit unsigned image copy for three-channel images.

- `NppStatus nppiCopy_16u_C4CR` (const `Npp16u *pSrc`, int `nSrcStep`, `Npp16u *pDst`, int `nDstStep`, `NppiSize oSizeROI`)

Select-channel 16-bit unsigned image copy for four-channel images.

- `NppStatus nppiCopy_32s_C3CR` (const `Npp32s *pSrc`, int `nSrcStep`, `Npp32s *pDst`, int `nDstStep`, `NppiSize oSizeROI`)

Select-channel 32-bit signed image copy for three-channel images.

- `NppStatus nppiCopy_32s_C4CR` (const `Npp32s *pSrc`, int `nSrcStep`, `Npp32s *pDst`, int `nDstStep`, `NppiSize oSizeROI`)

Select-channel 32-bit signed image copy for four-channel images.

- `NppStatus nppiCopy_32f_C3CR` (const `Npp32f *pSrc`, int `nSrcStep`, `Npp32f *pDst`, int `nDstStep`, `NppiSize oSizeROI`)

Select-channel 32-bit float image copy for three-channel images.

- `NppStatus nppiCopy_32f_C4CR` (const `Npp32f *pSrc`, int `nSrcStep`, `Npp32f *pDst`, int `nDstStep`, `NppiSize oSizeROI`)

Select-channel 32-bit float image copy for four-channel images.

Extract Channel Copy

The channel extract primitives copy a single color channel from a multi-channel source image to single-channel destination image.

The channel is selected by adjusting the source image pointer to point to the desired color channel (see [Channel-of-Interest API](#)).

- `NppStatus nppiCopy_8u_C3C1R` (const `Npp8u *pSrc`, int `nSrcStep`, `Npp8u *pDst`, int `nDstStep`, `NppiSize oSizeROI`)

Three-channel to single-channel 8-bit unsigned image copy.

- `NppStatus nppiCopy_8u_C4C1R` (const `Npp8u *pSrc`, int `nSrcStep`, `Npp8u *pDst`, int `nDstStep`, `NppiSize oSizeROI`)

Four-channel to single-channel 8-bit unsigned image copy.

- `NppStatus nppiCopy_16s_C3C1R` (const `Npp16s *pSrc`, int `nSrcStep`, `Npp16s *pDst`, int `nDstStep`, `NppiSize oSizeROI`)

Three-channel to single-channel 16-bit signed image copy.

- `NppStatus nppiCopy_16s_C4C1R` (const `Npp16s *pSrc`, int `nSrcStep`, `Npp16s *pDst`, int `nDstStep`, `NppiSize oSizeROI`)

Four-channel to single-channel 16-bit signed image copy.

- `NppStatus nppiCopy_16u_C3C1R` (const `Npp16u *pSrc`, int `nSrcStep`, `Npp16u *pDst`, int `nDstStep`, `NppiSize oSizeROI`)

Three-channel to single-channel 16-bit unsigned image copy.

- `NppStatus nppiCopy_16u_C4C1R` (const `Npp16u *pSrc`, int `nSrcStep`, `Npp16u *pDst`, int `nDstStep`, `NppiSize oSizeROI`)

Four-channel to single-channel 16-bit unsigned image copy.

- `NppStatus nppiCopy_32s_C3C1R` (const `Npp32s *pSrc`, int `nSrcStep`, `Npp32s *pDst`, int `nDstStep`, `NppiSize oSizeROI`)

Three-channel to single-channel 32-bit signed image copy.

- `NppStatus nppiCopy_32s_C4C1R` (const `Npp32s *pSrc`, int `nSrcStep`, `Npp32s *pDst`, int `nDstStep`, `NppiSize oSizeROI`)

Four-channel to single-channel 32-bit signed image copy.

- `NppStatus nppiCopy_32f_C3C1R` (const `Npp32f *pSrc`, int `nSrcStep`, `Npp32f *pDst`, int `nDstStep`, `NppiSize oSizeROI`)

Three-channel to single-channel 32-bit float image copy.

- `NppStatus nppiCopy_32f_C4C1R` (const `Npp32f *pSrc`, int `nSrcStep`, `Npp32f *pDst`, int `nDstStep`, `NppiSize oSizeROI`)

Four-channel to single-channel 32-bit float image copy.

Insert Channel Copy

The channel insert primitives copy a single-channel source image into one of the color channels in a multi-channel destination image.

The channel is selected by adjusting the destination image pointer to point to the desired color channel (see [Channel-of-Interest API](#)).

- `NppStatus nppiCopy_8u_C1C3R` (const `Npp8u *pSrc`, int `nSrcStep`, `Npp8u *pDst`, int `nDstStep`, `NppiSize oSizeROI`)

Single-channel to three-channel 8-bit unsigned image copy.

- `NppStatus nppiCopy_8u_C1C4R` (const `Npp8u *pSrc`, int `nSrcStep`, `Npp8u *pDst`, int `nDstStep`, `NppiSize oSizeROI`)

Single-channel to four-channel 8-bit unsigned image copy.

- `NppStatus nppiCopy_16s_C1C3R` (const `Npp16s *pSrc`, int `nSrcStep`, `Npp16s *pDst`, int `nDstStep`, `NppiSize oSizeROI`)

Single-channel to three-channel 16-bit signed image copy.

- `NppStatus nppiCopy_16s_C1C4R` (const `Npp16s *pSrc`, int `nSrcStep`, `Npp16s *pDst`, int `nDstStep`, `NppiSize oSizeROI`)

Single-channel to four-channel 16-bit signed image copy.

- `NppStatus nppiCopy_16u_C1C3R` (const `Npp16u *pSrc`, int `nSrcStep`, `Npp16u *pDst`, int `nDstStep`, `NppiSize oSizeROI`)

Single-channel to three-channel 16-bit unsigned image copy.

- `NppStatus nppiCopy_16u_C1C4R` (const `Npp16u *pSrc`, int `nSrcStep`, `Npp16u *pDst`, int `nDstStep`, `NppiSize oSizeROI`)

Single-channel to four-channel 16-bit unsigned image copy.

- `NppStatus nppiCopy_32s_C1C3R` (const `Npp32s *pSrc`, int `nSrcStep`, `Npp32s *pDst`, int `nDstStep`, `NppiSize oSizeROI`)

Single-channel to three-channel 32-bit signed image copy.

- `NppStatus nppiCopy_32s_C1C4R` (const `Npp32s *pSrc`, int `nSrcStep`, `Npp32s *pDst`, int `nDstStep`, `NppiSize oSizeROI`)

Single-channel to four-channel 32-bit signed image copy.

- `NppStatus nppiCopy_32f_C1C3R` (const `Npp32f *pSrc`, int `nSrcStep`, `Npp32f *pDst`, int `nDstStep`, `NppiSize oSizeROI`)

Single-channel to three-channel 32-bit float image copy.

- `NppStatus nppiCopy_32f_C1C4R` (const `Npp32f *pSrc`, int `nSrcStep`, `Npp32f *pDst`, int `nDstStep`, `NppiSize oSizeROI`)

Single-channel to four-channel 32-bit float image copy.

Packed-to-Planar Copy

Split a packed multi-channel image into a planar image.

E.g. copy the three channels of an RGB image into three separate single-channel images.

- `NppStatus nppiCopy_8u_C3P3R` (const `Npp8u` *pSrc, int nSrcStep, `Npp8u` *const aDst[3], int nDstStep, `NppiSize` oSizeROI)
Three-channel 8-bit unsigned packed to planar image copy.
- `NppStatus nppiCopy_8u_C4P4R` (const `Npp8u` *pSrc, int nSrcStep, `Npp8u` *const aDst[4], int nDstStep, `NppiSize` oSizeROI)
Four-channel 8-bit unsigned packed to planar image copy.
- `NppStatus nppiCopy_16s_C3P3R` (const `Npp16s` *pSrc, int nSrcStep, `Npp16s` *const aDst[3], int nDstStep, `NppiSize` oSizeROI)
Three-channel 16-bit signed packed to planar image copy.
- `NppStatus nppiCopy_16s_C4P4R` (const `Npp16s` *pSrc, int nSrcStep, `Npp16s` *const aDst[4], int nDstStep, `NppiSize` oSizeROI)
Four-channel 16-bit signed packed to planar image copy.
- `NppStatus nppiCopy_16u_C3P3R` (const `Npp16u` *pSrc, int nSrcStep, `Npp16u` *const aDst[3], int nDstStep, `NppiSize` oSizeROI)
Three-channel 16-bit unsigned packed to planar image copy.
- `NppStatus nppiCopy_16u_C4P4R` (const `Npp16u` *pSrc, int nSrcStep, `Npp16u` *const aDst[4], int nDstStep, `NppiSize` oSizeROI)
Four-channel 16-bit unsigned packed to planar image copy.
- `NppStatus nppiCopy_32s_C3P3R` (const `Npp32s` *pSrc, int nSrcStep, `Npp32s` *const aDst[3], int nDstStep, `NppiSize` oSizeROI)
Three-channel 32-bit signed packed to planar image copy.
- `NppStatus nppiCopy_32s_C4P4R` (const `Npp32s` *pSrc, int nSrcStep, `Npp32s` *const aDst[4], int nDstStep, `NppiSize` oSizeROI)
Four-channel 32-bit signed packed to planar image copy.
- `NppStatus nppiCopy_32f_C3P3R` (const `Npp32f` *pSrc, int nSrcStep, `Npp32f` *const aDst[3], int nDstStep, `NppiSize` oSizeROI)
Three-channel 32-bit float packed to planar image copy.
- `NppStatus nppiCopy_32f_C4P4R` (const `Npp32f` *pSrc, int nSrcStep, `Npp32f` *const aDst[4], int nDstStep, `NppiSize` oSizeROI)
Four-channel 32-bit float packed to planar image copy.

Planar-to-Packed Copy

Combine multiple image planes into a packed multi-channel image.

E.g. copy three single-channel images into a single 3-channel image.

- [NppStatus nppiCopy_8u_P3C3R](#) (const [Npp8u](#) *const aSrc[3], int nSrcStep, [Npp8u](#) *pDst, int nDstStep, [NppiSize](#) oSizeROI)
Three-channel 8-bit unsigned planar to packed image copy.
- [NppStatus nppiCopy_8u_P4C4R](#) (const [Npp8u](#) *const aSrc[4], int nSrcStep, [Npp8u](#) *pDst, int nDstStep, [NppiSize](#) oSizeROI)
Four-channel 8-bit unsigned planar to packed image copy.
- [NppStatus nppiCopy_16u_P3C3R](#) (const [Npp16u](#) *const aSrc[3], int nSrcStep, [Npp16u](#) *pDst, int nDstStep, [NppiSize](#) oSizeROI)
Three-channel 16-bit unsigned planar to packed image copy.
- [NppStatus nppiCopy_16u_P4C4R](#) (const [Npp16u](#) *const aSrc[4], int nSrcStep, [Npp16u](#) *pDst, int nDstStep, [NppiSize](#) oSizeROI)
Four-channel 16-bit unsigned planar to packed image copy.
- [NppStatus nppiCopy_16s_P3C3R](#) (const [Npp16s](#) *const aSrc[3], int nSrcStep, [Npp16s](#) *pDst, int nDstStep, [NppiSize](#) oSizeROI)
Three-channel 16-bit signed planar to packed image copy.
- [NppStatus nppiCopy_16s_P4C4R](#) (const [Npp16s](#) *const aSrc[4], int nSrcStep, [Npp16s](#) *pDst, int nDstStep, [NppiSize](#) oSizeROI)
Four-channel 16-bit signed planar to packed image copy.
- [NppStatus nppiCopy_32s_P3C3R](#) (const [Npp32s](#) *const aSrc[3], int nSrcStep, [Npp32s](#) *pDst, int nDstStep, [NppiSize](#) oSizeROI)
Three-channel 32-bit signed planar to packed image copy.
- [NppStatus nppiCopy_32s_P4C4R](#) (const [Npp32s](#) *const aSrc[4], int nSrcStep, [Npp32s](#) *pDst, int nDstStep, [NppiSize](#) oSizeROI)
Four-channel 32-bit signed planar to packed image copy.
- [NppStatus nppiCopy_32f_P3C3R](#) (const [Npp32f](#) *const aSrc[3], int nSrcStep, [Npp32f](#) *pDst, int nDstStep, [NppiSize](#) oSizeROI)
Three-channel 32-bit float planar to packed image copy.
- [NppStatus nppiCopy_32f_P4C4R](#) (const [Npp32f](#) *const aSrc[4], int nSrcStep, [Npp32f](#) *pDst, int nDstStep, [NppiSize](#) oSizeROI)
Four-channel 32-bit float planar to packed image copy.

7.7.1 Function Documentation

7.7.1.1 [NppStatus nppiCopy_16s_AC4MR](#) (const [Npp16s](#) *pSrc, int nSrcStep, [Npp16s](#) *pDst, int nDstStep, [NppiSize](#) oSizeROI, const [Npp8u](#) *pMask, int nMaskStep)

[Masked Operation](#) four channel 16-bit signed image copy, ignoring alpha.

Parameters:

pSrc [Source-Image Pointer](#).

nSrcStep Source-Image Line Step.
pDst Destination-Image Pointer.
nDstStep Destination-Image Line Step.
pMask Mask-Image Pointer.
nMaskStep Mask-Image Line Step.
oSizeROI Region-of-Interest (ROI).

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.7.1.2 NppStatus nppiCopy_16s_AC4R (const Npp16s * pSrc, int nSrcStep, Npp16s * pDst, int nDstStep, NppiSize oSizeROI)

4 channel 16-bit image copy, not affecting Alpha.

Parameters:

pSrc Source-Image Pointer.
nSrcStep Source-Image Line Step.
pDst Destination-Image Pointer.
nDstStep Destination-Image Line Step.
oSizeROI Region-of-Interest (ROI).

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.7.1.3 NppStatus nppiCopy_16s_C1C3R (const Npp16s * pSrc, int nSrcStep, Npp16s * pDst, int nDstStep, NppiSize oSizeROI)

Single-channel to three-channel 16-bit signed image copy.

Parameters:

pSrc Source-Image Pointer.
nSrcStep Source-Image Line Step.
pDst Select-Channel Destination-Image Pointer.
nDstStep Destination-Image Line Step.
oSizeROI Region-of-Interest (ROI).

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.7.1.4 `NppStatus nppiCopy_16s_C1C4R (const Npp16s * pSrc, int nSrcStep, Npp16s * pDst, int nDstStep, NppiSize oSizeROI)`

Single-channel to four-channel 16-bit signed image copy.

Parameters:

pSrc Source-Image Pointer.
nSrcStep Source-Image Line Step.
pDst Select-Channel Destination-Image Pointer.
nDstStep Destination-Image Line Step.
oSizeROI Region-of-Interest (ROI).

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.7.1.5 `NppStatus nppiCopy_16s_C1MR (const Npp16s * pSrc, int nSrcStep, Npp16s * pDst, int nDstStep, NppiSize oSizeROI, const Npp8u * pMask, int nMaskStep)`

Masked Operation 16-bit signed image copy.

Parameters:

pSrc Source-Image Pointer.
nSrcStep Source-Image Line Step.
pDst Destination-Image Pointer.
nDstStep Destination-Image Line Step.
pMask Mask-Image Pointer.
nMaskStep Mask-Image Line Step.
oSizeROI Region-of-Interest (ROI).

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.7.1.6 `NppStatus nppiCopy_16s_C1R (const Npp16s * pSrc, int nSrcStep, Npp16s * pDst, int nDstStep, NppiSize oSizeROI)`

16-bit image copy.

Parameters:

pSrc Source-Image Pointer.
nSrcStep Source-Image Line Step.
pDst Destination-Image Pointer.
nDstStep Destination-Image Line Step.
oSizeROI Region-of-Interest (ROI).

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.7.1.7 NppStatus nppiCopy_16s_C3C1R (const Npp16s * pSrc, int nSrcStep, Npp16s * pDst, int nDstStep, NppiSize oSizeROI)

Three-channel to single-channel 16-bit signed image copy.

Parameters:

pSrc Select-Channel Source-Image Pointer.
nSrcStep Source-Image Line Step.
pDst Destination-Image Pointer.
nDstStep Destination-Image Line Step.
oSizeROI Region-of-Interest (ROI).

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.7.1.8 NppStatus nppiCopy_16s_C3CR (const Npp16s * pSrc, int nSrcStep, Npp16s * pDst, int nDstStep, NppiSize oSizeROI)

Select-channel 16-bit signed image copy for three-channel images.

Parameters:

pSrc Select-Channel Source-Image Pointer.
nSrcStep Source-Image Line Step.
pDst Select-Channel Destination-Image Pointer.
nDstStep Destination-Image Line Step.
oSizeROI Region-of-Interest (ROI).

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.7.1.9 NppStatus nppiCopy_16s_C3MR (const Npp16s * pSrc, int nSrcStep, Npp16s * pDst, int nDstStep, NppiSize oSizeROI, const Npp8u * pMask, int nMaskStep)

Masked Operation three channel 16-bit signed image copy.

Parameters:

pSrc Source-Image Pointer.
nSrcStep Source-Image Line Step.
pDst Destination-Image Pointer.
nDstStep Destination-Image Line Step.
pMask Mask-Image Pointer.
nMaskStep Mask-Image Line Step.
oSizeROI Region-of-Interest (ROI).

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.7.1.10 NppStatus nppiCopy_16s_C3P3R (const Npp16s * pSrc, int nSrcStep, Npp16s *const aDst[3], int nDstStep, NppiSize oSizeROI)

Three-channel 16-bit signed packed to planar image copy.

Parameters:

- pSrc* Source-Image Pointer.
- nSrcStep* Source-Image Line Step.
- aDst* Destination-Planar-Image Pointer Array.
- nDstStep* Destination-Image Line Step.
- oSizeROI* Region-of-Interest (ROI).

Returns:

[Image Data Related Error Codes](#), [ROI Related Error Codes](#)

7.7.1.11 NppStatus nppiCopy_16s_C3R (const Npp16s * pSrc, int nSrcStep, Npp16s * pDst, int nDstStep, NppiSize oSizeROI)

Three channel 16-bit image copy.

Parameters:

- pSrc* Source-Image Pointer.
- nSrcStep* Source-Image Line Step.
- pDst* Destination-Image Pointer.
- nDstStep* Destination-Image Line Step.
- oSizeROI* Region-of-Interest (ROI).

Returns:

[Image Data Related Error Codes](#), [ROI Related Error Codes](#)

7.7.1.12 NppStatus nppiCopy_16s_C4C1R (const Npp16s * pSrc, int nSrcStep, Npp16s * pDst, int nDstStep, NppiSize oSizeROI)

Four-channel to single-channel 16-bit signed image copy.

Parameters:

- pSrc* Select-Channel Source-Image Pointer.
- nSrcStep* Source-Image Line Step.
- pDst* Destination-Image Pointer.
- nDstStep* Destination-Image Line Step.
- oSizeROI* Region-of-Interest (ROI).

Returns:

[Image Data Related Error Codes](#), [ROI Related Error Codes](#)

7.7.1.13 NppStatus nppiCopy_16s_C4CR (const Npp16s * pSrc, int nSrcStep, Npp16s * pDst, int nDstStep, NppiSize oSizeROI)

Select-channel 16-bit signed image copy for four-channel images.

Parameters:

pSrc Select-Channel Source-Image Pointer.
nSrcStep Source-Image Line Step.
pDst Select-Channel Destination-Image Pointer.
nDstStep Destination-Image Line Step.
oSizeROI Region-of-Interest (ROI).

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.7.1.14 NppStatus nppiCopy_16s_C4MR (const Npp16s * pSrc, int nSrcStep, Npp16s * pDst, int nDstStep, NppiSize oSizeROI, const Npp8u * pMask, int nMaskStep)

Masked Operation four channel 16-bit signed image copy.

Parameters:

pSrc Source-Image Pointer.
nSrcStep Source-Image Line Step.
pDst Destination-Image Pointer.
nDstStep Destination-Image Line Step.
pMask Mask-Image Pointer.
nMaskStep Mask-Image Line Step.
oSizeROI Region-of-Interest (ROI).

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.7.1.15 NppStatus nppiCopy_16s_C4P4R (const Npp16s * pSrc, int nSrcStep, Npp16s * const aDst[4], int nDstStep, NppiSize oSizeROI)

Four-channel 16-bit signed packed to planar image copy.

Parameters:

pSrc Source-Image Pointer.
nSrcStep Source-Image Line Step.
aDst Destination-Planar-Image Pointer Array.
nDstStep Destination-Image Line Step.
oSizeROI Region-of-Interest (ROI).

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.7.1.16 NppStatus nppiCopy_16s_C4R (const Npp16s * pSrc, int nSrcStep, Npp16s * pDst, int nDstStep, NppiSize oSizeROI)

4 channel 16-bit image copy.

Parameters:

pSrc Source-Image Pointer.
nSrcStep Source-Image Line Step.
pDst Destination-Image Pointer.
nDstStep Destination-Image Line Step.
oSizeROI Region-of-Interest (ROI).

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.7.1.17 NppStatus nppiCopy_16s_P3C3R (const Npp16s *const aSrc[3], int nSrcStep, Npp16s * pDst, int nDstStep, NppiSize oSizeROI)

Three-channel 16-bit signed planar to packed image copy.

Parameters:

aSrc Planar Source-Planar-Image Pointer Array.
nSrcStep Source-Image Line Step.
pDst Destination-Image Pointer.
nDstStep Destination-Image Line Step.
oSizeROI Region-of-Interest (ROI).

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.7.1.18 NppStatus nppiCopy_16s_P4C4R (const Npp16s *const aSrc[4], int nSrcStep, Npp16s * pDst, int nDstStep, NppiSize oSizeROI)

Four-channel 16-bit signed planar to packed image copy.

Parameters:

aSrc Planar Source-Planar-Image Pointer Array.
nSrcStep Source-Image Line Step.
pDst Destination-Image Pointer.
nDstStep Destination-Image Line Step.
oSizeROI Region-of-Interest (ROI).

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.7.1.19 NppStatus nppiCopy_16sc_AC4R (const Npp16sc * pSrc, int nSrcStep, Npp16sc * pDst, int nDstStep, NppiSize oSizeROI)

Four-channel 16-bit complex image copy, ignoring alpha.

Parameters:

pSrc Source-Image Pointer.
nSrcStep Source-Image Line Step.
pDst Destination-Image Pointer.
nDstStep Destination-Image Line Step.
oSizeROI Region-of-Interest (ROI).

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.7.1.20 NppStatus nppiCopy_16sc_C1R (const Npp16sc * pSrc, int nSrcStep, Npp16sc * pDst, int nDstStep, NppiSize oSizeROI)

16-bit complex image copy.

Parameters:

pSrc Source-Image Pointer.
nSrcStep Source-Image Line Step.
pDst Destination-Image Pointer.
nDstStep Destination-Image Line Step.
oSizeROI Region-of-Interest (ROI).

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.7.1.21 NppStatus nppiCopy_16sc_C2R (const Npp16sc * pSrc, int nSrcStep, Npp16sc * pDst, int nDstStep, NppiSize oSizeROI)

Two-channel 16-bit complex image copy.

Parameters:

pSrc Source-Image Pointer.
nSrcStep Source-Image Line Step.
pDst Destination-Image Pointer.
nDstStep Destination-Image Line Step.
oSizeROI Region-of-Interest (ROI).

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.7.1.22 NppStatus nppiCopy_16sc_C3R (const Npp16sc * pSrc, int nSrcStep, Npp16sc * pDst, int nDstStep, NppiSize oSizeROI)

Three-channel 16-bit complex image copy.

Parameters:

pSrc Source-Image Pointer.
nSrcStep Source-Image Line Step.
pDst Destination-Image Pointer.
nDstStep Destination-Image Line Step.
oSizeROI Region-of-Interest (ROI).

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.7.1.23 NppStatus nppiCopy_16sc_C4R (const Npp16sc * pSrc, int nSrcStep, Npp16sc * pDst, int nDstStep, NppiSize oSizeROI)

Four-channel 16-bit complex image copy.

Parameters:

pSrc Source-Image Pointer.
nSrcStep Source-Image Line Step.
pDst Destination-Image Pointer.
nDstStep Destination-Image Line Step.
oSizeROI Region-of-Interest (ROI).

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.7.1.24 NppStatus nppiCopy_16u_AC4MR (const Npp16u * pSrc, int nSrcStep, Npp16u * pDst, int nDstStep, NppiSize oSizeROI, const Npp8u * pMask, int nMaskStep)

Masked Operation four channel 16-bit unsigned image copy, ignoring alpha.

Parameters:

pSrc Source-Image Pointer.
nSrcStep Source-Image Line Step.
pDst Destination-Image Pointer.
nDstStep Destination-Image Line Step.
pMask Mask-Image Pointer.
nMaskStep Mask-Image Line Step.
oSizeROI Region-of-Interest (ROI).

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.7.1.25 NppStatus nppiCopy_16u_AC4R (const Npp16u * pSrc, int nSrcStep, Npp16u * pDst, int nDstStep, NppiSize oSizeROI)

4 channel 16-bit unsigned image copy, not affecting Alpha channel.

Parameters:

pSrc Source-Image Pointer.
nSrcStep Source-Image Line Step.
pDst Destination-Image Pointer.
nDstStep Destination-Image Line Step.
oSizeROI Region-of-Interest (ROI).

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.7.1.26 NppStatus nppiCopy_16u_C1C3R (const Npp16u * pSrc, int nSrcStep, Npp16u * pDst, int nDstStep, NppiSize oSizeROI)

Single-channel to three-channel 16-bit unsigned image copy.

Parameters:

pSrc Source-Image Pointer.
nSrcStep Source-Image Line Step.
pDst Select-Channel Destination-Image Pointer.
nDstStep Destination-Image Line Step.
oSizeROI Region-of-Interest (ROI).

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.7.1.27 NppStatus nppiCopy_16u_C1C4R (const Npp16u * pSrc, int nSrcStep, Npp16u * pDst, int nDstStep, NppiSize oSizeROI)

Single-channel to four-channel 16-bit unsigned image copy.

Parameters:

pSrc Source-Image Pointer.
nSrcStep Source-Image Line Step.
pDst Select-Channel Destination-Image Pointer.
nDstStep Destination-Image Line Step.
oSizeROI Region-of-Interest (ROI).

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.7.1.28 NppStatus nppiCopy_16u_C1MR (*const Npp16u * pSrc, int nSrcStep, Npp16u * pDst, int nDstStep, NppiSize oSizeROI, const Npp8u * pMask, int nMaskStep*)

[Masked Operation](#) 16-bit unsigned image copy.

Parameters:

pSrc Source-Image Pointer.
nSrcStep Source-Image Line Step.
pDst Destination-Image Pointer.
nDstStep Destination-Image Line Step.
pMask Mask-Image Pointer.
nMaskStep Mask-Image Line Step.
oSizeROI Region-of-Interest (ROI).

Returns:

[Image Data Related Error Codes](#), [ROI Related Error Codes](#)

7.7.1.29 NppStatus nppiCopy_16u_C1R (*const Npp16u * pSrc, int nSrcStep, Npp16u * pDst, int nDstStep, NppiSize oSizeROI*)

16-bit unsigned image copy.

Parameters:

pSrc Source-Image Pointer.
nSrcStep Source-Image Line Step.
pDst Destination-Image Pointer.
nDstStep Destination-Image Line Step.
oSizeROI Region-of-Interest (ROI).

Returns:

[Image Data Related Error Codes](#), [ROI Related Error Codes](#)

7.7.1.30 NppStatus nppiCopy_16u_C3C1R (*const Npp16u * pSrc, int nSrcStep, Npp16u * pDst, int nDstStep, NppiSize oSizeROI*)

Three-channel to single-channel 16-bit unsigned image copy.

Parameters:

pSrc Select-Channel Source-Image Pointer.
nSrcStep Source-Image Line Step.
pDst Destination-Image Pointer.
nDstStep Destination-Image Line Step.
oSizeROI Region-of-Interest (ROI).

Returns:

[Image Data Related Error Codes](#), [ROI Related Error Codes](#)

7.7.1.31 NppStatus nppiCopy_16u_C3CR (const Npp16u * pSrc, int nSrcStep, Npp16u * pDst, int nDstStep, NppiSize oSizeROI)

Select-channel 16-bit unsigned image copy for three-channel images.

Parameters:

pSrc Select-Channel Source-Image Pointer.
nSrcStep Source-Image Line Step.
pDst Select-Channel Destination-Image Pointer.
nDstStep Destination-Image Line Step.
oSizeROI Region-of-Interest (ROI).

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.7.1.32 NppStatus nppiCopy_16u_C3MR (const Npp16u * pSrc, int nSrcStep, Npp16u * pDst, int nDstStep, NppiSize oSizeROI, const Npp8u * pMask, int nMaskStep)

Masked Operation three channel 16-bit unsigned image copy.

Parameters:

pSrc Source-Image Pointer.
nSrcStep Source-Image Line Step.
pDst Destination-Image Pointer.
nDstStep Destination-Image Line Step.
pMask Mask-Image Pointer.
nMaskStep Mask-Image Line Step.
oSizeROI Region-of-Interest (ROI).

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.7.1.33 NppStatus nppiCopy_16u_C3P3R (const Npp16u * pSrc, int nSrcStep, Npp16u *const aDst[3], int nDstStep, NppiSize oSizeROI)

Three-channel 16-bit unsigned packed to planar image copy.

Parameters:

pSrc Source-Image Pointer.
nSrcStep Source-Image Line Step.
aDst Destination-Planar-Image Pointer Array.
nDstStep Destination-Image Line Step.
oSizeROI Region-of-Interest (ROI).

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.7.1.34 NppStatus nppiCopy_16u_C3R (const Npp16u * pSrc, int nSrcStep, Npp16u * pDst, int nDstStep, NppiSize oSizeROI)

Three channel 16-bit unsigned image copy.

Parameters:

pSrc Source-Image Pointer.
nSrcStep Source-Image Line Step.
pDst Destination-Image Pointer.
nDstStep Destination-Image Line Step.
oSizeROI Region-of-Interest (ROI).

Returns:

[Image Data Related Error Codes](#), [ROI Related Error Codes](#)

7.7.1.35 NppStatus nppiCopy_16u_C4C1R (const Npp16u * pSrc, int nSrcStep, Npp16u * pDst, int nDstStep, NppiSize oSizeROI)

Four-channel to single-channel 16-bit unsigned image copy.

Parameters:

pSrc Select-Channel Source-Image Pointer.
nSrcStep Source-Image Line Step.
pDst Destination-Image Pointer.
nDstStep Destination-Image Line Step.
oSizeROI Region-of-Interest (ROI).

Returns:

[Image Data Related Error Codes](#), [ROI Related Error Codes](#)

7.7.1.36 NppStatus nppiCopy_16u_C4CR (const Npp16u * pSrc, int nSrcStep, Npp16u * pDst, int nDstStep, NppiSize oSizeROI)

Select-channel 16-bit unsigned image copy for four-channel images.

Parameters:

pSrc Select-Channel Source-Image Pointer.
nSrcStep Source-Image Line Step.
pDst Select-Channel Destination-Image Pointer.
nDstStep Destination-Image Line Step.
oSizeROI Region-of-Interest (ROI).

Returns:

[Image Data Related Error Codes](#), [ROI Related Error Codes](#)

7.7.1.37 NppStatus nppiCopy_16u_C4MR (const Npp16u * *pSrc*, int *nSrcStep*, Npp16u * *pDst*, int *nDstStep*, NppiSize *oSizeROI*, const Npp8u * *pMask*, int *nMaskStep*)

Masked Operation four channel 16-bit unsigned image copy.

Parameters:

pSrc Source-Image Pointer.
nSrcStep Source-Image Line Step.
pDst Destination-Image Pointer.
nDstStep Destination-Image Line Step.
pMask Mask-Image Pointer.
nMaskStep Mask-Image Line Step.
oSizeROI Region-of-Interest (ROI).

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.7.1.38 NppStatus nppiCopy_16u_C4P4R (const Npp16u * *pSrc*, int *nSrcStep*, Npp16u *const *aDst*[4], int *nDstStep*, NppiSize *oSizeROI*)

Four-channel 16-bit unsigned packed to planar image copy.

Parameters:

pSrc Source-Image Pointer.
nSrcStep Source-Image Line Step.
aDst Destination-Planar-Image Pointer Array.
nDstStep Destination-Image Line Step.
oSizeROI Region-of-Interest (ROI).

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.7.1.39 NppStatus nppiCopy_16u_C4R (const Npp16u * *pSrc*, int *nSrcStep*, Npp16u * *pDst*, int *nDstStep*, NppiSize *oSizeROI*)

4 channel 16-bit unsigned image copy.

Parameters:

pSrc Source-Image Pointer.
nSrcStep Source-Image Line Step.
pDst Destination-Image Pointer.
nDstStep Destination-Image Line Step.
oSizeROI Region-of-Interest (ROI).

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.7.1.40 `NppStatus nppiCopy_16u_P3C3R (const Npp16u *const aSrc[3], int nSrcStep, Npp16u * pDst, int nDstStep, NppiSize oSizeROI)`

Three-channel 16-bit unsigned planar to packed image copy.

Parameters:

aSrc Planar [Source-Planar-Image Pointer Array](#).
nSrcStep [Source-Image Line Step](#).
pDst [Destination-Image Pointer](#).
nDstStep [Destination-Image Line Step](#).
oSizeROI [Region-of-Interest \(ROI\)](#).

Returns:

[Image Data Related Error Codes](#), [ROI Related Error Codes](#)

7.7.1.41 `NppStatus nppiCopy_16u_P4C4R (const Npp16u *const aSrc[4], int nSrcStep, Npp16u * pDst, int nDstStep, NppiSize oSizeROI)`

Four-channel 16-bit unsigned planar to packed image copy.

Parameters:

aSrc Planar [Source-Planar-Image Pointer Array](#).
nSrcStep [Source-Image Line Step](#).
pDst [Destination-Image Pointer](#).
nDstStep [Destination-Image Line Step](#).
oSizeROI [Region-of-Interest \(ROI\)](#).

Returns:

[Image Data Related Error Codes](#), [ROI Related Error Codes](#)

7.7.1.42 `NppStatus nppiCopy_32f_AC4MR (const Npp32f * pSrc, int nSrcStep, Npp32f * pDst, int nDstStep, NppiSize oSizeROI, const Npp8u * pMask, int nMaskStep)`

[Masked Operation](#) four channel 32-bit float image copy, ignoring alpha.

Parameters:

pSrc [Source-Image Pointer](#).
nSrcStep [Source-Image Line Step](#).
pDst [Destination-Image Pointer](#).
nDstStep [Destination-Image Line Step](#).
pMask [Mask-Image Pointer](#).
nMaskStep [Mask-Image Line Step](#).
oSizeROI [Region-of-Interest \(ROI\)](#).

Returns:

[Image Data Related Error Codes](#), [ROI Related Error Codes](#)

7.7.1.43 NppStatus nppiCopy_32f_AC4R (const Npp32f * pSrc, int nSrcStep, Npp32f * pDst, int nDstStep, NppiSize oSizeROI)

4 channel 32-bit floating point image copy, not affecting Alpha.

Parameters:

pSrc Source-Image Pointer.
nSrcStep Source-Image Line Step.
pDst Destination-Image Pointer.
nDstStep Destination-Image Line Step.
oSizeROI Region-of-Interest (ROI).

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.7.1.44 NppStatus nppiCopy_32f_C1C3R (const Npp32f * pSrc, int nSrcStep, Npp32f * pDst, int nDstStep, NppiSize oSizeROI)

Single-channel to three-channel 32-bit float image copy.

Parameters:

pSrc Source-Image Pointer.
nSrcStep Source-Image Line Step.
pDst Select-Channel Destination-Image Pointer.
nDstStep Destination-Image Line Step.
oSizeROI Region-of-Interest (ROI).

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.7.1.45 NppStatus nppiCopy_32f_C1C4R (const Npp32f * pSrc, int nSrcStep, Npp32f * pDst, int nDstStep, NppiSize oSizeROI)

Single-channel to four-channel 32-bit float image copy.

Parameters:

pSrc Source-Image Pointer.
nSrcStep Source-Image Line Step.
pDst Select-Channel Destination-Image Pointer.
nDstStep Destination-Image Line Step.
oSizeROI Region-of-Interest (ROI).

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.7.1.46 `NppStatus nppiCopy_32f_C1MR (const Npp32f * pSrc, int nSrcStep, Npp32f * pDst, int nDstStep, NppiSize oSizeROI, const Npp8u * pMask, int nMaskStep)`

Masked Operation 32-bit float image copy.

Parameters:

pSrc Source-Image Pointer.
nSrcStep Source-Image Line Step.
pDst Destination-Image Pointer.
nDstStep Destination-Image Line Step.
pMask Mask-Image Pointer.
nMaskStep Mask-Image Line Step.
oSizeROI Region-of-Interest (ROI).

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.7.1.47 `NppStatus nppiCopy_32f_C1R (const Npp32f * pSrc, int nSrcStep, Npp32f * pDst, int nDstStep, NppiSize oSizeROI)`

32-bit floating point image copy.

Parameters:

pSrc Source-Image Pointer.
nSrcStep Source-Image Line Step.
pDst Destination-Image Pointer.
nDstStep Destination-Image Line Step.
oSizeROI Region-of-Interest (ROI).

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.7.1.48 `NppStatus nppiCopy_32f_C3C1R (const Npp32f * pSrc, int nSrcStep, Npp32f * pDst, int nDstStep, NppiSize oSizeROI)`

Three-channel to single-channel 32-bit float image copy.

Parameters:

pSrc Select-Channel Source-Image Pointer.
nSrcStep Source-Image Line Step.
pDst Destination-Image Pointer.
nDstStep Destination-Image Line Step.
oSizeROI Region-of-Interest (ROI).

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.7.1.49 NppStatus nppiCopy_32f_C3CR (const Npp32f * pSrc, int nSrcStep, Npp32f * pDst, int nDstStep, NppiSize oSizeROI)

Select-channel 32-bit float image copy for three-channel images.

Parameters:

pSrc Select-Channel Source-Image Pointer.
nSrcStep Source-Image Line Step.
pDst Select-Channel Destination-Image Pointer.
nDstStep Destination-Image Line Step.
oSizeROI Region-of-Interest (ROI).

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.7.1.50 NppStatus nppiCopy_32f_C3MR (const Npp32f * pSrc, int nSrcStep, Npp32f * pDst, int nDstStep, NppiSize oSizeROI, const Npp8u * pMask, int nMaskStep)

Masked Operation three channel 32-bit float image copy.

Parameters:

pSrc Source-Image Pointer.
nSrcStep Source-Image Line Step.
pDst Destination-Image Pointer.
nDstStep Destination-Image Line Step.
pMask Mask-Image Pointer.
nMaskStep Mask-Image Line Step.
oSizeROI Region-of-Interest (ROI).

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.7.1.51 NppStatus nppiCopy_32f_C3P3R (const Npp32f * pSrc, int nSrcStep, Npp32f * const aDst[3], int nDstStep, NppiSize oSizeROI)

Three-channel 32-bit float packed to planar image copy.

Parameters:

pSrc Source-Image Pointer.
nSrcStep Source-Image Line Step.
aDst Destination-Planar-Image Pointer Array.
nDstStep Destination-Image Line Step.
oSizeROI Region-of-Interest (ROI).

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.7.1.52 NppStatus nppiCopy_32f_C3R (const Npp32f * pSrc, int nSrcStep, Npp32f * pDst, int nDstStep, NppiSize oSizeROI)

Three channel 32-bit floating point image copy.

Parameters:

- pSrc* Source-Image Pointer.
- nSrcStep* Source-Image Line Step.
- pDst* Destination-Image Pointer.
- nDstStep* Destination-Image Line Step.
- oSizeROI* Region-of-Interest (ROI).

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.7.1.53 NppStatus nppiCopy_32f_C4C1R (const Npp32f * pSrc, int nSrcStep, Npp32f * pDst, int nDstStep, NppiSize oSizeROI)

Four-channel to single-channel 32-bit float image copy.

Parameters:

- pSrc* Select-Channel Source-Image Pointer.
- nSrcStep* Source-Image Line Step.
- pDst* Destination-Image Pointer.
- nDstStep* Destination-Image Line Step.
- oSizeROI* Region-of-Interest (ROI).

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.7.1.54 NppStatus nppiCopy_32f_C4CR (const Npp32f * pSrc, int nSrcStep, Npp32f * pDst, int nDstStep, NppiSize oSizeROI)

Select-channel 32-bit float image copy for four-channel images.

Parameters:

- pSrc* Select-Channel Source-Image Pointer.
- nSrcStep* Source-Image Line Step.
- pDst* Select-Channel Destination-Image Pointer.
- nDstStep* Destination-Image Line Step.
- oSizeROI* Region-of-Interest (ROI).

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.7.1.55 `NppStatus nppiCopy_32f_C4MR (const Npp32f * pSrc, int nSrcStep, Npp32f * pDst, int nDstStep, NppiSize oSizeROI, const Npp8u * pMask, int nMaskStep)`

Masked Operation four channel 32-bit float image copy.

Parameters:

pSrc Source-Image Pointer.
nSrcStep Source-Image Line Step.
pDst Destination-Image Pointer.
nDstStep Destination-Image Line Step.
pMask Mask-Image Pointer.
nMaskStep Mask-Image Line Step.
oSizeROI Region-of-Interest (ROI).

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.7.1.56 `NppStatus nppiCopy_32f_C4P4R (const Npp32f * pSrc, int nSrcStep, Npp32f *const aDst[4], int nDstStep, NppiSize oSizeROI)`

Four-channel 32-bit float packed to planar image copy.

Parameters:

pSrc Source-Image Pointer.
nSrcStep Source-Image Line Step.
aDst Destination-Planar-Image Pointer Array.
nDstStep Destination-Image Line Step.
oSizeROI Region-of-Interest (ROI).

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.7.1.57 `NppStatus nppiCopy_32f_C4R (const Npp32f * pSrc, int nSrcStep, Npp32f * pDst, int nDstStep, NppiSize oSizeROI)`

4 channel 32-bit floating point image copy.

Parameters:

pSrc Source-Image Pointer.
nSrcStep Source-Image Line Step.
pDst Destination-Image Pointer.
nDstStep Destination-Image Line Step.
oSizeROI Region-of-Interest (ROI).

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.7.1.58 NppStatus nppiCopy_32f_P3C3R (const Npp32f *const aSrc[3], int nSrcStep, Npp32f * pDst, int nDstStep, NppiSize oSizeROI)

Three-channel 32-bit float planar to packed image copy.

Parameters:

aSrc Planar [Source-Planar-Image Pointer Array](#).
nSrcStep [Source-Image Line Step](#).
pDst [Destination-Image Pointer](#).
nDstStep [Destination-Image Line Step](#).
oSizeROI [Region-of-Interest \(ROI\)](#).

Returns:

[Image Data Related Error Codes](#), [ROI Related Error Codes](#)

7.7.1.59 NppStatus nppiCopy_32f_P4C4R (const Npp32f *const aSrc[4], int nSrcStep, Npp32f * pDst, int nDstStep, NppiSize oSizeROI)

Four-channel 32-bit float planar to packed image copy.

Parameters:

aSrc Planar [Source-Planar-Image Pointer Array](#).
nSrcStep [Source-Image Line Step](#).
pDst [Destination-Image Pointer](#).
nDstStep [Destination-Image Line Step](#).
oSizeROI [Region-of-Interest \(ROI\)](#).

Returns:

[Image Data Related Error Codes](#), [ROI Related Error Codes](#)

7.7.1.60 NppStatus nppiCopy_32fc_AC4R (const Npp32fc * pSrc, int nSrcStep, Npp32fc * pDst, int nDstStep, NppiSize oSizeROI)

Four-channel 32-bit floating-point complex image copy, ignoring alpha.

Parameters:

pSrc [Source-Image Pointer](#).
nSrcStep [Source-Image Line Step](#).
pDst [Destination-Image Pointer](#).
nDstStep [Destination-Image Line Step](#).
oSizeROI [Region-of-Interest \(ROI\)](#).

Returns:

[Image Data Related Error Codes](#), [ROI Related Error Codes](#)

7.7.1.61 NppStatus nppiCopy_32fc_C1R (const Npp32fc * pSrc, int nSrcStep, Npp32fc * pDst, int nDstStep, NppiSize oSizeROI)

32-bit floating-point complex image copy.

Parameters:

- pSrc* Source-Image Pointer.
- nSrcStep* Source-Image Line Step.
- pDst* Destination-Image Pointer.
- nDstStep* Destination-Image Line Step.
- oSizeROI* Region-of-Interest (ROI).

Returns:

[Image Data Related Error Codes](#), [ROI Related Error Codes](#)

7.7.1.62 NppStatus nppiCopy_32fc_C2R (const Npp32fc * pSrc, int nSrcStep, Npp32fc * pDst, int nDstStep, NppiSize oSizeROI)

Two-channel 32-bit floating-point complex image copy.

Parameters:

- pSrc* Source-Image Pointer.
- nSrcStep* Source-Image Line Step.
- pDst* Destination-Image Pointer.
- nDstStep* Destination-Image Line Step.
- oSizeROI* Region-of-Interest (ROI).

Returns:

[Image Data Related Error Codes](#), [ROI Related Error Codes](#)

7.7.1.63 NppStatus nppiCopy_32fc_C3R (const Npp32fc * pSrc, int nSrcStep, Npp32fc * pDst, int nDstStep, NppiSize oSizeROI)

Three-channel 32-bit floating-point complex image copy.

Parameters:

- pSrc* Source-Image Pointer.
- nSrcStep* Source-Image Line Step.
- pDst* Destination-Image Pointer.
- nDstStep* Destination-Image Line Step.
- oSizeROI* Region-of-Interest (ROI).

Returns:

[Image Data Related Error Codes](#), [ROI Related Error Codes](#)

7.7.1.64 NppStatus nppiCopy_32fc_C4R (const Npp32fc * pSrc, int nSrcStep, Npp32fc * pDst, int nDstStep, NppiSize oSizeROI)

Four-channel 32-bit floating-point complex image copy.

Parameters:

pSrc Source-Image Pointer.
nSrcStep Source-Image Line Step.
pDst Destination-Image Pointer.
nDstStep Destination-Image Line Step.
oSizeROI Region-of-Interest (ROI).

Returns:

[Image Data Related Error Codes](#), [ROI Related Error Codes](#)

7.7.1.65 NppStatus nppiCopy_32s_AC4MR (const Npp32s * pSrc, int nSrcStep, Npp32s * pDst, int nDstStep, NppiSize oSizeROI, const Npp8u * pMask, int nMaskStep)

[Masked Operation](#) four channel 32-bit signed image copy, ignoring alpha.

Parameters:

pSrc Source-Image Pointer.
nSrcStep Source-Image Line Step.
pDst Destination-Image Pointer.
nDstStep Destination-Image Line Step.
pMask Mask-Image Pointer.
nMaskStep Mask-Image Line Step.
oSizeROI Region-of-Interest (ROI).

Returns:

[Image Data Related Error Codes](#), [ROI Related Error Codes](#)

7.7.1.66 NppStatus nppiCopy_32s_AC4R (const Npp32s * pSrc, int nSrcStep, Npp32s * pDst, int nDstStep, NppiSize oSizeROI)

4 channel 32-bit image copy, not affecting Alpha.

Parameters:

pSrc Source-Image Pointer.
nSrcStep Source-Image Line Step.
pDst Destination-Image Pointer.
nDstStep Destination-Image Line Step.
oSizeROI Region-of-Interest (ROI).

Returns:

[Image Data Related Error Codes](#), [ROI Related Error Codes](#)

7.7.1.67 NppStatus nppiCopy_32s_C1C3R (const Npp32s * pSrc, int nSrcStep, Npp32s * pDst, int nDstStep, NppiSize oSizeROI)

Single-channel to three-channel 32-bit signed image copy.

Parameters:

pSrc Source-Image Pointer.
nSrcStep Source-Image Line Step.
pDst Select-Channel Destination-Image Pointer.
nDstStep Destination-Image Line Step.
oSizeROI Region-of-Interest (ROI).

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.7.1.68 NppStatus nppiCopy_32s_C1C4R (const Npp32s * pSrc, int nSrcStep, Npp32s * pDst, int nDstStep, NppiSize oSizeROI)

Single-channel to four-channel 32-bit signed image copy.

Parameters:

pSrc Source-Image Pointer.
nSrcStep Source-Image Line Step.
pDst Select-Channel Destination-Image Pointer.
nDstStep Destination-Image Line Step.
oSizeROI Region-of-Interest (ROI).

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.7.1.69 NppStatus nppiCopy_32s_C1MR (const Npp32s * pSrc, int nSrcStep, Npp32s * pDst, int nDstStep, NppiSize oSizeROI, const Npp8u * pMask, int nMaskStep)

Masked Operation 32-bit signed image copy.

Parameters:

pSrc Source-Image Pointer.
nSrcStep Source-Image Line Step.
pDst Destination-Image Pointer.
nDstStep Destination-Image Line Step.
pMask Mask-Image Pointer.
nMaskStep Mask-Image Line Step.
oSizeROI Region-of-Interest (ROI).

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.7.1.70 NppStatus nppiCopy_32s_C1R (const Npp32s * pSrc, int nSrcStep, Npp32s * pDst, int nDstStep, NppiSize oSizeROI)

32-bit image copy.

Parameters:

pSrc Source-Image Pointer.
nSrcStep Source-Image Line Step.
pDst Destination-Image Pointer.
nDstStep Destination-Image Line Step.
oSizeROI Region-of-Interest (ROI).

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.7.1.71 NppStatus nppiCopy_32s_C3C1R (const Npp32s * pSrc, int nSrcStep, Npp32s * pDst, int nDstStep, NppiSize oSizeROI)

Three-channel to single-channel 32-bit signed image copy.

Parameters:

pSrc Select-Channel Source-Image Pointer.
nSrcStep Source-Image Line Step.
pDst Destination-Image Pointer.
nDstStep Destination-Image Line Step.
oSizeROI Region-of-Interest (ROI).

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.7.1.72 NppStatus nppiCopy_32s_C3CR (const Npp32s * pSrc, int nSrcStep, Npp32s * pDst, int nDstStep, NppiSize oSizeROI)

Select-channel 32-bit signed image copy for three-channel images.

Parameters:

pSrc Select-Channel Source-Image Pointer.
nSrcStep Source-Image Line Step.
pDst Select-Channel Destination-Image Pointer.
nDstStep Destination-Image Line Step.
oSizeROI Region-of-Interest (ROI).

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.7.1.73 `NppStatus nppiCopy_32s_C3MR (const Npp32s * pSrc, int nSrcStep, Npp32s * pDst, int nDstStep, NppiSize oSizeROI, const Npp8u * pMask, int nMaskStep)`

Masked Operation three channel 32-bit signed image copy.

Parameters:

pSrc Source-Image Pointer.
nSrcStep Source-Image Line Step.
pDst Destination-Image Pointer.
nDstStep Destination-Image Line Step.
pMask Mask-Image Pointer.
nMaskStep Mask-Image Line Step.
oSizeROI Region-of-Interest (ROI).

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.7.1.74 `NppStatus nppiCopy_32s_C3P3R (const Npp32s * pSrc, int nSrcStep, Npp32s *const aDst[3], int nDstStep, NppiSize oSizeROI)`

Three-channel 32-bit signed packed to planar image copy.

Parameters:

pSrc Source-Image Pointer.
nSrcStep Source-Image Line Step.
aDst Destination-Planar-Image Pointer Array.
nDstStep Destination-Image Line Step.
oSizeROI Region-of-Interest (ROI).

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.7.1.75 `NppStatus nppiCopy_32s_C3R (const Npp32s * pSrc, int nSrcStep, Npp32s * pDst, int nDstStep, NppiSize oSizeROI)`

Three channel 32-bit image copy.

Parameters:

pSrc Source-Image Pointer.
nSrcStep Source-Image Line Step.
pDst Destination-Image Pointer.
nDstStep Destination-Image Line Step.
oSizeROI Region-of-Interest (ROI).

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.7.1.76 NppStatus nppiCopy_32s_C4C1R (const Npp32s * pSrc, int nSrcStep, Npp32s * pDst, int nDstStep, NppiSize oSizeROI)

Four-channel to single-channel 32-bit signed image copy.

Parameters:

pSrc Select-Channel Source-Image Pointer.
nSrcStep Source-Image Line Step.
pDst Destination-Image Pointer.
nDstStep Destination-Image Line Step.
oSizeROI Region-of-Interest (ROI).

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.7.1.77 NppStatus nppiCopy_32s_C4CR (const Npp32s * pSrc, int nSrcStep, Npp32s * pDst, int nDstStep, NppiSize oSizeROI)

Select-channel 32-bit signed image copy for four-channel images.

Parameters:

pSrc Select-Channel Source-Image Pointer.
nSrcStep Source-Image Line Step.
pDst Select-Channel Destination-Image Pointer.
nDstStep Destination-Image Line Step.
oSizeROI Region-of-Interest (ROI).

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.7.1.78 NppStatus nppiCopy_32s_C4MR (const Npp32s * pSrc, int nSrcStep, Npp32s * pDst, int nDstStep, NppiSize oSizeROI, const Npp8u * pMask, int nMaskStep)

Masked Operation four channel 32-bit signed image copy.

Parameters:

pSrc Source-Image Pointer.
nSrcStep Source-Image Line Step.
pDst Destination-Image Pointer.
nDstStep Destination-Image Line Step.
pMask Mask-Image Pointer.
nMaskStep Mask-Image Line Step.
oSizeROI Region-of-Interest (ROI).

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.7.1.79 NppStatus nppiCopy_32s_C4P4R (const Npp32s * pSrc, int nSrcStep, Npp32s *const aDst[4], int nDstStep, NppiSize oSizeROI)

Four-channel 32-bit signed packed to planar image copy.

Parameters:

pSrc Source-Image Pointer.
nSrcStep Source-Image Line Step.
aDst Destination-Planar-Image Pointer Array.
nDstStep Destination-Image Line Step.
oSizeROI Region-of-Interest (ROI).

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.7.1.80 NppStatus nppiCopy_32s_C4R (const Npp32s * pSrc, int nSrcStep, Npp32s * pDst, int nDstStep, NppiSize oSizeROI)

4 channel 32-bit image copy.

Parameters:

pSrc Source-Image Pointer.
nSrcStep Source-Image Line Step.
pDst Destination-Image Pointer.
nDstStep Destination-Image Line Step.
oSizeROI Region-of-Interest (ROI).

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.7.1.81 NppStatus nppiCopy_32s_P3C3R (const Npp32s *const aSrc[3], int nSrcStep, Npp32s * pDst, int nDstStep, NppiSize oSizeROI)

Three-channel 32-bit signed planar to packed image copy.

Parameters:

aSrc Planar Source-Planar-Image Pointer Array.
nSrcStep Source-Image Line Step.
pDst Destination-Image Pointer.
nDstStep Destination-Image Line Step.
oSizeROI Region-of-Interest (ROI).

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.7.1.82 NppStatus nppiCopy_32s_P4C4R (const Npp32s *const aSrc[4], int nSrcStep, Npp32s * pDst, int nDstStep, NppiSize oSizeROI)

Four-channel 32-bit signed planar to packed image copy.

Parameters:

aSrc Planar Source-Planar-Image Pointer Array.
nSrcStep Source-Image Line Step.
pDst Destination-Image Pointer.
nDstStep Destination-Image Line Step.
oSizeROI Region-of-Interest (ROI).

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.7.1.83 NppStatus nppiCopy_32sc_AC4R (const Npp32sc * pSrc, int nSrcStep, Npp32sc * pDst, int nDstStep, NppiSize oSizeROI)

Four-channel 32-bit complex image copy, ignoring alpha.

Parameters:

pSrc Source-Image Pointer.
nSrcStep Source-Image Line Step.
pDst Destination-Image Pointer.
nDstStep Destination-Image Line Step.
oSizeROI Region-of-Interest (ROI).

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.7.1.84 NppStatus nppiCopy_32sc_C1R (const Npp32sc * pSrc, int nSrcStep, Npp32sc * pDst, int nDstStep, NppiSize oSizeROI)

32-bit complex image copy.

Parameters:

pSrc Source-Image Pointer.
nSrcStep Source-Image Line Step.
pDst Destination-Image Pointer.
nDstStep Destination-Image Line Step.
oSizeROI Region-of-Interest (ROI).

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.7.1.85 NppStatus nppiCopy_32sc_C2R (const Npp32sc * pSrc, int nSrcStep, Npp32sc * pDst, int nDstStep, NppiSize oSizeROI)

Two-channel 32-bit complex image copy.

Parameters:

- pSrc* Source-Image Pointer.
- nSrcStep* Source-Image Line Step.
- pDst* Destination-Image Pointer.
- nDstStep* Destination-Image Line Step.
- oSizeROI* Region-of-Interest (ROI).

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.7.1.86 NppStatus nppiCopy_32sc_C3R (const Npp32sc * pSrc, int nSrcStep, Npp32sc * pDst, int nDstStep, NppiSize oSizeROI)

Three-channel 32-bit complex image copy.

Parameters:

- pSrc* Source-Image Pointer.
- nSrcStep* Source-Image Line Step.
- pDst* Destination-Image Pointer.
- nDstStep* Destination-Image Line Step.
- oSizeROI* Region-of-Interest (ROI).

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.7.1.87 NppStatus nppiCopy_32sc_C4R (const Npp32sc * pSrc, int nSrcStep, Npp32sc * pDst, int nDstStep, NppiSize oSizeROI)

Four-channel 32-bit complex image copy.

Parameters:

- pSrc* Source-Image Pointer.
- nSrcStep* Source-Image Line Step.
- pDst* Destination-Image Pointer.
- nDstStep* Destination-Image Line Step.
- oSizeROI* Region-of-Interest (ROI).

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.7.1.88 NppStatus nppiCopy_8s_AC4R (const Npp8s * pSrc, int nSrcStep, Npp8s * pDst, int nDstStep, NppiSize oSizeROI)

Four-channel 8-bit image copy, ignoring alpha channel.

Parameters:

pSrc Source-Image Pointer.
nSrcStep Source-Image Line Step.
pDst Destination-Image Pointer.
nDstStep Destination-Image Line Step.
oSizeROI Region-of-Interest (ROI).

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.7.1.89 NppStatus nppiCopy_8s_C1R (const Npp8s * pSrc, int nSrcStep, Npp8s * pDst, int nDstStep, NppiSize oSizeROI)

8-bit image copy.

Parameters:

pSrc Source-Image Pointer.
nSrcStep Source-Image Line Step.
pDst Destination-Image Pointer.
nDstStep Destination-Image Line Step.
oSizeROI Region-of-Interest (ROI).

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.7.1.90 NppStatus nppiCopy_8s_C2R (const Npp8s * pSrc, int nSrcStep, Npp8s * pDst, int nDstStep, NppiSize oSizeROI)

Two-channel 8-bit image copy.

Parameters:

pSrc Source-Image Pointer.
nSrcStep Source-Image Line Step.
pDst Destination-Image Pointer.
nDstStep Destination-Image Line Step.
oSizeROI Region-of-Interest (ROI).

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.7.1.91 NppStatus nppiCopy_8s_C3R (const Npp8s * pSrc, int nSrcStep, Npp8s * pDst, int nDstStep, NppiSize oSizeROI)

Three-channel 8-bit image copy.

Parameters:

pSrc Source-Image Pointer.
nSrcStep Source-Image Line Step.
pDst Destination-Image Pointer.
nDstStep Destination-Image Line Step.
oSizeROI Region-of-Interest (ROI).

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.7.1.92 NppStatus nppiCopy_8s_C4R (const Npp8s * pSrc, int nSrcStep, Npp8s * pDst, int nDstStep, NppiSize oSizeROI)

Four-channel 8-bit image copy.

Parameters:

pSrc Source-Image Pointer.
nSrcStep Source-Image Line Step.
pDst Destination-Image Pointer.
nDstStep Destination-Image Line Step.
oSizeROI Region-of-Interest (ROI).

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.7.1.93 NppStatus nppiCopy_8u_AC4MR (const Npp8u * pSrc, int nSrcStep, Npp8u * pDst, int nDstStep, NppiSize oSizeROI, const Npp8u * pMask, int nMaskStep)

Masked Operation four channel 8-bit unsigned image copy, ignoring alpha.

Parameters:

pSrc Source-Image Pointer.
nSrcStep Source-Image Line Step.
pDst Destination-Image Pointer.
nDstStep Destination-Image Line Step.
pMask Mask-Image Pointer.
nMaskStep Mask-Image Line Step.
oSizeROI Region-of-Interest (ROI).

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.7.1.94 NppStatus nppiCopy_8u_AC4R (const Npp8u * pSrc, int nSrcStep, Npp8u * pDst, int nDstStep, NppiSize oSizeROI)

4 channel 8-bit unsigned image copy, not affecting Alpha channel.

Parameters:

pSrc Source-Image Pointer.
nSrcStep Source-Image Line Step.
pDst Destination-Image Pointer.
nDstStep Destination-Image Line Step.
oSizeROI Region-of-Interest (ROI).

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.7.1.95 NppStatus nppiCopy_8u_C1C3R (const Npp8u * pSrc, int nSrcStep, Npp8u * pDst, int nDstStep, NppiSize oSizeROI)

Single-channel to three-channel 8-bit unsigned image copy.

Parameters:

pSrc Source-Image Pointer.
nSrcStep Source-Image Line Step.
pDst Select-Channel Destination-Image Pointer.
nDstStep Destination-Image Line Step.
oSizeROI Region-of-Interest (ROI).

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.7.1.96 NppStatus nppiCopy_8u_C1C4R (const Npp8u * pSrc, int nSrcStep, Npp8u * pDst, int nDstStep, NppiSize oSizeROI)

Single-channel to four-channel 8-bit unsigned image copy.

Parameters:

pSrc Source-Image Pointer.
nSrcStep Source-Image Line Step.
pDst Select-Channel Destination-Image Pointer.
nDstStep Destination-Image Line Step.
oSizeROI Region-of-Interest (ROI).

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.7.1.97 `NppStatus nppiCopy_8u_C1MR (const Npp8u * pSrc, int nSrcStep, Npp8u * pDst, int nDstStep, NppiSize oSizeROI, const Npp8u * pMask, int nMaskStep)`

Masked Operation 8-bit unsigned image copy.

Parameters:

pSrc Source-Image Pointer.
nSrcStep Source-Image Line Step.
pDst Destination-Image Pointer.
nDstStep Destination-Image Line Step.
pMask Mask-Image Pointer.
nMaskStep Mask-Image Line Step.
oSizeROI Region-of-Interest (ROI).

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.7.1.98 `NppStatus nppiCopy_8u_C1R (const Npp8u * pSrc, int nSrcStep, Npp8u * pDst, int nDstStep, NppiSize oSizeROI)`

8-bit unsigned image copy.

Parameters:

pSrc Source-Image Pointer.
nSrcStep Source-Image Line Step.
pDst Destination-Image Pointer.
nDstStep Destination-Image Line Step.
oSizeROI Region-of-Interest (ROI).

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.7.1.99 `NppStatus nppiCopy_8u_C3C1R (const Npp8u * pSrc, int nSrcStep, Npp8u * pDst, int nDstStep, NppiSize oSizeROI)`

Three-channel to single-channel 8-bit unsigned image copy.

Parameters:

pSrc Select-Channel Source-Image Pointer.
nSrcStep Source-Image Line Step.
pDst Destination-Image Pointer.
nDstStep Destination-Image Line Step.
oSizeROI Region-of-Interest (ROI).

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.7.1.100 `NppStatus nppiCopy_8u_C3CR (const Npp8u * pSrc, int nSrcStep, Npp8u * pDst, int nDstStep, NppiSize oSizeROI)`

Select-channel 8-bit unsigned image copy for three-channel images.

Parameters:

pSrc Select-Channel Source-Image Pointer.
nSrcStep Source-Image Line Step.
pDst Select-Channel Destination-Image Pointer.
nDstStep Destination-Image Line Step.
oSizeROI Region-of-Interest (ROI).

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.7.1.101 `NppStatus nppiCopy_8u_C3MR (const Npp8u * pSrc, int nSrcStep, Npp8u * pDst, int nDstStep, NppiSize oSizeROI, const Npp8u * pMask, int nMaskStep)`

Masked Operation three channel 8-bit unsigned image copy.

Parameters:

pSrc Source-Image Pointer.
nSrcStep Source-Image Line Step.
pDst Destination-Image Pointer.
nDstStep Destination-Image Line Step.
pMask Mask-Image Pointer.
nMaskStep Mask-Image Line Step.
oSizeROI Region-of-Interest (ROI).

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.7.1.102 `NppStatus nppiCopy_8u_C3P3R (const Npp8u * pSrc, int nSrcStep, Npp8u *const aDst[3], int nDstStep, NppiSize oSizeROI)`

Three-channel 8-bit unsigned packed to planar image copy.

Parameters:

pSrc Source-Image Pointer.
nSrcStep Source-Image Line Step.
aDst Destination-Planar-Image Pointer Array.
nDstStep Destination-Image Line Step.
oSizeROI Region-of-Interest (ROI).

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.7.1.103 NppStatus nppiCopy_8u_C3R (const Npp8u * pSrc, int nSrcStep, Npp8u * pDst, int nDstStep, NppiSize oSizeROI)

Three channel 8-bit unsigned image copy.

Parameters:

- pSrc* Source-Image Pointer.
- nSrcStep* Source-Image Line Step.
- pDst* Destination-Image Pointer.
- nDstStep* Destination-Image Line Step.
- oSizeROI* Region-of-Interest (ROI).

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.7.1.104 NppStatus nppiCopy_8u_C4C1R (const Npp8u * pSrc, int nSrcStep, Npp8u * pDst, int nDstStep, NppiSize oSizeROI)

Four-channel to single-channel 8-bit unsigned image copy.

Parameters:

- pSrc* Select-Channel Source-Image Pointer.
- nSrcStep* Source-Image Line Step.
- pDst* Destination-Image Pointer.
- nDstStep* Destination-Image Line Step.
- oSizeROI* Region-of-Interest (ROI).

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.7.1.105 NppStatus nppiCopy_8u_C4CR (const Npp8u * pSrc, int nSrcStep, Npp8u * pDst, int nDstStep, NppiSize oSizeROI)

Select-channel 8-bit unsigned image copy for four-channel images.

Parameters:

- pSrc* Select-Channel Source-Image Pointer.
- nSrcStep* Source-Image Line Step.
- pDst* Select-Channel Destination-Image Pointer.
- nDstStep* Destination-Image Line Step.
- oSizeROI* Region-of-Interest (ROI).

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.7.1.106 `NppStatus nppiCopy_8u_C4MR (const Npp8u * pSrc, int nSrcStep, Npp8u * pDst, int nDstStep, NppiSize oSizeROI, const Npp8u * pMask, int nMaskStep)`

[Masked Operation](#) four channel 8-bit unsigned image copy.

Parameters:

pSrc Source-Image Pointer.
nSrcStep Source-Image Line Step.
pDst Destination-Image Pointer.
nDstStep Destination-Image Line Step.
pMask Mask-Image Pointer.
nMaskStep Mask-Image Line Step.
oSizeROI Region-of-Interest (ROI).

Returns:

[Image Data Related Error Codes](#), [ROI Related Error Codes](#)

7.7.1.107 `NppStatus nppiCopy_8u_C4P4R (const Npp8u * pSrc, int nSrcStep, Npp8u *const aDst[4], int nDstStep, NppiSize oSizeROI)`

Four-channel 8-bit unsigned packed to planar image copy.

Parameters:

pSrc Source-Image Pointer.
nSrcStep Source-Image Line Step.
aDst Destination-Planar-Image Pointer Array.
nDstStep Destination-Image Line Step.
oSizeROI Region-of-Interest (ROI).

Returns:

[Image Data Related Error Codes](#), [ROI Related Error Codes](#)

7.7.1.108 `NppStatus nppiCopy_8u_C4R (const Npp8u * pSrc, int nSrcStep, Npp8u * pDst, int nDstStep, NppiSize oSizeROI)`

4 channel 8-bit unsigned image copy.

Parameters:

pSrc Source-Image Pointer.
nSrcStep Source-Image Line Step.
pDst Destination-Image Pointer.
nDstStep Destination-Image Line Step.
oSizeROI Region-of-Interest (ROI).

Returns:

[Image Data Related Error Codes](#), [ROI Related Error Codes](#)

7.7.1.109 `NppStatus nppiCopy_8u_P3C3R (const Npp8u *const aSrc[3], int nSrcStep, Npp8u * pDst, int nDstStep, NppiSize oSizeROI)`

Three-channel 8-bit unsigned planar to packed image copy.

Parameters:

aSrc Planar Source-Image Pointer.
nSrcStep Source-Planar-Image Pointer Array.
pDst Destination-Image Pointer.
nDstStep Destination-Image Line Step.
oSizeROI Region-of-Interest (ROI).

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.7.1.110 `NppStatus nppiCopy_8u_P4C4R (const Npp8u *const aSrc[4], int nSrcStep, Npp8u * pDst, int nDstStep, NppiSize oSizeROI)`

Four-channel 8-bit unsigned planar to packed image copy.

Parameters:

aSrc Planar Source-Planar-Image Pointer Array.
nSrcStep Source-Image Line Step.
pDst Destination-Image Pointer.
nDstStep Destination-Image Line Step.
oSizeROI Region-of-Interest (ROI).

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.8 Convert

Convert to Increase Bit-Depth

The integer conversion methods do not involve any scaling.

Also, even when increasing the bit-depth loss of information may occur:

- When converting integers (e.g. `Npp32u`) to float (e.g. `Npp32f`) integervalue not accurately representable by the float are rounded to the closest floating-point value.
- When converting signed integers to unsigned integers all negative values are lost (saturated to 0).
- `NppStatus nppiConvert_8u16u_C1R` (const `Npp8u` *pSrc, int nSrcStep, `Npp16u` *pDst, int nDstStep, `NppiSize` oSizeROI)
Single channel 8-bit unsigned to 16-bit unsigned conversion.
- `NppStatus nppiConvert_8u16u_C3R` (const `Npp8u` *pSrc, int nSrcStep, `Npp16u` *pDst, int nDstStep, `NppiSize` oSizeROI)
Three channel 8-bit unsigned to 16-bit unsigned conversion.
- `NppStatus nppiConvert_8u16u_C4R` (const `Npp8u` *pSrc, int nSrcStep, `Npp16u` *pDst, int nDstStep, `NppiSize` oSizeROI)
Four channel 8-bit unsigned to 16-bit unsigned conversion.
- `NppStatus nppiConvert_8u16u_AC4R` (const `Npp8u` *pSrc, int nSrcStep, `Npp16u` *pDst, int nDstStep, `NppiSize` oSizeROI)
Four channel 8-bit unsigned to 16-bit unsigned conversion, not affecting Alpha.
- `NppStatus nppiConvert_8u16s_C1R` (const `Npp8u` *pSrc, int nSrcStep, `Npp16s` *pDst, int nDstStep, `NppiSize` oSizeROI)
Single channel 8-bit unsigned to 16-bit signed conversion.
- `NppStatus nppiConvert_8u16s_C3R` (const `Npp8u` *pSrc, int nSrcStep, `Npp16s` *pDst, int nDstStep, `NppiSize` oSizeROI)
Three channel 8-bit unsigned to 16-bit signed conversion.
- `NppStatus nppiConvert_8u16s_C4R` (const `Npp8u` *pSrc, int nSrcStep, `Npp16s` *pDst, int nDstStep, `NppiSize` oSizeROI)
Four channel 8-bit unsigned to 16-bit signed conversion.
- `NppStatus nppiConvert_8u16s_AC4R` (const `Npp8u` *pSrc, int nSrcStep, `Npp16s` *pDst, int nDstStep, `NppiSize` oSizeROI)
Four channel 8-bit unsigned to 16-bit signed conversion, not affecting Alpha.
- `NppStatus nppiConvert_8u32s_C1R` (const `Npp8u` *pSrc, int nSrcStep, `Npp32s` *pDst, int nDstStep, `NppiSize` oSizeROI)
Single channel 8-bit unsigned to 32-bit signed conversion.
- `NppStatus nppiConvert_8u32s_C3R` (const `Npp8u` *pSrc, int nSrcStep, `Npp32s` *pDst, int nDstStep, `NppiSize` oSizeROI)

Three channel 8-bit unsigned to 32-bit signed conversion.

- `NppStatus nppiConvert_8u32s_C4R` (const `Npp8u` *pSrc, int nSrcStep, `Npp32s` *pDst, int nDstStep, `NppiSize` oSizeROI)

Four channel 8-bit unsigned to 32-bit signed conversion.

- `NppStatus nppiConvert_8u32s_AC4R` (const `Npp8u` *pSrc, int nSrcStep, `Npp32s` *pDst, int nDstStep, `NppiSize` oSizeROI)

Four channel 8-bit unsigned to 32-bit signed conversion, not affecting Alpha.

- `NppStatus nppiConvert_8u32f_C1R` (const `Npp8u` *pSrc, int nSrcStep, `Npp32f` *pDst, int nDstStep, `NppiSize` oSizeROI)

Single channel 8-bit unsigned to 32-bit floating-point conversion.

- `NppStatus nppiConvert_8u32f_C3R` (const `Npp8u` *pSrc, int nSrcStep, `Npp32f` *pDst, int nDstStep, `NppiSize` oSizeROI)

Three channel 8-bit unsigned to 32-bit floating-point conversion.

- `NppStatus nppiConvert_8u32f_C4R` (const `Npp8u` *pSrc, int nSrcStep, `Npp32f` *pDst, int nDstStep, `NppiSize` oSizeROI)

Four channel 8-bit unsigned to 32-bit floating-point conversion.

- `NppStatus nppiConvert_8u32f_AC4R` (const `Npp8u` *pSrc, int nSrcStep, `Npp32f` *pDst, int nDstStep, `NppiSize` oSizeROI)

Four channel 8-bit unsigned to 32-bit floating-point conversion, not affecting Alpha.

- `NppStatus nppiConvert_8s32s_C1R` (const `Npp8s` *pSrc, int nSrcStep, `Npp32s` *pDst, int nDstStep, `NppiSize` oSizeROI)

Single channel 8-bit signed to 32-bit signed conversion.

- `NppStatus nppiConvert_8s32s_C3R` (const `Npp8s` *pSrc, int nSrcStep, `Npp32s` *pDst, int nDstStep, `NppiSize` oSizeROI)

Three channel 8-bit signed to 32-bit signed conversion.

- `NppStatus nppiConvert_8s32s_C4R` (const `Npp8s` *pSrc, int nSrcStep, `Npp32s` *pDst, int nDstStep, `NppiSize` oSizeROI)

Four channel 8-bit signed to 32-bit signed conversion.

- `NppStatus nppiConvert_8s32s_AC4R` (const `Npp8s` *pSrc, int nSrcStep, `Npp32s` *pDst, int nDstStep, `NppiSize` oSizeROI)

Four channel 8-bit signed to 32-bit signed conversion, not affecting Alpha.

- `NppStatus nppiConvert_8s32f_C1R` (const `Npp8s` *pSrc, int nSrcStep, `Npp32f` *pDst, int nDstStep, `NppiSize` oSizeROI)

Single channel 8-bit signed to 32-bit floating-point conversion.

- `NppStatus nppiConvert_8s32f_C3R` (const `Npp8s` *pSrc, int nSrcStep, `Npp32f` *pDst, int nDstStep, `NppiSize` oSizeROI)

Three channel 8-bit signed to 32-bit floating-point conversion.

- [NppStatus nppiConvert_8s32f_C4R](#) (const [Npp8s](#) *pSrc, int nSrcStep, [Npp32f](#) *pDst, int nDstStep, [NppiSize](#) oSizeROI)
Four channel 8-bit signed to 32-bit floating-point conversion.
- [NppStatus nppiConvert_8s32f_AC4R](#) (const [Npp8s](#) *pSrc, int nSrcStep, [Npp32f](#) *pDst, int nDstStep, [NppiSize](#) oSizeROI)
Four channel 8-bit signed to 32-bit floating-point conversion, not affecting Alpha.
- [NppStatus nppiConvert_16u32s_C1R](#) (const [Npp16u](#) *pSrc, int nSrcStep, [Npp32s](#) *pDst, int nDstStep, [NppiSize](#) oSizeROI)
Single channel 16-bit unsigned to 32-bit signed conversion.
- [NppStatus nppiConvert_16u32s_C3R](#) (const [Npp16u](#) *pSrc, int nSrcStep, [Npp32s](#) *pDst, int nDstStep, [NppiSize](#) oSizeROI)
Three channel 16-bit unsigned to 32-bit signed conversion.
- [NppStatus nppiConvert_16u32s_C4R](#) (const [Npp16u](#) *pSrc, int nSrcStep, [Npp32s](#) *pDst, int nDstStep, [NppiSize](#) oSizeROI)
Four channel 16-bit unsigned to 32-bit signed conversion.
- [NppStatus nppiConvert_16u32s_AC4R](#) (const [Npp16u](#) *pSrc, int nSrcStep, [Npp32s](#) *pDst, int nDstStep, [NppiSize](#) oSizeROI)
Four channel 16-bit unsigned to 32-bit signed conversion, not affecting Alpha.
- [NppStatus nppiConvert_16u32f_C1R](#) (const [Npp16u](#) *pSrc, int nSrcStep, [Npp32f](#) *pDst, int nDstStep, [NppiSize](#) oSizeROI)
Single channel 16-bit unsigned to 32-bit floating-point conversion.
- [NppStatus nppiConvert_16u32f_C3R](#) (const [Npp16u](#) *pSrc, int nSrcStep, [Npp32f](#) *pDst, int nDstStep, [NppiSize](#) oSizeROI)
Three channel 16-bit unsigned to 32-bit floating-point conversion.
- [NppStatus nppiConvert_16u32f_C4R](#) (const [Npp16u](#) *pSrc, int nSrcStep, [Npp32f](#) *pDst, int nDstStep, [NppiSize](#) oSizeROI)
Four channel 16-bit unsigned to 32-bit floating-point conversion.
- [NppStatus nppiConvert_16u32f_AC4R](#) (const [Npp16u](#) *pSrc, int nSrcStep, [Npp32f](#) *pDst, int nDstStep, [NppiSize](#) oSizeROI)
Four channel 16-bit unsigned to 32-bit floating-point conversion, not affecting Alpha.
- [NppStatus nppiConvert_16s32s_C1R](#) (const [Npp16s](#) *pSrc, int nSrcStep, [Npp32s](#) *pDst, int nDstStep, [NppiSize](#) oSizeROI)
Single channel 16-bit signed to 32-bit signed conversion.
- [NppStatus nppiConvert_16s32s_C3R](#) (const [Npp16s](#) *pSrc, int nSrcStep, [Npp32s](#) *pDst, int nDstStep, [NppiSize](#) oSizeROI)
Three channel 16-bit signed to 32-bit signed conversion.
- [NppStatus nppiConvert_16s32s_C4R](#) (const [Npp16s](#) *pSrc, int nSrcStep, [Npp32s](#) *pDst, int nDstStep, [NppiSize](#) oSizeROI)
Four channel 16-bit signed to 32-bit signed conversion.

- `NppStatus nppiConvert_16s32s_AC4R` (const `Npp16s` *pSrc, int nSrcStep, `Npp32s` *pDst, int nDstStep, `NppiSize` oSizeROI)
Four channel 16-bit signed to 32-bit signed conversion, not affecting Alpha.
- `NppStatus nppiConvert_16s32f_C1R` (const `Npp16s` *pSrc, int nSrcStep, `Npp32f` *pDst, int nDstStep, `NppiSize` oSizeROI)
Single channel 16-bit signed to 32-bit floating-point conversion.
- `NppStatus nppiConvert_16s32f_C3R` (const `Npp16s` *pSrc, int nSrcStep, `Npp32f` *pDst, int nDstStep, `NppiSize` oSizeROI)
Three channel 16-bit signed to 32-bit floating-point conversion.
- `NppStatus nppiConvert_16s32f_C4R` (const `Npp16s` *pSrc, int nSrcStep, `Npp32f` *pDst, int nDstStep, `NppiSize` oSizeROI)
Four channel 16-bit signed to 32-bit floating-point conversion.
- `NppStatus nppiConvert_16s32f_AC4R` (const `Npp16s` *pSrc, int nSrcStep, `Npp32f` *pDst, int nDstStep, `NppiSize` oSizeROI)
Four channel 16-bit signed to 32-bit floating-point conversion, not affecting Alpha.
- `NppStatus nppiConvert_8s8u_C1Rs` (const `Npp8s` *pSrc, int nSrcStep, `Npp8u` *pDst, int nDstStep, `NppiSize` oSizeROI)
Single channel 8-bit signed to 8-bit unsigned conversion with saturation.
- `NppStatus nppiConvert_8s16u_C1Rs` (const `Npp8s` *pSrc, int nSrcStep, `Npp16u` *pDst, int nDstStep, `NppiSize` oSizeROI)
Single channel 8-bit signed to 16-bit unsigned conversion with saturation.
- `NppStatus nppiConvert_8s16s_C1R` (const `Npp8s` *pSrc, int nSrcStep, `Npp16s` *pDst, int nDstStep, `NppiSize` oSizeROI)
Single channel 8-bit signed to 16-bit signed conversion.
- `NppStatus nppiConvert_8s32u_C1Rs` (const `Npp8s` *pSrc, int nSrcStep, `Npp32u` *pDst, int nDstStep, `NppiSize` oSizeROI)
Single channel 8-bit signed to 32-bit unsigned conversion with saturation.
- `NppStatus nppiConvert_16s16u_C1Rs` (const `Npp16s` *pSrc, int nSrcStep, `Npp16u` *pDst, int nDstStep, `NppiSize` oSizeROI)
Single channel 16-bit signed to 16-bit unsigned conversion with saturation.
- `NppStatus nppiConvert_16s32u_C1Rs` (const `Npp16s` *pSrc, int nSrcStep, `Npp32u` *pDst, int nDstStep, `NppiSize` oSizeROI)
Single channel 16-bit signed to 32-bit unsigned conversion with saturation.
- `NppStatus nppiConvert_16u32u_C1R` (const `Npp16u` *pSrc, int nSrcStep, `Npp32u` *pDst, int nDstStep, `NppiSize` oSizeROI)
Single channel 16-bit unsigned to 32-bit unsigned conversion.
- `NppStatus nppiConvert_32s32u_C1Rs` (const `Npp32s` *pSrc, int nSrcStep, `Npp32u` *pDst, int nDstStep, `NppiSize` oSizeROI)

Single channel 32-bit signed to 32-bit unsigned conversion with saturation.

- `NppStatus nppiConvert_32s32f_C1R` (const `Npp32s` *pSrc, int nSrcStep, `Npp32f` *pDst, int nDstStep, `NppiSize` oSizeROI)

Single channel 32-bit signed to 32-bit floating-point conversion.

- `NppStatus nppiConvert_32u32f_C1R` (const `Npp32u` *pSrc, int nSrcStep, `Npp32f` *pDst, int nDstStep, `NppiSize` oSizeROI)

Single channel 32-bit unsigned to 32-bit floating-point conversion.

Convert to Decrease Bit-Depth

The integer conversion methods do not involve any scaling.

When converting floating-point values to integers the user may choose the most appropriate rounding-mode. Typically information is lost when converting to lower bit depth:

- All converted values are saturated to the destination type's range. E.g. any values larger than the largest value of the destination type are clamped to the destination's maximum.
- Converting floating-point values to integer also involves rounding, effectively losing all fractional value information in the process.

- `NppStatus nppiConvert_16u8u_C1R` (const `Npp16u` *pSrc, int nSrcStep, `Npp8u` *pDst, int nDstStep, `NppiSize` oSizeROI)

Single channel 16-bit unsigned to 8-bit unsigned conversion.

- `NppStatus nppiConvert_16u8u_C3R` (const `Npp16u` *pSrc, int nSrcStep, `Npp8u` *pDst, int nDstStep, `NppiSize` oSizeROI)

Three channel 16-bit unsigned to 8-bit unsigned conversion.

- `NppStatus nppiConvert_16u8u_C4R` (const `Npp16u` *pSrc, int nSrcStep, `Npp8u` *pDst, int nDstStep, `NppiSize` oSizeROI)

Four channel 16-bit unsigned to 8-bit unsigned conversion.

- `NppStatus nppiConvert_16u8u_AC4R` (const `Npp16u` *pSrc, int nSrcStep, `Npp8u` *pDst, int nDstStep, `NppiSize` oSizeROI)

Four channel 16-bit unsigned to 8-bit unsigned conversion, not affecting Alpha.

- `NppStatus nppiConvert_16s8u_C1R` (const `Npp16s` *pSrc, int nSrcStep, `Npp8u` *pDst, int nDstStep, `NppiSize` oSizeROI)

Single channel 16-bit signed to 8-bit unsigned conversion.

- `NppStatus nppiConvert_16s8u_C3R` (const `Npp16s` *pSrc, int nSrcStep, `Npp8u` *pDst, int nDstStep, `NppiSize` oSizeROI)

Three channel 16-bit signed to 8-bit unsigned conversion.

- `NppStatus nppiConvert_16s8u_C4R` (const `Npp16s` *pSrc, int nSrcStep, `Npp8u` *pDst, int nDstStep, `NppiSize` oSizeROI)

Four channel 16-bit signed to 8-bit unsigned conversion.

- `NppStatus nppiConvert_16s8u_AC4R` (const `Npp16s` *pSrc, int nSrcStep, `Npp8u` *pDst, int nDstStep, `NppiSize` oSizeROI)
Four channel 16-bit signed to 8-bit unsigned conversion, not affecting Alpha.
- `NppStatus nppiConvert_32s8u_C1R` (const `Npp32s` *pSrc, int nSrcStep, `Npp8u` *pDst, int nDstStep, `NppiSize` oSizeROI)
Single channel 32-bit signed to 8-bit unsigned conversion.
- `NppStatus nppiConvert_32s8u_C3R` (const `Npp32s` *pSrc, int nSrcStep, `Npp8u` *pDst, int nDstStep, `NppiSize` oSizeROI)
Three channel 32-bit signed to 8-bit unsigned conversion.
- `NppStatus nppiConvert_32s8u_C4R` (const `Npp32s` *pSrc, int nSrcStep, `Npp8u` *pDst, int nDstStep, `NppiSize` oSizeROI)
Four channel 32-bit signed to 8-bit unsigned conversion.
- `NppStatus nppiConvert_32s8u_AC4R` (const `Npp32s` *pSrc, int nSrcStep, `Npp8u` *pDst, int nDstStep, `NppiSize` oSizeROI)
Four channel 32-bit signed to 8-bit unsigned conversion, not affecting Alpha.
- `NppStatus nppiConvert_32s8s_C1R` (const `Npp32s` *pSrc, int nSrcStep, `Npp8s` *pDst, int nDstStep, `NppiSize` oSizeROI)
Single channel 32-bit signed to 8-bit signed conversion.
- `NppStatus nppiConvert_32s8s_C3R` (const `Npp32s` *pSrc, int nSrcStep, `Npp8s` *pDst, int nDstStep, `NppiSize` oSizeROI)
Three channel 32-bit signed to 8-bit signed conversion.
- `NppStatus nppiConvert_32s8s_C4R` (const `Npp32s` *pSrc, int nSrcStep, `Npp8s` *pDst, int nDstStep, `NppiSize` oSizeROI)
Four channel 32-bit signed to 8-bit signed conversion.
- `NppStatus nppiConvert_32s8s_AC4R` (const `Npp32s` *pSrc, int nSrcStep, `Npp8s` *pDst, int nDstStep, `NppiSize` oSizeROI)
Four channel 32-bit signed to 8-bit signed conversion, not affecting Alpha.
- `NppStatus nppiConvert_8u8s_C1RSfs` (const `Npp8u` *pSrc, int nSrcStep, `Npp8s` *pDst, int nDstStep, `NppiSize` oSizeROI, `NppRoundMode` eRoundMode, int nScaleFactor)
Single channel 8-bit unsigned to 8-bit signed conversion.
- `NppStatus nppiConvert_16u8s_C1RSfs` (const `Npp16u` *pSrc, int nSrcStep, `Npp8s` *pDst, int nDstStep, `NppiSize` oSizeROI, `NppRoundMode` eRoundMode, int nScaleFactor)
Single channel 16-bit unsigned to 8-bit signed conversion.
- `NppStatus nppiConvert_16s8s_C1RSfs` (const `Npp16s` *pSrc, int nSrcStep, `Npp8s` *pDst, int nDstStep, `NppiSize` oSizeROI, `NppRoundMode` eRoundMode, int nScaleFactor)
Single channel 16-bit signed to 8-bit signed conversion.
- `NppStatus nppiConvert_16u16s_C1RSfs` (const `Npp16u` *pSrc, int nSrcStep, `Npp16s` *pDst, int nDstStep, `NppiSize` oSizeROI, `NppRoundMode` eRoundMode, int nScaleFactor)

Single channel 16-bit unsigned to 16-bit signed conversion.

- `NppStatus nppiConvert_32u8u_C1RSfs` (const `Npp32u` *pSrc, int nSrcStep, `Npp8u` *pDst, int nDstStep, `NppiSize` oSizeROI, `NppRoundMode` eRoundMode, int nScaleFactor)

Single channel 32-bit unsigned to 8-bit unsigned conversion.

- `NppStatus nppiConvert_32u8s_C1RSfs` (const `Npp32u` *pSrc, int nSrcStep, `Npp8s` *pDst, int nDstStep, `NppiSize` oSizeROI, `NppRoundMode` eRoundMode, int nScaleFactor)

Single channel 32-bit unsigned to 8-bit signed conversion.

- `NppStatus nppiConvert_32u16u_C1RSfs` (const `Npp32u` *pSrc, int nSrcStep, `Npp16u` *pDst, int nDstStep, `NppiSize` oSizeROI, `NppRoundMode` eRoundMode, int nScaleFactor)

Single channel 32-bit unsigned to 16-bit unsigned conversion.

- `NppStatus nppiConvert_32u16s_C1RSfs` (const `Npp32u` *pSrc, int nSrcStep, `Npp16s` *pDst, int nDstStep, `NppiSize` oSizeROI, `NppRoundMode` eRoundMode, int nScaleFactor)

Single channel 32-bit unsigned to 16-bit signed conversion.

- `NppStatus nppiConvert_32u32s_C1RSfs` (const `Npp32u` *pSrc, int nSrcStep, `Npp32s` *pDst, int nDstStep, `NppiSize` oSizeROI, `NppRoundMode` eRoundMode, int nScaleFactor)

Single channel 32-bit unsigned to 32-bit signed conversion.

- `NppStatus nppiConvert_32s16u_C1RSfs` (const `Npp32s` *pSrc, int nSrcStep, `Npp16u` *pDst, int nDstStep, `NppiSize` oSizeROI, `NppRoundMode` eRoundMode, int nScaleFactor)

Single channel 32-bit unsigned to 16-bit unsigned conversion.

- `NppStatus nppiConvert_32s16s_C1RSfs` (const `Npp32s` *pSrc, int nSrcStep, `Npp16s` *pDst, int nDstStep, `NppiSize` oSizeROI, `NppRoundMode` eRoundMode, int nScaleFactor)

Single channel 32-bit unsigned to 16-bit signed conversion.

- `NppStatus nppiConvert_32f8u_C1R` (const `Npp32f` *pSrc, int nSrcStep, `Npp8u` *pDst, int nDstStep, `NppiSize` oSizeROI, `NppRoundMode` eRoundMode)

Single channel 32-bit floating point to 8-bit unsigned conversion.

- `NppStatus nppiConvert_32f8u_C3R` (const `Npp32f` *pSrc, int nSrcStep, `Npp8u` *pDst, int nDstStep, `NppiSize` oSizeROI, `NppRoundMode` eRoundMode)

Three channel 32-bit floating point to 8-bit unsigned conversion.

- `NppStatus nppiConvert_32f8u_C4R` (const `Npp32f` *pSrc, int nSrcStep, `Npp8u` *pDst, int nDstStep, `NppiSize` oSizeROI, `NppRoundMode` eRoundMode)

Four channel 32-bit floating point to 8-bit unsigned conversion.

- `NppStatus nppiConvert_32f8u_AC4R` (const `Npp32f` *pSrc, int nSrcStep, `Npp8u` *pDst, int nDstStep, `NppiSize` oSizeROI, `NppRoundMode` eRoundMode)

Four channel 32-bit floating point to 8-bit unsigned conversion, not affecting Alpha.

- `NppStatus nppiConvert_32f8s_C1R` (const `Npp32f` *pSrc, int nSrcStep, `Npp8s` *pDst, int nDstStep, `NppiSize` oSizeROI, `NppRoundMode` eRoundMode)

Single channel 32-bit floating point to 8-bit signed conversion.

- `NppStatus nppiConvert_32f8s_C3R` (const `Npp32f *pSrc`, int `nSrcStep`, `Npp8s *pDst`, int `nDstStep`, `NppiSize oSizeROI`, `NppRoundMode eRoundMode`)
Three channel 32-bit floating point to 8-bit signed conversion.
- `NppStatus nppiConvert_32f8s_C4R` (const `Npp32f *pSrc`, int `nSrcStep`, `Npp8s *pDst`, int `nDstStep`, `NppiSize oSizeROI`, `NppRoundMode eRoundMode`)
Four channel 32-bit floating point to 8-bit signed conversion.
- `NppStatus nppiConvert_32f8s_AC4R` (const `Npp32f *pSrc`, int `nSrcStep`, `Npp8s *pDst`, int `nDstStep`, `NppiSize oSizeROI`, `NppRoundMode eRoundMode`)
Four channel 32-bit floating point to 8-bit signed conversion, not affecting Alpha.
- `NppStatus nppiConvert_32f16u_C1R` (const `Npp32f *pSrc`, int `nSrcStep`, `Npp16u *pDst`, int `nDstStep`, `NppiSize oSizeROI`, `NppRoundMode eRoundMode`)
Single channel 32-bit floating point to 16-bit unsigned conversion.
- `NppStatus nppiConvert_32f16u_C3R` (const `Npp32f *pSrc`, int `nSrcStep`, `Npp16u *pDst`, int `nDstStep`, `NppiSize oSizeROI`, `NppRoundMode eRoundMode`)
Three channel 32-bit floating point to 16-bit unsigned conversion.
- `NppStatus nppiConvert_32f16u_C4R` (const `Npp32f *pSrc`, int `nSrcStep`, `Npp16u *pDst`, int `nDstStep`, `NppiSize oSizeROI`, `NppRoundMode eRoundMode`)
Four channel 32-bit floating point to 16-bit unsigned conversion.
- `NppStatus nppiConvert_32f16u_AC4R` (const `Npp32f *pSrc`, int `nSrcStep`, `Npp16u *pDst`, int `nDstStep`, `NppiSize oSizeROI`, `NppRoundMode eRoundMode`)
Four channel 32-bit floating point to 16-bit unsigned conversion, not affecting Alpha.
- `NppStatus nppiConvert_32f16s_C1R` (const `Npp32f *pSrc`, int `nSrcStep`, `Npp16s *pDst`, int `nDstStep`, `NppiSize oSizeROI`, `NppRoundMode eRoundMode`)
Single channel 32-bit floating point to 16-bit signed conversion.
- `NppStatus nppiConvert_32f16s_C3R` (const `Npp32f *pSrc`, int `nSrcStep`, `Npp16s *pDst`, int `nDstStep`, `NppiSize oSizeROI`, `NppRoundMode eRoundMode`)
Three channel 32-bit floating point to 16-bit signed conversion.
- `NppStatus nppiConvert_32f16s_C4R` (const `Npp32f *pSrc`, int `nSrcStep`, `Npp16s *pDst`, int `nDstStep`, `NppiSize oSizeROI`, `NppRoundMode eRoundMode`)
Four channel 32-bit floating point to 16-bit signed conversion.
- `NppStatus nppiConvert_32f16s_AC4R` (const `Npp32f *pSrc`, int `nSrcStep`, `Npp16s *pDst`, int `nDstStep`, `NppiSize oSizeROI`, `NppRoundMode eRoundMode`)
Four channel 32-bit floating point to 16-bit signed conversion.
- `NppStatus nppiConvert_32f8u_C1RSfs` (const `Npp32f *pSrc`, int `nSrcStep`, `Npp8u *pDst`, int `nDstStep`, `NppiSize oSizeROI`, `NppRoundMode eRoundMode`, int `nScaleFactor`)
Single channel 32-bit floating point to 8-bit unsigned conversion.
- `NppStatus nppiConvert_32f8s_C1RSfs` (const `Npp32f *pSrc`, int `nSrcStep`, `Npp8s *pDst`, int `nDstStep`, `NppiSize oSizeROI`, `NppRoundMode eRoundMode`, int `nScaleFactor`)
Single channel 32-bit floating point to 8-bit signed conversion.

- `NppStatus nppiConvert_32f16u_C1RSfs` (const `Npp32f *pSrc`, int `nSrcStep`, `Npp16u *pDst`, int `nDstStep`, `NppiSize oSizeROI`, `NppRoundMode eRoundMode`, int `nScaleFactor`)
Single channel 32-bit floating point to 16-bit unsigned conversion.
- `NppStatus nppiConvert_32f16s_C1RSfs` (const `Npp32f *pSrc`, int `nSrcStep`, `Npp16s *pDst`, int `nDstStep`, `NppiSize oSizeROI`, `NppRoundMode eRoundMode`, int `nScaleFactor`)
Single channel 32-bit floating point to 16-bit signed conversion.
- `NppStatus nppiConvert_32f32u_C1RSfs` (const `Npp32f *pSrc`, int `nSrcStep`, `Npp32u *pDst`, int `nDstStep`, `NppiSize oSizeROI`, `NppRoundMode eRoundMode`, int `nScaleFactor`)
Single channel 32-bit floating point to 32-bit unsigned conversion.
- `NppStatus nppiConvert_32f32s_C1RSfs` (const `Npp32f *pSrc`, int `nSrcStep`, `Npp32s *pDst`, int `nDstStep`, `NppiSize oSizeROI`, `NppRoundMode eRoundMode`, int `nScaleFactor`)
Single channel 32-bit floating point to 32-bit signed conversion.

7.8.1 Function Documentation

7.8.1.1 `NppStatus nppiConvert_16s16u_C1Rs` (const `Npp16s *pSrc`, int `nSrcStep`, `Npp16u *pDst`, int `nDstStep`, `NppiSize oSizeROI`)

Single channel 16-bit signed to 16-bit unsigned conversion with saturation.

Parameters:

- `pSrc` Source-Image Pointer.
- `nSrcStep` Source-Image Line Step.
- `pDst` Destination-Image Pointer.
- `nDstStep` Destination-Image Line Step.
- `oSizeROI` Region-of-Interest (ROI).

Returns:

[Image Data Related Error Codes](#), [ROI Related Error Codes](#)

7.8.1.2 `NppStatus nppiConvert_16s32f_AC4R` (const `Npp16s *pSrc`, int `nSrcStep`, `Npp32f *pDst`, int `nDstStep`, `NppiSize oSizeROI`)

Four channel 16-bit signed to 32-bit floating-point conversion, not affecting Alpha.

Parameters:

- `pSrc` Source-Image Pointer.
- `nSrcStep` Source-Image Line Step.
- `pDst` Destination-Image Pointer.
- `nDstStep` Destination-Image Line Step.
- `oSizeROI` Region-of-Interest (ROI).

Returns:

[Image Data Related Error Codes](#), [ROI Related Error Codes](#)

7.8.1.3 NppStatus nppiConvert_16s32f_C1R (const Npp16s * pSrc, int nSrcStep, Npp32f * pDst, int nDstStep, NppiSize oSizeROI)

Single channel 16-bit signed to 32-bit floating-point conversion.

Parameters:

pSrc Source-Image Pointer.
nSrcStep Source-Image Line Step.
pDst Destination-Image Pointer.
nDstStep Destination-Image Line Step.
oSizeROI Region-of-Interest (ROI).

Returns:

[Image Data Related Error Codes](#), [ROI Related Error Codes](#)

7.8.1.4 NppStatus nppiConvert_16s32f_C3R (const Npp16s * pSrc, int nSrcStep, Npp32f * pDst, int nDstStep, NppiSize oSizeROI)

Three channel 16-bit signed to 32-bit floating-point conversion.

Parameters:

pSrc Source-Image Pointer.
nSrcStep Source-Image Line Step.
pDst Destination-Image Pointer.
nDstStep Destination-Image Line Step.
oSizeROI Region-of-Interest (ROI).

Returns:

[Image Data Related Error Codes](#), [ROI Related Error Codes](#)

7.8.1.5 NppStatus nppiConvert_16s32f_C4R (const Npp16s * pSrc, int nSrcStep, Npp32f * pDst, int nDstStep, NppiSize oSizeROI)

Four channel 16-bit signed to 32-bit floating-point conversion.

Parameters:

pSrc Source-Image Pointer.
nSrcStep Source-Image Line Step.
pDst Destination-Image Pointer.
nDstStep Destination-Image Line Step.

oSizeROI Region-of-Interest (ROI).

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.8.1.6 NppStatus nppiConvert_16s32s_AC4R (const Npp16s * pSrc, int nSrcStep, Npp32s * pDst, int nDstStep, NppiSize oSizeROI)

Four channel 16-bit signed to 32-bit signed conversion, not affecting Alpha.

Parameters:

pSrc Source-Image Pointer.
nSrcStep Source-Image Line Step.
pDst Destination-Image Pointer.
nDstStep Destination-Image Line Step.
oSizeROI Region-of-Interest (ROI).

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.8.1.7 NppStatus nppiConvert_16s32s_C1R (const Npp16s * pSrc, int nSrcStep, Npp32s * pDst, int nDstStep, NppiSize oSizeROI)

Single channel 16-bit signed to 32-bit signed conversion.

Parameters:

pSrc Source-Image Pointer.
nSrcStep Source-Image Line Step.
pDst Destination-Image Pointer.
nDstStep Destination-Image Line Step.
oSizeROI Region-of-Interest (ROI).

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.8.1.8 NppStatus nppiConvert_16s32s_C3R (const Npp16s * pSrc, int nSrcStep, Npp32s * pDst, int nDstStep, NppiSize oSizeROI)

Three channel 16-bit signed to 32-bit signed conversion.

Parameters:

pSrc Source-Image Pointer.
nSrcStep Source-Image Line Step.

pDst Destination-Image Pointer.
nDstStep Destination-Image Line Step.
oSizeROI Region-of-Interest (ROI).

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.8.1.9 NppStatus nppiConvert_16s32s_C4R (const Npp16s * pSrc, int nSrcStep, Npp32s * pDst, int nDstStep, NppiSize oSizeROI)

Four channel 16-bit signed to 32-bit signed conversion.

Parameters:

pSrc Source-Image Pointer.
nSrcStep Source-Image Line Step.
pDst Destination-Image Pointer.
nDstStep Destination-Image Line Step.
oSizeROI Region-of-Interest (ROI).

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.8.1.10 NppStatus nppiConvert_16s32u_C1Rs (const Npp16s * pSrc, int nSrcStep, Npp32u * pDst, int nDstStep, NppiSize oSizeROI)

Single channel 16-bit signed to 32-bit unsigned conversion with saturation.

Parameters:

pSrc Source-Image Pointer.
nSrcStep Source-Image Line Step.
pDst Destination-Image Pointer.
nDstStep Destination-Image Line Step.
oSizeROI Region-of-Interest (ROI).

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.8.1.11 NppStatus nppiConvert_16s8s_C1RSfs (const Npp16s * pSrc, int nSrcStep, Npp8s * pDst, int nDstStep, NppiSize oSizeROI, NppRoundMode eRoundMode, int nScaleFactor)

Single channel 16-bit signed to 8-bit signed conversion.

Parameters:

pSrc Source-Image Pointer.

nSrcStep Source-Image Line Step.
pDst Destination-Image Pointer.
nDstStep Destination-Image Line Step.
oSizeROI Region-of-Interest (ROI).
eRoundMode Rounding Mode Parameter.
nScaleFactor Integer Result Scaling.

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.8.1.12 **NppStatus nppiConvert_16s8u_AC4R** (const Npp16s * *pSrc*, int *nSrcStep*, Npp8u * *pDst*, int *nDstStep*, NppiSize *oSizeROI*)

Four channel 16-bit signed to 8-bit unsigned conversion, not affecting Alpha.

Parameters:

pSrc Source-Image Pointer.
nSrcStep Source-Image Line Step.
pDst Destination-Image Pointer.
nDstStep Destination-Image Line Step.
oSizeROI Region-of-Interest (ROI).

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.8.1.13 **NppStatus nppiConvert_16s8u_C1R** (const Npp16s * *pSrc*, int *nSrcStep*, Npp8u * *pDst*, int *nDstStep*, NppiSize *oSizeROI*)

Single channel 16-bit signed to 8-bit unsigned conversion.

Parameters:

pSrc Source-Image Pointer.
nSrcStep Source-Image Line Step.
pDst Destination-Image Pointer.
nDstStep Destination-Image Line Step.
oSizeROI Region-of-Interest (ROI).

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.8.1.14 NppStatus nppiConvert_16s8u_C3R (const Npp16s * pSrc, int nSrcStep, Npp8u * pDst, int nDstStep, NppiSize oSizeROI)

Three channel 16-bit signed to 8-bit unsigned conversion.

Parameters:

pSrc Source-Image Pointer.
nSrcStep Source-Image Line Step.
pDst Destination-Image Pointer.
nDstStep Destination-Image Line Step.
oSizeROI Region-of-Interest (ROI).

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.8.1.15 NppStatus nppiConvert_16s8u_C4R (const Npp16s * pSrc, int nSrcStep, Npp8u * pDst, int nDstStep, NppiSize oSizeROI)

Four channel 16-bit signed to 8-bit unsigned conversion.

Parameters:

pSrc Source-Image Pointer.
nSrcStep Source-Image Line Step.
pDst Destination-Image Pointer.
nDstStep Destination-Image Line Step.
oSizeROI Region-of-Interest (ROI).

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.8.1.16 NppStatus nppiConvert_16u16s_C1RSfs (const Npp16u * pSrc, int nSrcStep, Npp16s * pDst, int nDstStep, NppiSize oSizeROI, NppRoundMode eRoundMode, int nScaleFactor)

Single channel 16-bit unsigned to 16-bit signed conversion.

Parameters:

pSrc Source-Image Pointer.
nSrcStep Source-Image Line Step.
pDst Destination-Image Pointer.
nDstStep Destination-Image Line Step.
oSizeROI Region-of-Interest (ROI).
eRoundMode Rounding Mode Parameter.
nScaleFactor Integer Result Scaling.

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.8.1.17 NppStatus nppiConvert_16u32f_AC4R (const Npp16u * pSrc, int nSrcStep, Npp32f * pDst, int nDstStep, NppiSize oSizeROI)

Four channel 16-bit unsigned to 32-bit floating-point conversion, not affecting Alpha.

Parameters:

pSrc Source-Image Pointer.
nSrcStep Source-Image Line Step.
pDst Destination-Image Pointer.
nDstStep Destination-Image Line Step.
oSizeROI Region-of-Interest (ROI).

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.8.1.18 NppStatus nppiConvert_16u32f_C1R (const Npp16u * pSrc, int nSrcStep, Npp32f * pDst, int nDstStep, NppiSize oSizeROI)

Single channel 16-bit unsigned to 32-bit floating-point conversion.

Parameters:

pSrc Source-Image Pointer.
nSrcStep Source-Image Line Step.
pDst Destination-Image Pointer.
nDstStep Destination-Image Line Step.
oSizeROI Region-of-Interest (ROI).

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.8.1.19 NppStatus nppiConvert_16u32f_C3R (const Npp16u * pSrc, int nSrcStep, Npp32f * pDst, int nDstStep, NppiSize oSizeROI)

Three channel 16-bit unsigned to 32-bit floating-point conversion.

Parameters:

pSrc Source-Image Pointer.
nSrcStep Source-Image Line Step.
pDst Destination-Image Pointer.
nDstStep Destination-Image Line Step.
oSizeROI Region-of-Interest (ROI).

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.8.1.20 NppStatus nppiConvert_16u32f_C4R (const Npp16u * pSrc, int nSrcStep, Npp32f * pDst, int nDstStep, NppiSize oSizeROI)

Four channel 16-bit unsigned to 32-bit floating-point conversion.

Parameters:

pSrc Source-Image Pointer.
nSrcStep Source-Image Line Step.
pDst Destination-Image Pointer.
nDstStep Destination-Image Line Step.
oSizeROI Region-of-Interest (ROI).

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.8.1.21 NppStatus nppiConvert_16u32s_AC4R (const Npp16u * pSrc, int nSrcStep, Npp32s * pDst, int nDstStep, NppiSize oSizeROI)

Four channel 16-bit unsigned to 32-bit signed conversion, not affecting Alpha.

Parameters:

pSrc Source-Image Pointer.
nSrcStep Source-Image Line Step.
pDst Destination-Image Pointer.
nDstStep Destination-Image Line Step.
oSizeROI Region-of-Interest (ROI).

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.8.1.22 NppStatus nppiConvert_16u32s_C1R (const Npp16u * pSrc, int nSrcStep, Npp32s * pDst, int nDstStep, NppiSize oSizeROI)

Single channel 16-bit unsigned to 32-bit signed conversion.

Parameters:

pSrc Source-Image Pointer.
nSrcStep Source-Image Line Step.
pDst Destination-Image Pointer.
nDstStep Destination-Image Line Step.
oSizeROI Region-of-Interest (ROI).

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.8.1.23 NppStatus nppiConvert_16u32s_C3R (const Npp16u * pSrc, int nSrcStep, Npp32s * pDst, int nDstStep, NppiSize oSizeROI)

Three channel 16-bit unsigned to 32-bit signed conversion.

Parameters:

pSrc Source-Image Pointer.
nSrcStep Source-Image Line Step.
pDst Destination-Image Pointer.
nDstStep Destination-Image Line Step.
oSizeROI Region-of-Interest (ROI).

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.8.1.24 NppStatus nppiConvert_16u32s_C4R (const Npp16u * pSrc, int nSrcStep, Npp32s * pDst, int nDstStep, NppiSize oSizeROI)

Four channel 16-bit unsigned to 32-bit signed conversion.

Parameters:

pSrc Source-Image Pointer.
nSrcStep Source-Image Line Step.
pDst Destination-Image Pointer.
nDstStep Destination-Image Line Step.
oSizeROI Region-of-Interest (ROI).

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.8.1.25 NppStatus nppiConvert_16u32u_C1R (const Npp16u * pSrc, int nSrcStep, Npp32u * pDst, int nDstStep, NppiSize oSizeROI)

Single channel 16-bit unsigned to 32-bit unsigned conversion.

Parameters:

pSrc Source-Image Pointer.
nSrcStep Source-Image Line Step.
pDst Destination-Image Pointer.
nDstStep Destination-Image Line Step.
oSizeROI Region-of-Interest (ROI).

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.8.1.26 NppStatus nppiConvert_16u8s_C1RSfs (const Npp16u * *pSrc*, int *nSrcStep*, Npp8s * *pDst*, int *nDstStep*, NppiSize *oSizeROI*, NppRoundMode *eRoundMode*, int *nScaleFactor*)

Single channel 16-bit unsigned to 8-bit signed conversion.

Parameters:

pSrc Source-Image Pointer.
nSrcStep Source-Image Line Step.
pDst Destination-Image Pointer.
nDstStep Destination-Image Line Step.
oSizeROI Region-of-Interest (ROI).
eRoundMode Rounding Mode Parameter.
nScaleFactor Integer Result Scaling.

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.8.1.27 NppStatus nppiConvert_16u8u_AC4R (const Npp16u * *pSrc*, int *nSrcStep*, Npp8u * *pDst*, int *nDstStep*, NppiSize *oSizeROI*)

Four channel 16-bit unsigned to 8-bit unsigned conversion, not affecting Alpha.

Parameters:

pSrc Source-Image Pointer.
nSrcStep Source-Image Line Step.
pDst Destination-Image Pointer.
nDstStep Destination-Image Line Step.
oSizeROI Region-of-Interest (ROI).

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.8.1.28 NppStatus nppiConvert_16u8u_C1R (const Npp16u * *pSrc*, int *nSrcStep*, Npp8u * *pDst*, int *nDstStep*, NppiSize *oSizeROI*)

Single channel 16-bit unsigned to 8-bit unsigned conversion.

Parameters:

pSrc Source-Image Pointer.
nSrcStep Source-Image Line Step.
pDst Destination-Image Pointer.
nDstStep Destination-Image Line Step.
oSizeROI Region-of-Interest (ROI).

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.8.1.29 NppStatus nppiConvert_16u8u_C3R (const Npp16u * pSrc, int nSrcStep, Npp8u * pDst, int nDstStep, NppiSize oSizeROI)

Three channel 16-bit unsigned to 8-bit unsigned conversion.

Parameters:

pSrc Source-Image Pointer.
nSrcStep Source-Image Line Step.
pDst Destination-Image Pointer.
nDstStep Destination-Image Line Step.
oSizeROI Region-of-Interest (ROI).

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.8.1.30 NppStatus nppiConvert_16u8u_C4R (const Npp16u * pSrc, int nSrcStep, Npp8u * pDst, int nDstStep, NppiSize oSizeROI)

Four channel 16-bit unsigned to 8-bit unsigned conversion.

Parameters:

pSrc Source-Image Pointer.
nSrcStep Source-Image Line Step.
pDst Destination-Image Pointer.
nDstStep Destination-Image Line Step.
oSizeROI Region-of-Interest (ROI).

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.8.1.31 NppStatus nppiConvert_32f16s_AC4R (const Npp32f * pSrc, int nSrcStep, Npp16s * pDst, int nDstStep, NppiSize oSizeROI, NppRoundMode eRoundMode)

Four channel 32-bit floating point to 16-bit signed conversion.

Parameters:

pSrc Source-Image Pointer.
nSrcStep Source-Image Line Step.
pDst Destination-Image Pointer.
nDstStep Destination-Image Line Step.
oSizeROI Region-of-Interest (ROI).
eRoundMode Flag specifying how fractional float values are rounded to integer values.

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.8.1.32 NppStatus nppiConvert_32f16s_C1R (const Npp32f * pSrc, int nSrcStep, Npp16s * pDst, int nDstStep, NppiSize oSizeROI, NppRoundMode eRoundMode)

Single channel 32-bit floating point to 16-bit signed conversion.

Parameters:

pSrc Source-Image Pointer.

nSrcStep Source-Image Line Step.

pDst Destination-Image Pointer.

nDstStep Destination-Image Line Step.

oSizeROI Region-of-Interest (ROI).

eRoundMode Flag specifying how fractional float values are rounded to integer values.

Returns:

[Image Data Related Error Codes](#), [ROI Related Error Codes](#)

7.8.1.33 NppStatus nppiConvert_32f16s_C1RSfs (const Npp32f * pSrc, int nSrcStep, Npp16s * pDst, int nDstStep, NppiSize oSizeROI, NppRoundMode eRoundMode, int nScaleFactor)

Single channel 32-bit floating point to 16-bit signed conversion.

Parameters:

pSrc Source-Image Pointer.

nSrcStep Source-Image Line Step.

pDst Destination-Image Pointer.

nDstStep Destination-Image Line Step.

oSizeROI Region-of-Interest (ROI).

eRoundMode Flag specifying how fractional float values are rounded to integer values.

nScaleFactor Integer Result Scaling.

Returns:

[Image Data Related Error Codes](#), [ROI Related Error Codes](#)

7.8.1.34 NppStatus nppiConvert_32f16s_C3R (const Npp32f * pSrc, int nSrcStep, Npp16s * pDst, int nDstStep, NppiSize oSizeROI, NppRoundMode eRoundMode)

Three channel 32-bit floating point to 16-bit signed conversion.

Parameters:

pSrc Source-Image Pointer.

nSrcStep Source-Image Line Step.

pDst Destination-Image Pointer.

nDstStep Destination-Image Line Step.

oSizeROI Region-of-Interest (ROI).

eRoundMode Flag specifying how fractional float values are rounded to integer values.

Returns:

[Image Data Related Error Codes](#), [ROI Related Error Codes](#)

7.8.1.35 `NppStatus nppiConvert_32f16s_C4R (const Npp32f * pSrc, int nSrcStep, Npp16s * pDst, int nDstStep, NppiSize oSizeROI, NppRoundMode eRoundMode)`

Four channel 32-bit floating point to 16-bit signed conversion.

Parameters:

pSrc Source-Image Pointer.

nSrcStep Source-Image Line Step.

pDst Destination-Image Pointer.

nDstStep Destination-Image Line Step.

oSizeROI Region-of-Interest (ROI).

eRoundMode Flag specifying how fractional float values are rounded to integer values.

Returns:

[Image Data Related Error Codes](#), [ROI Related Error Codes](#)

7.8.1.36 `NppStatus nppiConvert_32f16u_AC4R (const Npp32f * pSrc, int nSrcStep, Npp16u * pDst, int nDstStep, NppiSize oSizeROI, NppRoundMode eRoundMode)`

Four channel 32-bit floating point to 16-bit unsigned conversion, not affecting Alpha.

Parameters:

pSrc Source-Image Pointer.

nSrcStep Source-Image Line Step.

pDst Destination-Image Pointer.

nDstStep Destination-Image Line Step.

oSizeROI Region-of-Interest (ROI).

eRoundMode Flag specifying how fractional float values are rounded to integer values.

Returns:

[Image Data Related Error Codes](#), [ROI Related Error Codes](#)

7.8.1.37 NppStatus nppiConvert_32f16u_C1R (const Npp32f * pSrc, int nSrcStep, Npp16u * pDst, int nDstStep, NppiSize oSizeROI, NppRoundMode eRoundMode)

Single channel 32-bit floating point to 16-bit unsigned conversion.

Parameters:

pSrc Source-Image Pointer.

nSrcStep Source-Image Line Step.

pDst Destination-Image Pointer.

nDstStep Destination-Image Line Step.

oSizeROI Region-of-Interest (ROI).

eRoundMode Flag specifying how fractional float values are rounded to integer values.

Returns:

[Image Data Related Error Codes](#), [ROI Related Error Codes](#)

7.8.1.38 NppStatus nppiConvert_32f16u_C1RSfs (const Npp32f * pSrc, int nSrcStep, Npp16u * pDst, int nDstStep, NppiSize oSizeROI, NppRoundMode eRoundMode, int nScaleFactor)

Single channel 32-bit floating point to 16-bit unsigned conversion.

Parameters:

pSrc Source-Image Pointer.

nSrcStep Source-Image Line Step.

pDst Destination-Image Pointer.

nDstStep Destination-Image Line Step.

oSizeROI Region-of-Interest (ROI).

eRoundMode Flag specifying how fractional float values are rounded to integer values.

nScaleFactor Integer Result Scaling.

Returns:

[Image Data Related Error Codes](#), [ROI Related Error Codes](#)

7.8.1.39 NppStatus nppiConvert_32f16u_C3R (const Npp32f * pSrc, int nSrcStep, Npp16u * pDst, int nDstStep, NppiSize oSizeROI, NppRoundMode eRoundMode)

Three channel 32-bit floating point to 16-bit unsigned conversion.

Parameters:

pSrc Source-Image Pointer.

nSrcStep Source-Image Line Step.

pDst Destination-Image Pointer.

nDstStep Destination-Image Line Step.

oSizeROI Region-of-Interest (ROI).

eRoundMode Flag specifying how fractional float values are rounded to integer values.

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.8.1.40 `NppStatus nppiConvert_32f16u_C4R (const Npp32f * pSrc, int nSrcStep, Npp16u * pDst, int nDstStep, NppiSize oSizeROI, NppRoundMode eRoundMode)`

Four channel 32-bit floating point to 16-bit unsigned conversion.

Parameters:

pSrc Source-Image Pointer.

nSrcStep Source-Image Line Step.

pDst Destination-Image Pointer.

nDstStep Destination-Image Line Step.

oSizeROI Region-of-Interest (ROI).

eRoundMode Flag specifying how fractional float values are rounded to integer values.

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.8.1.41 `NppStatus nppiConvert_32f32s_C1RSfs (const Npp32f * pSrc, int nSrcStep, Npp32s * pDst, int nDstStep, NppiSize oSizeROI, NppRoundMode eRoundMode, int nScaleFactor)`

Single channel 32-bit floating point to 32-bit signed conversion.

Parameters:

pSrc Source-Image Pointer.

nSrcStep Source-Image Line Step.

pDst Destination-Image Pointer.

nDstStep Destination-Image Line Step.

oSizeROI Region-of-Interest (ROI).

eRoundMode Flag specifying how fractional float values are rounded to integer values.

nScaleFactor Integer Result Scaling.

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.8.1.42 NppStatus nppiConvert_32f32u_C1RSfs (const Npp32f * pSrc, int nSrcStep, Npp32u * pDst, int nDstStep, NppiSize oSizeROI, NppRoundMode eRoundMode, int nScaleFactor)

Single channel 32-bit floating point to 32-bit unsigned conversion.

Parameters:

pSrc Source-Image Pointer.

nSrcStep Source-Image Line Step.

pDst Destination-Image Pointer.

nDstStep Destination-Image Line Step.

oSizeROI Region-of-Interest (ROI).

eRoundMode Flag specifying how fractional float values are rounded to integer values.

nScaleFactor Integer Result Scaling.

Returns:

[Image Data Related Error Codes](#), [ROI Related Error Codes](#)

7.8.1.43 NppStatus nppiConvert_32f8s_AC4R (const Npp32f * pSrc, int nSrcStep, Npp8s * pDst, int nDstStep, NppiSize oSizeROI, NppRoundMode eRoundMode)

Four channel 32-bit floating point to 8-bit signed conversion, not affecting Alpha.

Parameters:

pSrc Source-Image Pointer.

nSrcStep Source-Image Line Step.

pDst Destination-Image Pointer.

nDstStep Destination-Image Line Step.

oSizeROI Region-of-Interest (ROI).

eRoundMode Flag specifying how fractional float values are rounded to integer values.

Returns:

[Image Data Related Error Codes](#), [ROI Related Error Codes](#)

7.8.1.44 NppStatus nppiConvert_32f8s_C1R (const Npp32f * pSrc, int nSrcStep, Npp8s * pDst, int nDstStep, NppiSize oSizeROI, NppRoundMode eRoundMode)

Single channel 32-bit floating point to 8-bit signed conversion.

Parameters:

pSrc Source-Image Pointer.

nSrcStep Source-Image Line Step.

pDst Destination-Image Pointer.

nDstStep Destination-Image Line Step.

oSizeROI Region-of-Interest (ROI).

eRoundMode Flag specifying how fractional float values are rounded to integer values.

Returns:

[Image Data Related Error Codes](#), [ROI Related Error Codes](#)

7.8.1.45 `NppStatus nppiConvert_32f8s_C1RSfs (const Npp32f * pSrc, int nSrcStep, Npp8s * pDst, int nDstStep, NppiSize oSizeROI, NppRoundMode eRoundMode, int nScaleFactor)`

Single channel 32-bit floating point to 8-bit signed conversion.

Parameters:

pSrc Source-Image Pointer.

nSrcStep Source-Image Line Step.

pDst Destination-Image Pointer.

nDstStep Destination-Image Line Step.

oSizeROI Region-of-Interest (ROI).

eRoundMode Flag specifying how fractional float values are rounded to integer values.

nScaleFactor Integer Result Scaling.

Returns:

[Image Data Related Error Codes](#), [ROI Related Error Codes](#)

7.8.1.46 `NppStatus nppiConvert_32f8s_C3R (const Npp32f * pSrc, int nSrcStep, Npp8s * pDst, int nDstStep, NppiSize oSizeROI, NppRoundMode eRoundMode)`

Three channel 32-bit floating point to 8-bit signed conversion.

Parameters:

pSrc Source-Image Pointer.

nSrcStep Source-Image Line Step.

pDst Destination-Image Pointer.

nDstStep Destination-Image Line Step.

oSizeROI Region-of-Interest (ROI).

eRoundMode Flag specifying how fractional float values are rounded to integer values.

Returns:

[Image Data Related Error Codes](#), [ROI Related Error Codes](#)

7.8.1.47 NppStatus nppiConvert_32f8s_C4R (const Npp32f * pSrc, int nSrcStep, Npp8s * pDst, int nDstStep, NppiSize oSizeROI, NppRoundMode eRoundMode)

Four channel 32-bit floating point to 8-bit signed conversion.

Parameters:

pSrc Source-Image Pointer.

nSrcStep Source-Image Line Step.

pDst Destination-Image Pointer.

nDstStep Destination-Image Line Step.

oSizeROI Region-of-Interest (ROI).

eRoundMode Flag specifying how fractional float values are rounded to integer values.

Returns:

[Image Data Related Error Codes](#), [ROI Related Error Codes](#)

7.8.1.48 NppStatus nppiConvert_32f8u_AC4R (const Npp32f * pSrc, int nSrcStep, Npp8u * pDst, int nDstStep, NppiSize oSizeROI, NppRoundMode eRoundMode)

Four channel 32-bit floating point to 8-bit unsigned conversion, not affecting Alpha.

Parameters:

pSrc Source-Image Pointer.

nSrcStep Source-Image Line Step.

pDst Destination-Image Pointer.

nDstStep Destination-Image Line Step.

oSizeROI Region-of-Interest (ROI).

eRoundMode Flag specifying how fractional float values are rounded to integer values.

Returns:

[Image Data Related Error Codes](#), [ROI Related Error Codes](#)

7.8.1.49 NppStatus nppiConvert_32f8u_C1R (const Npp32f * pSrc, int nSrcStep, Npp8u * pDst, int nDstStep, NppiSize oSizeROI, NppRoundMode eRoundMode)

Single channel 32-bit floating point to 8-bit unsigned conversion.

Parameters:

pSrc Source-Image Pointer.

nSrcStep Source-Image Line Step.

pDst Destination-Image Pointer.

nDstStep Destination-Image Line Step.

oSizeROI Region-of-Interest (ROI).

eRoundMode Flag specifying how fractional float values are rounded to integer values.

Returns:

[Image Data Related Error Codes](#), [ROI Related Error Codes](#)

7.8.1.50 NppStatus nppiConvert_32f8u_C1RSfs (const Npp32f * pSrc, int nSrcStep, Npp8u * pDst, int nDstStep, NppiSize oSizeROI, NppRoundMode eRoundMode, int nScaleFactor)

Single channel 32-bit floating point to 8-bit unsigned conversion.

Parameters:

pSrc Source-Image Pointer.

nSrcStep Source-Image Line Step.

pDst Destination-Image Pointer.

nDstStep Destination-Image Line Step.

oSizeROI Region-of-Interest (ROI).

eRoundMode Flag specifying how fractional float values are rounded to integer values.

nScaleFactor Integer Result Scaling.

Returns:

[Image Data Related Error Codes](#), [ROI Related Error Codes](#)

7.8.1.51 NppStatus nppiConvert_32f8u_C3R (const Npp32f * pSrc, int nSrcStep, Npp8u * pDst, int nDstStep, NppiSize oSizeROI, NppRoundMode eRoundMode)

Three channel 32-bit floating point to 8-bit unsigned conversion.

Parameters:

pSrc Source-Image Pointer.

nSrcStep Source-Image Line Step.

pDst Destination-Image Pointer.

nDstStep Destination-Image Line Step.

oSizeROI Region-of-Interest (ROI).

eRoundMode Flag specifying how fractional float values are rounded to integer values.

Returns:

[Image Data Related Error Codes](#), [ROI Related Error Codes](#)

7.8.1.52 NppStatus nppiConvert_32f8u_C4R (const Npp32f * pSrc, int nSrcStep, Npp8u * pDst, int nDstStep, NppiSize oSizeROI, NppRoundMode eRoundMode)

Four channel 32-bit floating point to 8-bit unsigned conversion.

Parameters:

pSrc Source-Image Pointer.

nSrcStep Source-Image Line Step.

pDst Destination-Image Pointer.

nDstStep Destination-Image Line Step.

oSizeROI Region-of-Interest (ROI).

eRoundMode Flag specifying how fractional float values are rounded to integer values.

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.8.1.53 NppStatus nppiConvert_32s16s_C1RSfs (const Npp32s * pSrc, int nSrcStep, Npp16s * pDst, int nDstStep, NppiSize oSizeROI, NppRoundMode eRoundMode, int nScaleFactor)

Single channel 32-bit unsigned to 16-bit signed conversion.

Parameters:

pSrc Source-Image Pointer.

nSrcStep Source-Image Line Step.

pDst Destination-Image Pointer.

nDstStep Destination-Image Line Step.

oSizeROI Region-of-Interest (ROI).

eRoundMode Rounding Mode Parameter.

nScaleFactor Integer Result Scaling.

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.8.1.54 NppStatus nppiConvert_32s16u_C1RSfs (const Npp32s * pSrc, int nSrcStep, Npp16u * pDst, int nDstStep, NppiSize oSizeROI, NppRoundMode eRoundMode, int nScaleFactor)

Single channel 32-bit unsigned to 16-bit unsigned conversion.

Parameters:

pSrc Source-Image Pointer.

nSrcStep Source-Image Line Step.

pDst Destination-Image Pointer.

nDstStep Destination-Image Line Step.

oSizeROI Region-of-Interest (ROI).

eRoundMode Rounding Mode Parameter.

nScaleFactor Integer Result Scaling.

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.8.1.55 NppStatus nppiConvert_32s32f_C1R (const Npp32s * pSrc, int nSrcStep, Npp32f * pDst, int nDstStep, NppiSize oSizeROI)

Single channel 32-bit signed to 32-bit floating-point conversion.

Parameters:

pSrc Source-Image Pointer.
nSrcStep Source-Image Line Step.
pDst Destination-Image Pointer.
nDstStep Destination-Image Line Step.
oSizeROI Region-of-Interest (ROI).

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.8.1.56 NppStatus nppiConvert_32s32u_C1Rs (const Npp32s * pSrc, int nSrcStep, Npp32u * pDst, int nDstStep, NppiSize oSizeROI)

Single channel 32-bit signed to 32-bit unsigned conversion with saturation.

Parameters:

pSrc Source-Image Pointer.
nSrcStep Source-Image Line Step.
pDst Destination-Image Pointer.
nDstStep Destination-Image Line Step.
oSizeROI Region-of-Interest (ROI).

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.8.1.57 NppStatus nppiConvert_32s8s_AC4R (const Npp32s * pSrc, int nSrcStep, Npp8s * pDst, int nDstStep, NppiSize oSizeROI)

Four channel 32-bit signed to 8-bit signed conversion, not affecting Alpha.

Parameters:

pSrc Source-Image Pointer.
nSrcStep Source-Image Line Step.
pDst Destination-Image Pointer.
nDstStep Destination-Image Line Step.
oSizeROI Region-of-Interest (ROI).

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.8.1.58 NppStatus nppiConvert_32s8s_C1R (const Npp32s * pSrc, int nSrcStep, Npp8s * pDst, int nDstStep, NppiSize oSizeROI)

Single channel 32-bit signed to 8-bit signed conversion.

Parameters:

pSrc Source-Image Pointer.
nSrcStep Source-Image Line Step.
pDst Destination-Image Pointer.
nDstStep Destination-Image Line Step.
oSizeROI Region-of-Interest (ROI).

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.8.1.59 NppStatus nppiConvert_32s8s_C3R (const Npp32s * pSrc, int nSrcStep, Npp8s * pDst, int nDstStep, NppiSize oSizeROI)

Three channel 32-bit signed to 8-bit signed conversion.

Parameters:

pSrc Source-Image Pointer.
nSrcStep Source-Image Line Step.
pDst Destination-Image Pointer.
nDstStep Destination-Image Line Step.
oSizeROI Region-of-Interest (ROI).

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.8.1.60 NppStatus nppiConvert_32s8s_C4R (const Npp32s * pSrc, int nSrcStep, Npp8s * pDst, int nDstStep, NppiSize oSizeROI)

Four channel 32-bit signed to 8-bit signed conversion.

Parameters:

pSrc Source-Image Pointer.
nSrcStep Source-Image Line Step.
pDst Destination-Image Pointer.
nDstStep Destination-Image Line Step.
oSizeROI Region-of-Interest (ROI).

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.8.1.61 NppStatus nppiConvert_32s8u_AC4R (const Npp32s * pSrc, int nSrcStep, Npp8u * pDst, int nDstStep, NppiSize oSizeROI)

Four channel 32-bit signed to 8-bit unsigned conversion, not affecting Alpha.

Parameters:

pSrc Source-Image Pointer.
nSrcStep Source-Image Line Step.
pDst Destination-Image Pointer.
nDstStep Destination-Image Line Step.
oSizeROI Region-of-Interest (ROI).

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.8.1.62 NppStatus nppiConvert_32s8u_C1R (const Npp32s * pSrc, int nSrcStep, Npp8u * pDst, int nDstStep, NppiSize oSizeROI)

Single channel 32-bit signed to 8-bit unsigned conversion.

Parameters:

pSrc Source-Image Pointer.
nSrcStep Source-Image Line Step.
pDst Destination-Image Pointer.
nDstStep Destination-Image Line Step.
oSizeROI Region-of-Interest (ROI).

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.8.1.63 NppStatus nppiConvert_32s8u_C3R (const Npp32s * pSrc, int nSrcStep, Npp8u * pDst, int nDstStep, NppiSize oSizeROI)

Three channel 32-bit signed to 8-bit unsigned conversion.

Parameters:

pSrc Source-Image Pointer.
nSrcStep Source-Image Line Step.
pDst Destination-Image Pointer.
nDstStep Destination-Image Line Step.
oSizeROI Region-of-Interest (ROI).

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.8.1.64 NppStatus nppiConvert_32s8u_C4R (const Npp32s * pSrc, int nSrcStep, Npp8u * pDst, int nDstStep, NppiSize oSizeROI)

Four channel 32-bit signed to 8-bit unsigned conversion.

Parameters:

pSrc Source-Image Pointer.
nSrcStep Source-Image Line Step.
pDst Destination-Image Pointer.
nDstStep Destination-Image Line Step.
oSizeROI Region-of-Interest (ROI).

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.8.1.65 NppStatus nppiConvert_32u16s_C1RSfs (const Npp32u * pSrc, int nSrcStep, Npp16s * pDst, int nDstStep, NppiSize oSizeROI, NppRoundMode eRoundMode, int nScaleFactor)

Single channel 32-bit unsigned to 16-bit signed conversion.

Parameters:

pSrc Source-Image Pointer.
nSrcStep Source-Image Line Step.
pDst Destination-Image Pointer.
nDstStep Destination-Image Line Step.
oSizeROI Region-of-Interest (ROI).
eRoundMode Rounding Mode Parameter.
nScaleFactor Integer Result Scaling.

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.8.1.66 NppStatus nppiConvert_32u16u_C1RSfs (const Npp32u * pSrc, int nSrcStep, Npp16u * pDst, int nDstStep, NppiSize oSizeROI, NppRoundMode eRoundMode, int nScaleFactor)

Single channel 32-bit unsigned to 16-bit unsigned conversion.

Parameters:

pSrc Source-Image Pointer.
nSrcStep Source-Image Line Step.
pDst Destination-Image Pointer.
nDstStep Destination-Image Line Step.
oSizeROI Region-of-Interest (ROI).

eRoundMode Rounding Mode Parameter.

nScaleFactor Integer Result Scaling.

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.8.1.67 `NppStatus nppiConvert_32u32f_C1R (const Npp32u * pSrc, int nSrcStep, Npp32f * pDst, int nDstStep, NppiSize oSizeROI)`

Single channel 32-bit unsigned to 32-bit floating-point conversion.

Parameters:

pSrc Source-Image Pointer.

nSrcStep Source-Image Line Step.

pDst Destination-Image Pointer.

nDstStep Destination-Image Line Step.

oSizeROI Region-of-Interest (ROI).

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.8.1.68 `NppStatus nppiConvert_32u32s_C1RSfs (const Npp32u * pSrc, int nSrcStep, Npp32s * pDst, int nDstStep, NppiSize oSizeROI, NppRoundMode eRoundMode, int nScaleFactor)`

Single channel 32-bit unsigned to 32-bit signed conversion.

Parameters:

pSrc Source-Image Pointer.

nSrcStep Source-Image Line Step.

pDst Destination-Image Pointer.

nDstStep Destination-Image Line Step.

oSizeROI Region-of-Interest (ROI).

eRoundMode Rounding Mode Parameter.

nScaleFactor Integer Result Scaling.

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.8.1.69 `NppStatus nppiConvert_32u8s_C1RSfs (const Npp32u * pSrc, int nSrcStep, Npp8s * pDst, int nDstStep, NppiSize oSizeROI, NppRoundMode eRoundMode, int nScaleFactor)`

Single channel 32-bit unsigned to 8-bit signed conversion.

Parameters:

pSrc Source-Image Pointer.
nSrcStep Source-Image Line Step.
pDst Destination-Image Pointer.
nDstStep Destination-Image Line Step.
oSizeROI Region-of-Interest (ROI).
eRoundMode Rounding Mode Parameter.
nScaleFactor Integer Result Scaling.

Returns:

[Image Data Related Error Codes](#), [ROI Related Error Codes](#)

7.8.1.70 `NppStatus nppiConvert_32u8u_C1RSfs (const Npp32u * pSrc, int nSrcStep, Npp8u * pDst, int nDstStep, NppiSize oSizeROI, NppRoundMode eRoundMode, int nScaleFactor)`

Single channel 32-bit unsigned to 8-bit unsigned conversion.

Parameters:

pSrc Source-Image Pointer.
nSrcStep Source-Image Line Step.
pDst Destination-Image Pointer.
nDstStep Destination-Image Line Step.
oSizeROI Region-of-Interest (ROI).
eRoundMode Rounding Mode Parameter.
nScaleFactor Integer Result Scaling.

Returns:

[Image Data Related Error Codes](#), [ROI Related Error Codes](#)

7.8.1.71 `NppStatus nppiConvert_8s16s_C1R (const Npp8s * pSrc, int nSrcStep, Npp16s * pDst, int nDstStep, NppiSize oSizeROI)`

Single channel 8-bit signed to 16-bit signed conversion.

Parameters:

pSrc Source-Image Pointer.
nSrcStep Source-Image Line Step.
pDst Destination-Image Pointer.

nDstStep Destination-Image Line Step.

oSizeROI Region-of-Interest (ROI).

Returns:

[Image Data Related Error Codes](#), [ROI Related Error Codes](#)

7.8.1.72 `NppStatus nppiConvert_8s16u_C1Rs (const Npp8s * pSrc, int nSrcStep, Npp16u * pDst, int nDstStep, NppiSize oSizeROI)`

Single channel 8-bit signed to 16-bit unsigned conversion with saturation.

Parameters:

pSrc Source-Image Pointer.

nSrcStep Source-Image Line Step.

pDst Destination-Image Pointer.

nDstStep Destination-Image Line Step.

oSizeROI Region-of-Interest (ROI).

Returns:

[Image Data Related Error Codes](#), [ROI Related Error Codes](#)

7.8.1.73 `NppStatus nppiConvert_8s32f_AC4R (const Npp8s * pSrc, int nSrcStep, Npp32f * pDst, int nDstStep, NppiSize oSizeROI)`

Four channel 8-bit signed to 32-bit floating-point conversion, not affecting Alpha.

Parameters:

pSrc Source-Image Pointer.

nSrcStep Source-Image Line Step.

pDst Destination-Image Pointer.

nDstStep Destination-Image Line Step.

oSizeROI Region-of-Interest (ROI).

Returns:

[Image Data Related Error Codes](#), [ROI Related Error Codes](#)

7.8.1.74 `NppStatus nppiConvert_8s32f_C1R (const Npp8s * pSrc, int nSrcStep, Npp32f * pDst, int nDstStep, NppiSize oSizeROI)`

Single channel 8-bit signed to 32-bit floating-point conversion.

Parameters:

pSrc Source-Image Pointer.

nSrcStep Source-Image Line Step.
pDst Destination-Image Pointer.
nDstStep Destination-Image Line Step.
oSizeROI Region-of-Interest (ROI).

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.8.1.75 `NppStatus nppiConvert_8s32f_C3R (const Npp8s * pSrc, int nSrcStep, Npp32f * pDst, int nDstStep, NppiSize oSizeROI)`

Three channel 8-bit signed to 32-bit floating-point conversion.

Parameters:

pSrc Source-Image Pointer.
nSrcStep Source-Image Line Step.
pDst Destination-Image Pointer.
nDstStep Destination-Image Line Step.
oSizeROI Region-of-Interest (ROI).

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.8.1.76 `NppStatus nppiConvert_8s32f_C4R (const Npp8s * pSrc, int nSrcStep, Npp32f * pDst, int nDstStep, NppiSize oSizeROI)`

Four channel 8-bit signed to 32-bit floating-point conversion.

Parameters:

pSrc Source-Image Pointer.
nSrcStep Source-Image Line Step.
pDst Destination-Image Pointer.
nDstStep Destination-Image Line Step.
oSizeROI Region-of-Interest (ROI).

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.8.1.77 NppStatus nppiConvert_8s32s_AC4R (const Npp8s * pSrc, int nSrcStep, Npp32s * pDst, int nDstStep, NppiSize oSizeROI)

Four channel 8-bit signed to 32-bit signed conversion, not affecting Alpha.

Parameters:

pSrc Source-Image Pointer.
nSrcStep Source-Image Line Step.
pDst Destination-Image Pointer.
nDstStep Destination-Image Line Step.
oSizeROI Region-of-Interest (ROI).

Returns:

[Image Data Related Error Codes](#), [ROI Related Error Codes](#)

7.8.1.78 NppStatus nppiConvert_8s32s_C1R (const Npp8s * pSrc, int nSrcStep, Npp32s * pDst, int nDstStep, NppiSize oSizeROI)

Single channel 8-bit signed to 32-bit signed conversion.

Parameters:

pSrc Source-Image Pointer.
nSrcStep Source-Image Line Step.
pDst Destination-Image Pointer.
nDstStep Destination-Image Line Step.
oSizeROI Region-of-Interest (ROI).

Returns:

[Image Data Related Error Codes](#), [ROI Related Error Codes](#)

7.8.1.79 NppStatus nppiConvert_8s32s_C3R (const Npp8s * pSrc, int nSrcStep, Npp32s * pDst, int nDstStep, NppiSize oSizeROI)

Three channel 8-bit signed to 32-bit signed conversion.

Parameters:

pSrc Source-Image Pointer.
nSrcStep Source-Image Line Step.
pDst Destination-Image Pointer.
nDstStep Destination-Image Line Step.
oSizeROI Region-of-Interest (ROI).

Returns:

[Image Data Related Error Codes](#), [ROI Related Error Codes](#)

7.8.1.80 NppStatus nppiConvert_8s32s_C4R (const Npp8s * pSrc, int nSrcStep, Npp32s * pDst, int nDstStep, NppiSize oSizeROI)

Four channel 8-bit signed to 32-bit signed conversion.

Parameters:

pSrc Source-Image Pointer.
nSrcStep Source-Image Line Step.
pDst Destination-Image Pointer.
nDstStep Destination-Image Line Step.
oSizeROI Region-of-Interest (ROI).

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.8.1.81 NppStatus nppiConvert_8s32u_C1Rs (const Npp8s * pSrc, int nSrcStep, Npp32u * pDst, int nDstStep, NppiSize oSizeROI)

Single channel 8-bit signed to 32-bit unsigned conversion with saturation.

Parameters:

pSrc Source-Image Pointer.
nSrcStep Source-Image Line Step.
pDst Destination-Image Pointer.
nDstStep Destination-Image Line Step.
oSizeROI Region-of-Interest (ROI).

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.8.1.82 NppStatus nppiConvert_8s8u_C1Rs (const Npp8s * pSrc, int nSrcStep, Npp8u * pDst, int nDstStep, NppiSize oSizeROI)

Single channel 8-bit signed to 8-bit unsigned conversion with saturation.

Parameters:

pSrc Source-Image Pointer.
nSrcStep Source-Image Line Step.
pDst Destination-Image Pointer.
nDstStep Destination-Image Line Step.
oSizeROI Region-of-Interest (ROI).

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.8.1.83 NppStatus nppiConvert_8u16s_AC4R (const Npp8u * pSrc, int nSrcStep, Npp16s * pDst, int nDstStep, NppiSize oSizeROI)

Four channel 8-bit unsigned to 16-bit signed conversion, not affecting Alpha.

Parameters:

pSrc Source-Image Pointer.
nSrcStep Source-Image Line Step.
pDst Destination-Image Pointer.
nDstStep Destination-Image Line Step.
oSizeROI Region-of-Interest (ROI).

Returns:

[Image Data Related Error Codes](#), [ROI Related Error Codes](#)

7.8.1.84 NppStatus nppiConvert_8u16s_C1R (const Npp8u * pSrc, int nSrcStep, Npp16s * pDst, int nDstStep, NppiSize oSizeROI)

Single channel 8-bit unsigned to 16-bit signed conversion.

Parameters:

pSrc Source-Image Pointer.
nSrcStep Source-Image Line Step.
pDst Destination-Image Pointer.
nDstStep Destination-Image Line Step.
oSizeROI Region-of-Interest (ROI).

Returns:

[Image Data Related Error Codes](#), [ROI Related Error Codes](#)

7.8.1.85 NppStatus nppiConvert_8u16s_C3R (const Npp8u * pSrc, int nSrcStep, Npp16s * pDst, int nDstStep, NppiSize oSizeROI)

Three channel 8-bit unsigned to 16-bit signed conversion.

Parameters:

pSrc Source-Image Pointer.
nSrcStep Source-Image Line Step.
pDst Destination-Image Pointer.
nDstStep Destination-Image Line Step.
oSizeROI Region-of-Interest (ROI).

Returns:

[Image Data Related Error Codes](#), [ROI Related Error Codes](#)

7.8.1.86 NppStatus nppiConvert_8u16s_C4R (const Npp8u * pSrc, int nSrcStep, Npp16s * pDst, int nDstStep, NppiSize oSizeROI)

Four channel 8-bit unsigned to 16-bit signed conversion.

Parameters:

pSrc Source-Image Pointer.
nSrcStep Source-Image Line Step.
pDst Destination-Image Pointer.
nDstStep Destination-Image Line Step.
oSizeROI Region-of-Interest (ROI).

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.8.1.87 NppStatus nppiConvert_8u16u_AC4R (const Npp8u * pSrc, int nSrcStep, Npp16u * pDst, int nDstStep, NppiSize oSizeROI)

Four channel 8-bit unsigned to 16-bit unsigned conversion, not affecting Alpha.

Parameters:

pSrc Source-Image Pointer.
nSrcStep Source-Image Line Step.
pDst Destination-Image Pointer.
nDstStep Destination-Image Line Step.
oSizeROI Region-of-Interest (ROI).

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.8.1.88 NppStatus nppiConvert_8u16u_C1R (const Npp8u * pSrc, int nSrcStep, Npp16u * pDst, int nDstStep, NppiSize oSizeROI)

Single channel 8-bit unsigned to 16-bit unsigned conversion.

Parameters:

pSrc Source-Image Pointer.
nSrcStep Source-Image Line Step.
pDst Destination-Image Pointer.
nDstStep Destination-Image Line Step.
oSizeROI Region-of-Interest (ROI).

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.8.1.89 NppStatus nppiConvert_8u16u_C3R (const Npp8u * pSrc, int nSrcStep, Npp16u * pDst, int nDstStep, NppiSize oSizeROI)

Three channel 8-bit unsigned to 16-bit unsigned conversion.

Parameters:

pSrc Source-Image Pointer.
nSrcStep Source-Image Line Step.
pDst Destination-Image Pointer.
nDstStep Destination-Image Line Step.
oSizeROI Region-of-Interest (ROI).

Returns:

[Image Data Related Error Codes](#), [ROI Related Error Codes](#)

7.8.1.90 NppStatus nppiConvert_8u16u_C4R (const Npp8u * pSrc, int nSrcStep, Npp16u * pDst, int nDstStep, NppiSize oSizeROI)

Four channel 8-bit unsigned to 16-bit unsigned conversion.

Parameters:

pSrc Source-Image Pointer.
nSrcStep Source-Image Line Step.
pDst Destination-Image Pointer.
nDstStep Destination-Image Line Step.
oSizeROI Region-of-Interest (ROI).

Returns:

[Image Data Related Error Codes](#), [ROI Related Error Codes](#)

7.8.1.91 NppStatus nppiConvert_8u32f_AC4R (const Npp8u * pSrc, int nSrcStep, Npp32f * pDst, int nDstStep, NppiSize oSizeROI)

Four channel 8-bit unsigned to 32-bit floating-point conversion, not affecting Alpha.

Parameters:

pSrc Source-Image Pointer.
nSrcStep Source-Image Line Step.
pDst Destination-Image Pointer.
nDstStep Destination-Image Line Step.
oSizeROI Region-of-Interest (ROI).

Returns:

[Image Data Related Error Codes](#), [ROI Related Error Codes](#)

7.8.1.92 NppStatus nppiConvert_8u32f_C1R (const Npp8u * pSrc, int nSrcStep, Npp32f * pDst, int nDstStep, NppiSize oSizeROI)

Single channel 8-bit unsigned to 32-bit floating-point conversion.

Parameters:

pSrc Source-Image Pointer.
nSrcStep Source-Image Line Step.
pDst Destination-Image Pointer.
nDstStep Destination-Image Line Step.
oSizeROI Region-of-Interest (ROI).

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.8.1.93 NppStatus nppiConvert_8u32f_C3R (const Npp8u * pSrc, int nSrcStep, Npp32f * pDst, int nDstStep, NppiSize oSizeROI)

Three channel 8-bit unsigned to 32-bit floating-point conversion.

Parameters:

pSrc Source-Image Pointer.
nSrcStep Source-Image Line Step.
pDst Destination-Image Pointer.
nDstStep Destination-Image Line Step.
oSizeROI Region-of-Interest (ROI).

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.8.1.94 NppStatus nppiConvert_8u32f_C4R (const Npp8u * pSrc, int nSrcStep, Npp32f * pDst, int nDstStep, NppiSize oSizeROI)

Four channel 8-bit unsigned to 32-bit floating-point conversion.

Parameters:

pSrc Source-Image Pointer.
nSrcStep Source-Image Line Step.
pDst Destination-Image Pointer.
nDstStep Destination-Image Line Step.
oSizeROI Region-of-Interest (ROI).

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.8.1.95 NppStatus nppiConvert_8u32s_AC4R (const Npp8u * pSrc, int nSrcStep, Npp32s * pDst, int nDstStep, NppiSize oSizeROI)

Four channel 8-bit unsigned to 32-bit signed conversion, not affecting Alpha.

Parameters:

pSrc Source-Image Pointer.
nSrcStep Source-Image Line Step.
pDst Destination-Image Pointer.
nDstStep Destination-Image Line Step.
oSizeROI Region-of-Interest (ROI).

Returns:

[Image Data Related Error Codes](#), [ROI Related Error Codes](#)

7.8.1.96 NppStatus nppiConvert_8u32s_C1R (const Npp8u * pSrc, int nSrcStep, Npp32s * pDst, int nDstStep, NppiSize oSizeROI)

Single channel 8-bit unsigned to 32-bit signed conversion.

Parameters:

pSrc Source-Image Pointer.
nSrcStep Source-Image Line Step.
pDst Destination-Image Pointer.
nDstStep Destination-Image Line Step.
oSizeROI Region-of-Interest (ROI).

Returns:

[Image Data Related Error Codes](#), [ROI Related Error Codes](#)

7.8.1.97 NppStatus nppiConvert_8u32s_C3R (const Npp8u * pSrc, int nSrcStep, Npp32s * pDst, int nDstStep, NppiSize oSizeROI)

Three channel 8-bit unsigned to 32-bit signed conversion.

Parameters:

pSrc Source-Image Pointer.
nSrcStep Source-Image Line Step.
pDst Destination-Image Pointer.
nDstStep Destination-Image Line Step.
oSizeROI Region-of-Interest (ROI).

Returns:

[Image Data Related Error Codes](#), [ROI Related Error Codes](#)

7.8.1.98 NppStatus nppiConvert_8u32s_C4R (const Npp8u * pSrc, int nSrcStep, Npp32s * pDst, int nDstStep, NppiSize oSizeROI)

Four channel 8-bit unsigned to 32-bit signed conversion.

Parameters:

- pSrc* Source-Image Pointer.
- nSrcStep* Source-Image Line Step.
- pDst* Destination-Image Pointer.
- nDstStep* Destination-Image Line Step.
- oSizeROI* Region-of-Interest (ROI).

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.8.1.99 NppStatus nppiConvert_8u8s_C1RSfs (const Npp8u * pSrc, int nSrcStep, Npp8s * pDst, int nDstStep, NppiSize oSizeROI, NppRoundMode eRoundMode, int nScaleFactor)

Single channel 8-bit unsigned to 8-bit signed conversion.

Parameters:

- pSrc* Source-Image Pointer.
- nSrcStep* Source-Image Line Step.
- pDst* Destination-Image Pointer.
- nDstStep* Destination-Image Line Step.
- oSizeROI* Region-of-Interest (ROI).
- eRoundMode* Rounding Mode Parameter.
- nScaleFactor* Integer Result Scaling.

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.9 Scale

Scaled Bit-Depth Conversion

Scale bit-depth up and down.

To map source pixel `srcPixelValue` to destination pixel `dstPixelValue` the following equation is used:

$$\text{dstPixelValue} = \text{dstMinRangeValue} + \text{scaleFactor} * (\text{srcPixelValue} - \text{srcMinRangeValue})$$

where $\text{scaleFactor} = (\text{dstMaxRangeValue} - \text{dstMinRangeValue}) / (\text{srcMaxRangeValue} - \text{srcMinRangeValue})$.

For conversions between integer data types, the entire integer numeric range of the input data type is mapped onto the entire integer numeric range of the output data type.

For conversions to floating point data types the floating point data range is defined by the user supplied floating point values of `nMax` and `nMin` which are used as the `dstMaxRangeValue` and `dstMinRangeValue` respectively in the `scaleFactor` and `dstPixelValue` calculations and also as the saturation values to which output data is clamped.

When converting from floating-point values to integer values, `nMax` and `nMin` are used as the `srcMaxRangeValue` and `srcMinRangeValue` respectively in the `scaleFactor` and `dstPixelValue` calculations. Output values are saturated and clamped to the full output integer pixel value range.

- `NppStatus nppiScale_8u16u_C1R` (const `Npp8u *pSrc`, int `nSrcStep`, `Npp16u *pDst`, int `nDstStep`, `NppiSize oSizeROI`)
Single channel 8-bit unsigned to 16-bit unsigned conversion.
- `NppStatus nppiScale_8u16u_C3R` (const `Npp8u *pSrc`, int `nSrcStep`, `Npp16u *pDst`, int `nDstStep`, `NppiSize oSizeROI`)
Three channel 8-bit unsigned to 16-bit unsigned conversion.
- `NppStatus nppiScale_8u16u_C4R` (const `Npp8u *pSrc`, int `nSrcStep`, `Npp16u *pDst`, int `nDstStep`, `NppiSize oSizeROI`)
Four channel 8-bit unsigned to 16-bit unsigned conversion.
- `NppStatus nppiScale_8u16u_AC4R` (const `Npp8u *pSrc`, int `nSrcStep`, `Npp16u *pDst`, int `nDstStep`, `NppiSize oSizeROI`)
Four channel 8-bit unsigned to 16-bit unsigned conversion, not affecting Alpha.
- `NppStatus nppiScale_8u16s_C1R` (const `Npp8u *pSrc`, int `nSrcStep`, `Npp16s *pDst`, int `nDstStep`, `NppiSize oSizeROI`)
Single channel 8-bit unsigned to 16-bit signed conversion.
- `NppStatus nppiScale_8u16s_C3R` (const `Npp8u *pSrc`, int `nSrcStep`, `Npp16s *pDst`, int `nDstStep`, `NppiSize oSizeROI`)
Three channel 8-bit unsigned to 16-bit signed conversion.
- `NppStatus nppiScale_8u16s_C4R` (const `Npp8u *pSrc`, int `nSrcStep`, `Npp16s *pDst`, int `nDstStep`, `NppiSize oSizeROI`)
Four channel 8-bit unsigned to 16-bit signed conversion.
- `NppStatus nppiScale_8u16s_AC4R` (const `Npp8u *pSrc`, int `nSrcStep`, `Npp16s *pDst`, int `nDstStep`, `NppiSize oSizeROI`)

Four channel 8-bit unsigned to 16-bit signed conversion, not affecting Alpha.

- `NppStatus nppiScale_8u32s_C1R` (const `Npp8u` *pSrc, int nSrcStep, `Npp32s` *pDst, int nDstStep, `NppiSize` oSizeROI)

Single channel 8-bit unsigned to 32-bit signed conversion.

- `NppStatus nppiScale_8u32s_C3R` (const `Npp8u` *pSrc, int nSrcStep, `Npp32s` *pDst, int nDstStep, `NppiSize` oSizeROI)

Three channel 8-bit unsigned to 32-bit signed conversion.

- `NppStatus nppiScale_8u32s_C4R` (const `Npp8u` *pSrc, int nSrcStep, `Npp32s` *pDst, int nDstStep, `NppiSize` oSizeROI)

Four channel 8-bit unsigned to 32-bit signed conversion.

- `NppStatus nppiScale_8u32s_AC4R` (const `Npp8u` *pSrc, int nSrcStep, `Npp32s` *pDst, int nDstStep, `NppiSize` oSizeROI)

Four channel 8-bit unsigned to 32-bit signed conversion, not affecting Alpha.

- `NppStatus nppiScale_8u32f_C1R` (const `Npp8u` *pSrc, int nSrcStep, `Npp32f` *pDst, int nDstStep, `NppiSize` oSizeROI, `Npp32f` nMin, `Npp32f` nMax)

Single channel 8-bit unsigned to 32-bit floating-point conversion.

- `NppStatus nppiScale_8u32f_C3R` (const `Npp8u` *pSrc, int nSrcStep, `Npp32f` *pDst, int nDstStep, `NppiSize` oSizeROI, `Npp32f` nMin, `Npp32f` nMax)

Three channel 8-bit unsigned to 32-bit floating-point conversion.

- `NppStatus nppiScale_8u32f_C4R` (const `Npp8u` *pSrc, int nSrcStep, `Npp32f` *pDst, int nDstStep, `NppiSize` oSizeROI, `Npp32f` nMin, `Npp32f` nMax)

Four channel 8-bit unsigned to 32-bit floating-point conversion.

- `NppStatus nppiScale_8u32f_AC4R` (const `Npp8u` *pSrc, int nSrcStep, `Npp32f` *pDst, int nDstStep, `NppiSize` oSizeROI, `Npp32f` nMin, `Npp32f` nMax)

Four channel 8-bit unsigned to 32-bit floating-point conversion, not affecting Alpha.

- `NppStatus nppiScale_16u8u_C1R` (const `Npp16u` *pSrc, int nSrcStep, `Npp8u` *pDst, int nDstStep, `NppiSize` oSizeROI, `NppHintAlgorithm` hint)

Single channel 16-bit unsigned to 8-bit unsigned conversion.

- `NppStatus nppiScale_16u8u_C3R` (const `Npp16u` *pSrc, int nSrcStep, `Npp8u` *pDst, int nDstStep, `NppiSize` oSizeROI, `NppHintAlgorithm` hint)

Three channel 16-bit unsigned to 8-bit unsigned conversion.

- `NppStatus nppiScale_16u8u_C4R` (const `Npp16u` *pSrc, int nSrcStep, `Npp8u` *pDst, int nDstStep, `NppiSize` oSizeROI, `NppHintAlgorithm` hint)

Four channel 16-bit unsigned to 8-bit unsigned conversion.

- `NppStatus nppiScale_16u8u_AC4R` (const `Npp16u` *pSrc, int nSrcStep, `Npp8u` *pDst, int nDstStep, `NppiSize` oSizeROI, `NppHintAlgorithm` hint)

Four channel 16-bit unsigned to 8-bit unsigned conversion, not affecting Alpha.

- `NppStatus nppiScale_16s8u_C1R` (const `Npp16s` *pSrc, int nSrcStep, `Npp8u` *pDst, int nDstStep, `NppiSize` oSizeROI, `NppHintAlgorithm` hint)
Single channel 16-bit signed to 8-bit unsigned conversion.
- `NppStatus nppiScale_16s8u_C3R` (const `Npp16s` *pSrc, int nSrcStep, `Npp8u` *pDst, int nDstStep, `NppiSize` oSizeROI, `NppHintAlgorithm` hint)
Three channel 16-bit signed to 8-bit unsigned conversion.
- `NppStatus nppiScale_16s8u_C4R` (const `Npp16s` *pSrc, int nSrcStep, `Npp8u` *pDst, int nDstStep, `NppiSize` oSizeROI, `NppHintAlgorithm` hint)
Four channel 16-bit signed to 8-bit unsigned conversion.
- `NppStatus nppiScale_16s8u_AC4R` (const `Npp16s` *pSrc, int nSrcStep, `Npp8u` *pDst, int nDstStep, `NppiSize` oSizeROI, `NppHintAlgorithm` hint)
Four channel 16-bit signed to 8-bit unsigned conversion, not affecting Alpha.
- `NppStatus nppiScale_32s8u_C1R` (const `Npp32s` *pSrc, int nSrcStep, `Npp8u` *pDst, int nDstStep, `NppiSize` oSizeROI, `NppHintAlgorithm` hint)
Single channel 32-bit signed to 8-bit unsigned conversion.
- `NppStatus nppiScale_32s8u_C3R` (const `Npp32s` *pSrc, int nSrcStep, `Npp8u` *pDst, int nDstStep, `NppiSize` oSizeROI, `NppHintAlgorithm` hint)
Three channel 32-bit signed to 8-bit unsigned conversion.
- `NppStatus nppiScale_32s8u_C4R` (const `Npp32s` *pSrc, int nSrcStep, `Npp8u` *pDst, int nDstStep, `NppiSize` oSizeROI, `NppHintAlgorithm` hint)
Four channel 32-bit signed to 8-bit unsigned conversion.
- `NppStatus nppiScale_32s8u_AC4R` (const `Npp32s` *pSrc, int nSrcStep, `Npp8u` *pDst, int nDstStep, `NppiSize` oSizeROI, `NppHintAlgorithm` hint)
Four channel 32-bit signed to 8-bit unsigned conversion, not affecting Alpha.
- `NppStatus nppiScale_32f8u_C1R` (const `Npp32f` *pSrc, int nSrcStep, `Npp8u` *pDst, int nDstStep, `NppiSize` oSizeROI, `Npp32f` nMin, `Npp32f` nMax)
Single channel 32-bit floating point to 8-bit unsigned conversion.
- `NppStatus nppiScale_32f8u_C3R` (const `Npp32f` *pSrc, int nSrcStep, `Npp8u` *pDst, int nDstStep, `NppiSize` oSizeROI, `Npp32f` nMin, `Npp32f` nMax)
Three channel 32-bit floating point to 8-bit unsigned conversion.
- `NppStatus nppiScale_32f8u_C4R` (const `Npp32f` *pSrc, int nSrcStep, `Npp8u` *pDst, int nDstStep, `NppiSize` oSizeROI, `Npp32f` nMin, `Npp32f` nMax)
Four channel 32-bit floating point to 8-bit unsigned conversion.
- `NppStatus nppiScale_32f8u_AC4R` (const `Npp32f` *pSrc, int nSrcStep, `Npp8u` *pDst, int nDstStep, `NppiSize` oSizeROI, `Npp32f` nMin, `Npp32f` nMax)
Four channel 32-bit floating point to 8-bit unsigned conversion, not affecting Alpha.

7.9.1 Function Documentation

7.9.1.1 `NppStatus nppiScale_16s8u_AC4R (const Npp16s * pSrc, int nSrcStep, Npp8u * pDst, int nDstStep, NppiSize oSizeROI, NppHintAlgorithm hint)`

Four channel 16-bit signed to 8-bit unsigned conversion, not affecting Alpha.

Parameters:

pSrc Source-Image Pointer.

nSrcStep Source-Image Line Step.

pDst Destination-Image Pointer.

nDstStep Destination-Image Line Step.

oSizeROI Region-of-Interest (ROI).

hint algorithm performance or accuracy selector, currently ignored

Returns:

[Image Data Related Error Codes](#), [ROI Related Error Codes](#)

7.9.1.2 `NppStatus nppiScale_16s8u_C1R (const Npp16s * pSrc, int nSrcStep, Npp8u * pDst, int nDstStep, NppiSize oSizeROI, NppHintAlgorithm hint)`

Single channel 16-bit signed to 8-bit unsigned conversion.

Parameters:

pSrc Source-Image Pointer.

nSrcStep Source-Image Line Step.

pDst Destination-Image Pointer.

nDstStep Destination-Image Line Step.

oSizeROI Region-of-Interest (ROI).

hint algorithm performance or accuracy selector, currently ignored

Returns:

[Image Data Related Error Codes](#), [ROI Related Error Codes](#)

7.9.1.3 `NppStatus nppiScale_16s8u_C3R (const Npp16s * pSrc, int nSrcStep, Npp8u * pDst, int nDstStep, NppiSize oSizeROI, NppHintAlgorithm hint)`

Three channel 16-bit signed to 8-bit unsigned conversion.

Parameters:

pSrc Source-Image Pointer.

nSrcStep Source-Image Line Step.

pDst Destination-Image Pointer.

nDstStep Destination-Image Line Step.

oSizeROI Region-of-Interest (ROI).

hint algorithm performance or accuracy selector, currently ignored

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.9.1.4 NppStatus nppiScale_16s8u_C4R (const Npp16s * pSrc, int nSrcStep, Npp8u * pDst, int nDstStep, NppiSize oSizeROI, NppHintAlgorithm hint)

Four channel 16-bit signed to 8-bit unsigned conversion.

Parameters:

pSrc Source-Image Pointer.

nSrcStep Source-Image Line Step.

pDst Destination-Image Pointer.

nDstStep Destination-Image Line Step.

oSizeROI Region-of-Interest (ROI).

hint algorithm performance or accuracy selector, currently ignored

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.9.1.5 NppStatus nppiScale_16u8u_AC4R (const Npp16u * pSrc, int nSrcStep, Npp8u * pDst, int nDstStep, NppiSize oSizeROI, NppHintAlgorithm hint)

Four channel 16-bit unsigned to 8-bit unsigned conversion, not affecting Alpha.

Parameters:

pSrc Source-Image Pointer.

nSrcStep Source-Image Line Step.

pDst Destination-Image Pointer.

nDstStep Destination-Image Line Step.

oSizeROI Region-of-Interest (ROI).

hint algorithm performance or accuracy selector, currently ignored

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.9.1.6 NppStatus nppiScale_16u8u_C1R (const Npp16u * pSrc, int nSrcStep, Npp8u * pDst, int nDstStep, NppiSize oSizeROI, NppHintAlgorithm hint)

Single channel 16-bit unsigned to 8-bit unsigned conversion.

Parameters:

pSrc Source-Image Pointer.

nSrcStep Source-Image Line Step.

pDst Destination-Image Pointer.

nDstStep Destination-Image Line Step.

oSizeROI Region-of-Interest (ROI).

hint algorithm performance or accuracy selector, currently ignored

Returns:

[Image Data Related Error Codes](#), [ROI Related Error Codes](#)

7.9.1.7 NppStatus nppiScale_16u8u_C3R (const Npp16u * pSrc, int nSrcStep, Npp8u * pDst, int nDstStep, NppiSize oSizeROI, NppHintAlgorithm hint)

Three channel 16-bit unsigned to 8-bit unsigned conversion.

Parameters:

pSrc Source-Image Pointer.

nSrcStep Source-Image Line Step.

pDst Destination-Image Pointer.

nDstStep Destination-Image Line Step.

oSizeROI Region-of-Interest (ROI).

hint algorithm performance or accuracy selector, currently ignored

Returns:

[Image Data Related Error Codes](#), [ROI Related Error Codes](#)

7.9.1.8 NppStatus nppiScale_16u8u_C4R (const Npp16u * pSrc, int nSrcStep, Npp8u * pDst, int nDstStep, NppiSize oSizeROI, NppHintAlgorithm hint)

Four channel 16-bit unsigned to 8-bit unsigned conversion.

Parameters:

pSrc Source-Image Pointer.

nSrcStep Source-Image Line Step.

pDst Destination-Image Pointer.

nDstStep Destination-Image Line Step.

oSizeROI Region-of-Interest (ROI).

hint algorithm performance or accuracy selector, currently ignored

Returns:

[Image Data Related Error Codes](#), [ROI Related Error Codes](#)

7.9.1.9 `NppStatus nppiScale_32f8u_AC4R (const Npp32f * pSrc, int nSrcStep, Npp8u * pDst, int nDstStep, NppiSize oSizeROI, Npp32f nMin, Npp32f nMax)`

Four channel 32-bit floating point to 8-bit unsigned conversion, not affecting Alpha.

Parameters:

pSrc Source-Image Pointer.

nSrcStep Source-Image Line Step.

pDst Destination-Image Pointer.

nDstStep Destination-Image Line Step.

oSizeROI Region-of-Interest (ROI).

nMin specifies the minimum saturation value to which every output value will be clamped.

nMax specifies the maximum saturation value to which every output value will be clamped.

Returns:

Image Data Related Error Codes, ROI Related Error Codes, `NPP_SCALE_RANGE_ERROR` indicates an error condition if $nMax \leq nMin$.

7.9.1.10 `NppStatus nppiScale_32f8u_C1R (const Npp32f * pSrc, int nSrcStep, Npp8u * pDst, int nDstStep, NppiSize oSizeROI, Npp32f nMin, Npp32f nMax)`

Single channel 32-bit floating point to 8-bit unsigned conversion.

Parameters:

pSrc Source-Image Pointer.

nSrcStep Source-Image Line Step.

pDst Destination-Image Pointer.

nDstStep Destination-Image Line Step.

oSizeROI Region-of-Interest (ROI).

nMin specifies the minimum saturation value to which every output value will be clamped.

nMax specifies the maximum saturation value to which every output value will be clamped.

Returns:

Image Data Related Error Codes, ROI Related Error Codes, `NPP_SCALE_RANGE_ERROR` indicates an error condition if $nMax \leq nMin$.

7.9.1.11 `NppStatus nppiScale_32f8u_C3R (const Npp32f * pSrc, int nSrcStep, Npp8u * pDst, int nDstStep, NppiSize oSizeROI, Npp32f nMin, Npp32f nMax)`

Three channel 32-bit floating point to 8-bit unsigned conversion.

Parameters:

pSrc Source-Image Pointer.

nSrcStep Source-Image Line Step.

pDst Destination-Image Pointer.

nDstStep Destination-Image Line Step.

oSizeROI Region-of-Interest (ROI).

nMin specifies the minimum saturation value to which every output value will be clamped.

nMax specifies the maximum saturation value to which every output value will be clamped.

Returns:

Image Data Related Error Codes, ROI Related Error Codes, `NPP_SCALE_RANGE_ERROR` indicates an error condition if $nMax \leq nMin$.

7.9.1.12 `NppStatus nppiScale_32f8u_C4R (const Npp32f * pSrc, int nSrcStep, Npp8u * pDst, int nDstStep, NppiSize oSizeROI, Npp32f nMin, Npp32f nMax)`

Four channel 32-bit floating point to 8-bit unsigned conversion.

Parameters:

pSrc Source-Image Pointer.

nSrcStep Source-Image Line Step.

pDst Destination-Image Pointer.

nDstStep Destination-Image Line Step.

oSizeROI Region-of-Interest (ROI).

nMin specifies the minimum saturation value to which every output value will be clamped.

nMax specifies the maximum saturation value to which every output value will be clamped.

Returns:

Image Data Related Error Codes, ROI Related Error Codes, `NPP_SCALE_RANGE_ERROR` indicates an error condition if $nMax \leq nMin$.

7.9.1.13 `NppStatus nppiScale_32s8u_AC4R (const Npp32s * pSrc, int nSrcStep, Npp8u * pDst, int nDstStep, NppiSize oSizeROI, NppHintAlgorithm hint)`

Four channel 32-bit signed to 8-bit unsigned conversion, not affecting Alpha.

Parameters:

pSrc Source-Image Pointer.

nSrcStep Source-Image Line Step.

pDst Destination-Image Pointer.

nDstStep Destination-Image Line Step.

oSizeROI Region-of-Interest (ROI).

hint algorithm performance or accuracy selector, currently ignored

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.9.1.14 `NppStatus nppiScale_32s8u_C1R (const Npp32s * pSrc, int nSrcStep, Npp8u * pDst, int nDstStep, NppiSize oSizeROI, NppHintAlgorithm hint)`

Single channel 32-bit signed to 8-bit unsigned conversion.

Parameters:

- pSrc* Source-Image Pointer.
- nSrcStep* Source-Image Line Step.
- pDst* Destination-Image Pointer.
- nDstStep* Destination-Image Line Step.
- oSizeROI* Region-of-Interest (ROI).
- hint* algorithm performance or accuracy selector, currently ignored

Returns:

[Image Data Related Error Codes](#), [ROI Related Error Codes](#)

7.9.1.15 `NppStatus nppiScale_32s8u_C3R (const Npp32s * pSrc, int nSrcStep, Npp8u * pDst, int nDstStep, NppiSize oSizeROI, NppHintAlgorithm hint)`

Three channel 32-bit signed to 8-bit unsigned conversion.

Parameters:

- pSrc* Source-Image Pointer.
- nSrcStep* Source-Image Line Step.
- pDst* Destination-Image Pointer.
- nDstStep* Destination-Image Line Step.
- oSizeROI* Region-of-Interest (ROI).
- hint* algorithm performance or accuracy selector, currently ignored

Returns:

[Image Data Related Error Codes](#), [ROI Related Error Codes](#)

7.9.1.16 `NppStatus nppiScale_32s8u_C4R (const Npp32s * pSrc, int nSrcStep, Npp8u * pDst, int nDstStep, NppiSize oSizeROI, NppHintAlgorithm hint)`

Four channel 32-bit signed to 8-bit unsigned conversion.

Parameters:

- pSrc* Source-Image Pointer.
- nSrcStep* Source-Image Line Step.
- pDst* Destination-Image Pointer.
- nDstStep* Destination-Image Line Step.
- oSizeROI* Region-of-Interest (ROI).
- hint* algorithm performance or accuracy selector, currently ignored

Returns:

[Image Data Related Error Codes](#), [ROI Related Error Codes](#)

7.9.1.17 NppStatus nppiScale_8u16s_AC4R (const Npp8u * pSrc, int nSrcStep, Npp16s * pDst, int nDstStep, NppiSize oSizeROI)

Four channel 8-bit unsigned to 16-bit signed conversion, not affecting Alpha.

Parameters:

pSrc Source-Image Pointer.
nSrcStep Source-Image Line Step.
pDst Destination-Image Pointer.
nDstStep Destination-Image Line Step.
oSizeROI Region-of-Interest (ROI).

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.9.1.18 NppStatus nppiScale_8u16s_C1R (const Npp8u * pSrc, int nSrcStep, Npp16s * pDst, int nDstStep, NppiSize oSizeROI)

Single channel 8-bit unsigned to 16-bit signed conversion.

Parameters:

pSrc Source-Image Pointer.
nSrcStep Source-Image Line Step.
pDst Destination-Image Pointer.
nDstStep Destination-Image Line Step.
oSizeROI Region-of-Interest (ROI).

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.9.1.19 NppStatus nppiScale_8u16s_C3R (const Npp8u * pSrc, int nSrcStep, Npp16s * pDst, int nDstStep, NppiSize oSizeROI)

Three channel 8-bit unsigned to 16-bit signed conversion.

Parameters:

pSrc Source-Image Pointer.
nSrcStep Source-Image Line Step.
pDst Destination-Image Pointer.
nDstStep Destination-Image Line Step.
oSizeROI Region-of-Interest (ROI).

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.9.1.20 NppStatus nppiScale_8u16s_C4R (const Npp8u * pSrc, int nSrcStep, Npp16s * pDst, int nDstStep, NppiSize oSizeROI)

Four channel 8-bit unsigned to 16-bit signed conversion.

Parameters:

pSrc Source-Image Pointer.
nSrcStep Source-Image Line Step.
pDst Destination-Image Pointer.
nDstStep Destination-Image Line Step.
oSizeROI Region-of-Interest (ROI).

Returns:

[Image Data Related Error Codes](#), [ROI Related Error Codes](#)

7.9.1.21 NppStatus nppiScale_8u16u_AC4R (const Npp8u * pSrc, int nSrcStep, Npp16u * pDst, int nDstStep, NppiSize oSizeROI)

Four channel 8-bit unsigned to 16-bit unsigned conversion, not affecting Alpha.

Parameters:

pSrc Source-Image Pointer.
nSrcStep Source-Image Line Step.
pDst Destination-Image Pointer.
nDstStep Destination-Image Line Step.
oSizeROI Region-of-Interest (ROI).

Returns:

[Image Data Related Error Codes](#), [ROI Related Error Codes](#)

7.9.1.22 NppStatus nppiScale_8u16u_C1R (const Npp8u * pSrc, int nSrcStep, Npp16u * pDst, int nDstStep, NppiSize oSizeROI)

Single channel 8-bit unsigned to 16-bit unsigned conversion.

Parameters:

pSrc Source-Image Pointer.
nSrcStep Source-Image Line Step.
pDst Destination-Image Pointer.
nDstStep Destination-Image Line Step.
oSizeROI Region-of-Interest (ROI).

Returns:

[Image Data Related Error Codes](#), [ROI Related Error Codes](#)

7.9.1.23 NppStatus nppiScale_8u16u_C3R (const Npp8u * pSrc, int nSrcStep, Npp16u * pDst, int nDstStep, NppiSize oSizeROI)

Three channel 8-bit unsigned to 16-bit unsigned conversion.

Parameters:

pSrc Source-Image Pointer.
nSrcStep Source-Image Line Step.
pDst Destination-Image Pointer.
nDstStep Destination-Image Line Step.
oSizeROI Region-of-Interest (ROI).

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.9.1.24 NppStatus nppiScale_8u16u_C4R (const Npp8u * pSrc, int nSrcStep, Npp16u * pDst, int nDstStep, NppiSize oSizeROI)

Four channel 8-bit unsigned to 16-bit unsigned conversion.

Parameters:

pSrc Source-Image Pointer.
nSrcStep Source-Image Line Step.
pDst Destination-Image Pointer.
nDstStep Destination-Image Line Step.
oSizeROI Region-of-Interest (ROI).

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.9.1.25 NppStatus nppiScale_8u32f_AC4R (const Npp8u * pSrc, int nSrcStep, Npp32f * pDst, int nDstStep, NppiSize oSizeROI, Npp32f nMin, Npp32f nMax)

Four channel 8-bit unsigned to 32-bit floating-point conversion, not affecting Alpha.

Parameters:

pSrc Source-Image Pointer.
nSrcStep Source-Image Line Step.
pDst Destination-Image Pointer.
nDstStep Destination-Image Line Step.
oSizeROI Region-of-Interest (ROI).
nMin specifies the minimum saturation value to which every output value will be clamped.
nMax specifies the maximum saturation value to which every output value will be clamped.

Returns:

Image Data Related Error Codes, ROI Related Error Codes, `NPP_SCALE_RANGE_ERROR` indicates an error condition if $nMax \leq nMin$.

7.9.1.26 NppStatus nppiScale_8u32f_C1R (const Npp8u * pSrc, int nSrcStep, Npp32f * pDst, int nDstStep, NppiSize oSizeROI, Npp32f nMin, Npp32f nMax)

Single channel 8-bit unsigned to 32-bit floating-point conversion.

Parameters:

pSrc Source-Image Pointer.

nSrcStep Source-Image Line Step.

pDst Destination-Image Pointer.

nDstStep Destination-Image Line Step.

oSizeROI Region-of-Interest (ROI).

nMin specifies the minimum saturation value to which every output value will be clamped.

nMax specifies the maximum saturation value to which every output value will be clamped.

Returns:

Image Data Related Error Codes, ROI Related Error Codes, `NPP_SCALE_RANGE_ERROR` indicates an error condition if $nMax \leq nMin$.

7.9.1.27 NppStatus nppiScale_8u32f_C3R (const Npp8u * pSrc, int nSrcStep, Npp32f * pDst, int nDstStep, NppiSize oSizeROI, Npp32f nMin, Npp32f nMax)

Three channel 8-bit unsigned to 32-bit floating-point conversion.

Parameters:

pSrc Source-Image Pointer.

nSrcStep Source-Image Line Step.

pDst Destination-Image Pointer.

nDstStep Destination-Image Line Step.

oSizeROI Region-of-Interest (ROI).

nMin specifies the minimum saturation value to which every output value will be clamped.

nMax specifies the maximum saturation value to which every output value will be clamped.

Returns:

Image Data Related Error Codes, ROI Related Error Codes, `NPP_SCALE_RANGE_ERROR` indicates an error condition if $nMax \leq nMin$.

7.9.1.28 NppStatus nppiScale_8u32f_C4R (const Npp8u * pSrc, int nSrcStep, Npp32f * pDst, int nDstStep, NppiSize oSizeROI, Npp32f nMin, Npp32f nMax)

Four channel 8-bit unsigned to 32-bit floating-point conversion.

Parameters:

pSrc Source-Image Pointer.

nSrcStep Source-Image Line Step.

pDst Destination-Image Pointer.

nDstStep Destination-Image Line Step.

oSizeROI Region-of-Interest (ROI).

nMin specifies the minimum saturation value to which every output value will be clamped.

nMax specifies the maximum saturation value to which every output value will be clamped.

Returns:

Image Data Related Error Codes, ROI Related Error Codes, `NPP_SCALE_RANGE_ERROR` indicates an error condition if $nMax \leq nMin$.

7.9.1.29 NppStatus nppiScale_8u32s_AC4R (const Npp8u * pSrc, int nSrcStep, Npp32s * pDst, int nDstStep, NppiSize oSizeROI)

Four channel 8-bit unsigned to 32-bit signed conversion, not affecting Alpha.

Parameters:

pSrc Source-Image Pointer.

nSrcStep Source-Image Line Step.

pDst Destination-Image Pointer.

nDstStep Destination-Image Line Step.

oSizeROI Region-of-Interest (ROI).

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.9.1.30 NppStatus nppiScale_8u32s_C1R (const Npp8u * pSrc, int nSrcStep, Npp32s * pDst, int nDstStep, NppiSize oSizeROI)

Single channel 8-bit unsigned to 32-bit signed conversion.

Parameters:

pSrc Source-Image Pointer.

nSrcStep Source-Image Line Step.

pDst Destination-Image Pointer.

nDstStep Destination-Image Line Step.

oSizeROI Region-of-Interest (ROI).

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.9.1.31 NppStatus nppiScale_8u32s_C3R (const Npp8u * pSrc, int nSrcStep, Npp32s * pDst, int nDstStep, NppiSize oSizeROI)

Three channel 8-bit unsigned to 32-bit signed conversion.

Parameters:

- pSrc* Source-Image Pointer.
- nSrcStep* Source-Image Line Step.
- pDst* Destination-Image Pointer.
- nDstStep* Destination-Image Line Step.
- oSizeROI* Region-of-Interest (ROI).

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.9.1.32 NppStatus nppiScale_8u32s_C4R (const Npp8u * pSrc, int nSrcStep, Npp32s * pDst, int nDstStep, NppiSize oSizeROI)

Four channel 8-bit unsigned to 32-bit signed conversion.

Parameters:

- pSrc* Source-Image Pointer.
- nSrcStep* Source-Image Line Step.
- pDst* Destination-Image Pointer.
- nDstStep* Destination-Image Line Step.
- oSizeROI* Region-of-Interest (ROI).

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.10 Copy Constant Border

CopyConstBorder

Methods for copying images and padding borders with a constant, user-specifiable color.

- `NppStatus nppiCopyConstBorder_8u_C1R` (const `Npp8u` *pSrc, int nSrcStep, `NppiSize` oSrcSizeROI, `Npp8u` *pDst, int nDstStep, `NppiSize` oDstSizeROI, int nTopBorderHeight, int nLeftBorderWidth, `Npp8u` nValue)

1 channel 8-bit unsigned integer image copy with constant border color.
- `NppStatus nppiCopyConstBorder_8u_C3R` (const `Npp8u` *pSrc, int nSrcStep, `NppiSize` oSrcSizeROI, `Npp8u` *pDst, int nDstStep, `NppiSize` oDstSizeROI, int nTopBorderHeight, int nLeftBorderWidth, const `Npp8u` aValue[3])

3 channel 8-bit unsigned integer image copy with constant border color.
- `NppStatus nppiCopyConstBorder_8u_C4R` (const `Npp8u` *pSrc, int nSrcStep, `NppiSize` oSrcSizeROI, `Npp8u` *pDst, int nDstStep, `NppiSize` oDstSizeROI, int nTopBorderHeight, int nLeftBorderWidth, const `Npp8u` aValue[4])

4 channel 8-bit unsigned integer image copy with constant border color.
- `NppStatus nppiCopyConstBorder_8u_AC4R` (const `Npp8u` *pSrc, int nSrcStep, `NppiSize` oSrcSizeROI, `Npp8u` *pDst, int nDstStep, `NppiSize` oDstSizeROI, int nTopBorderHeight, int nLeftBorderWidth, const `Npp8u` aValue[3])

4 channel 8-bit unsigned integer image copy with constant border color with alpha channel unaffected.
- `NppStatus nppiCopyConstBorder_16u_C1R` (const `Npp16u` *pSrc, int nSrcStep, `NppiSize` oSrcSizeROI, `Npp16u` *pDst, int nDstStep, `NppiSize` oDstSizeROI, int nTopBorderHeight, int nLeftBorderWidth, `Npp16u` nValue)

1 channel 16-bit unsigned integer image copy with constant border color.
- `NppStatus nppiCopyConstBorder_16u_C3R` (const `Npp16u` *pSrc, int nSrcStep, `NppiSize` oSrcSizeROI, `Npp16u` *pDst, int nDstStep, `NppiSize` oDstSizeROI, int nTopBorderHeight, int nLeftBorderWidth, const `Npp16u` aValue[3])

3 channel 16-bit unsigned integer image copy with constant border color.
- `NppStatus nppiCopyConstBorder_16u_C4R` (const `Npp16u` *pSrc, int nSrcStep, `NppiSize` oSrcSizeROI, `Npp16u` *pDst, int nDstStep, `NppiSize` oDstSizeROI, int nTopBorderHeight, int nLeftBorderWidth, const `Npp16u` aValue[4])

4 channel 16-bit unsigned integer image copy with constant border color.
- `NppStatus nppiCopyConstBorder_16u_AC4R` (const `Npp16u` *pSrc, int nSrcStep, `NppiSize` oSrcSizeROI, `Npp16u` *pDst, int nDstStep, `NppiSize` oDstSizeROI, int nTopBorderHeight, int nLeftBorderWidth, const `Npp16u` aValue[3])

4 channel 16-bit unsigned integer image copy with constant border color with alpha channel unaffected.
- `NppStatus nppiCopyConstBorder_16s_C1R` (const `Npp16s` *pSrc, int nSrcStep, `NppiSize` oSrcSizeROI, `Npp16s` *pDst, int nDstStep, `NppiSize` oDstSizeROI, int nTopBorderHeight, int nLeftBorderWidth, `Npp16s` nValue)

1 channel 16-bit signed integer image copy with constant border color.

- **NppStatus nppiCopyConstBorder_16s_C3R** (const **Npp16s** *pSrc, int nSrcStep, **NppiSize** oSrcSizeROI, **Npp16s** *pDst, int nDstStep, **NppiSize** oDstSizeROI, int nTopBorderHeight, int nLeftBorderWidth, const **Npp16s** aValue[3])

3 channel 16-bit signed integer image copy with constant border color.
- **NppStatus nppiCopyConstBorder_16s_C4R** (const **Npp16s** *pSrc, int nSrcStep, **NppiSize** oSrcSizeROI, **Npp16s** *pDst, int nDstStep, **NppiSize** oDstSizeROI, int nTopBorderHeight, int nLeftBorderWidth, const **Npp16s** aValue[4])

4 channel 16-bit signed integer image copy with constant border color.
- **NppStatus nppiCopyConstBorder_16s_AC4R** (const **Npp16s** *pSrc, int nSrcStep, **NppiSize** oSrcSizeROI, **Npp16s** *pDst, int nDstStep, **NppiSize** oDstSizeROI, int nTopBorderHeight, int nLeftBorderWidth, const **Npp16s** aValue[3])

4 channel 16-bit signed integer image copy with constant border color with alpha channel unaffected.
- **NppStatus nppiCopyConstBorder_32s_C1R** (const **Npp32s** *pSrc, int nSrcStep, **NppiSize** oSrcSizeROI, **Npp32s** *pDst, int nDstStep, **NppiSize** oDstSizeROI, int nTopBorderHeight, int nLeftBorderWidth, **Npp32s** nValue)

1 channel 32-bit signed integer image copy with constant border color.
- **NppStatus nppiCopyConstBorder_32s_C3R** (const **Npp32s** *pSrc, int nSrcStep, **NppiSize** oSrcSizeROI, **Npp32s** *pDst, int nDstStep, **NppiSize** oDstSizeROI, int nTopBorderHeight, int nLeftBorderWidth, const **Npp32s** aValue[3])

3 channel 32-bit signed integer image copy with constant border color.
- **NppStatus nppiCopyConstBorder_32s_C4R** (const **Npp32s** *pSrc, int nSrcStep, **NppiSize** oSrcSizeROI, **Npp32s** *pDst, int nDstStep, **NppiSize** oDstSizeROI, int nTopBorderHeight, int nLeftBorderWidth, const **Npp32s** aValue[4])

4 channel 32-bit signed integer image copy with constant border color.
- **NppStatus nppiCopyConstBorder_32s_AC4R** (const **Npp32s** *pSrc, int nSrcStep, **NppiSize** oSrcSizeROI, **Npp32s** *pDst, int nDstStep, **NppiSize** oDstSizeROI, int nTopBorderHeight, int nLeftBorderWidth, const **Npp32s** aValue[3])

4 channel 32-bit signed integer image copy with constant border color with alpha channel unaffected.
- **NppStatus nppiCopyConstBorder_32f_C1R** (const **Npp32f** *pSrc, int nSrcStep, **NppiSize** oSrcSizeROI, **Npp32f** *pDst, int nDstStep, **NppiSize** oDstSizeROI, int nTopBorderHeight, int nLeftBorderWidth, **Npp32f** nValue)

1 channel 32-bit floating point image copy with constant border color.
- **NppStatus nppiCopyConstBorder_32f_C3R** (const **Npp32f** *pSrc, int nSrcStep, **NppiSize** oSrcSizeROI, **Npp32f** *pDst, int nDstStep, **NppiSize** oDstSizeROI, int nTopBorderHeight, int nLeftBorderWidth, const **Npp32f** aValue[3])

3 channel 32-bit floating point image copy with constant border color.
- **NppStatus nppiCopyConstBorder_32f_C4R** (const **Npp32f** *pSrc, int nSrcStep, **NppiSize** oSrcSizeROI, **Npp32f** *pDst, int nDstStep, **NppiSize** oDstSizeROI, int nTopBorderHeight, int nLeftBorderWidth, const **Npp32f** aValue[4])

4 channel 32-bit floating point image copy with constant border color.

- `NppStatus nppiCopyConstBorder_32f_AC4R` (const `Npp32f *pSrc`, int `nSrcStep`, `NppiSize` `oSrcSizeROI`, `Npp32f *pDst`, int `nDstStep`, `NppiSize` `oDstSizeROI`, int `nTopBorderHeight`, int `nLeftBorderWidth`, const `Npp32f aValue[3]`)

4 channel 32-bit floating point image copy with constant border color with alpha channel unaffected.

7.10.1 Function Documentation

- 7.10.1.1** `NppStatus nppiCopyConstBorder_16s_AC4R` (const `Npp16s *pSrc`, int `nSrcStep`, `NppiSize` `oSrcSizeROI`, `Npp16s *pDst`, int `nDstStep`, `NppiSize` `oDstSizeROI`, int `nTopBorderHeight`, int `nLeftBorderWidth`, const `Npp16s aValue[3]`)

4 channel 16-bit signed integer image copy with constant border color with alpha channel unaffected.

See `nppiCopyConstBorder_16s_C1R()` for detailed documentation.

Parameters:

pSrc Source-Image Pointer.

nSrcStep Source-Image Line Step.

oSrcSizeROI Size of the source region-of-interest.

pDst Destination-Image Pointer.

nDstStep Destination-Image Line Step.

oDstSizeROI Size of the destination region-of-interest.

nTopBorderHeight Height of top border.

nLeftBorderWidth Width of left border.

aValue Vector of the RGB values of the border pixels. Because this method does not affect the destination image's alpha channel, only three components of the border color are needed.

Returns:

[Image Data Related Error Codes](#), [ROI Related Error Codes](#)

- 7.10.1.2** `NppStatus nppiCopyConstBorder_16s_C1R` (const `Npp16s *pSrc`, int `nSrcStep`, `NppiSize` `oSrcSizeROI`, `Npp16s *pDst`, int `nDstStep`, `NppiSize` `oDstSizeROI`, int `nTopBorderHeight`, int `nLeftBorderWidth`, `Npp16s nValue`)

1 channel 16-bit signed integer image copy with constant border color.

Parameters:

pSrc Source-Image Pointer.

nSrcStep Source-Image Line Step.

oSrcSizeROI Size of the source region of pixels.

pDst Destination-Image Pointer.

nDstStep Destination-Image Line Step.

oDstSizeROI Size (width, height) of the destination region, i.e. the region that gets filled with data from the source image (inner part) and constant border color (outer part).

nTopBorderHeight Height (in pixels) of the top border. The number of pixel rows at the top of the destination ROI that will be filled with the constant border color. $nBottomBorderHeight = oDstSizeROI.height - nTopBorderHeight - oSrcSizeROI.height$.

nLeftBorderWidth Width (in pixels) of the left border. The width of the border at the right side of the destination ROI is implicitly defined by the size of the source ROI: $nRightBorderWidth = oDstSizeROI.width - nLeftBorderWidth - oSrcSizeROI.width$.

nValue The pixel value to be set for border pixels.

Returns:

[Image Data Related Error Codes](#), [ROI Related Error Codes](#)

7.10.1.3 `NppStatus nppiCopyConstBorder_16s_C3R (const Npp16s * pSrc, int nSrcStep, NppiSize oSrcSizeROI, Npp16s * pDst, int nDstStep, NppiSize oDstSizeROI, int nTopBorderHeight, int nLeftBorderWidth, const Npp16s aValue[3])`

3 channel 16-bit signed integer image copy with constant border color.

See [nppiCopyConstBorder_16s_C1R\(\)](#) for detailed documentation.

Parameters:

pSrc Source-Image Pointer.

nSrcStep Source-Image Line Step.

oSrcSizeROI Size of the source region-of-interest.

pDst Destination-Image Pointer.

nDstStep Destination-Image Line Step.

oDstSizeROI Size of the destination region-of-interest.

nTopBorderHeight Height of top border.

nLeftBorderWidth Width of left border.

aValue Vector of the RGBA values of the border pixels to be set.

Returns:

[Image Data Related Error Codes](#), [ROI Related Error Codes](#)

7.10.1.4 `NppStatus nppiCopyConstBorder_16s_C4R (const Npp16s * pSrc, int nSrcStep, NppiSize oSrcSizeROI, Npp16s * pDst, int nDstStep, NppiSize oDstSizeROI, int nTopBorderHeight, int nLeftBorderWidth, const Npp16s aValue[4])`

4 channel 16-bit signed integer image copy with constant border color.

See [nppiCopyConstBorder_16s_C1R\(\)](#) for detailed documentation.

Parameters:

pSrc Source-Image Pointer.

nSrcStep Source-Image Line Step.

oSrcSizeROI Size of the source region-of-interest.

pDst Destination-Image Pointer.

nDstStep Destination-Image Line Step.

oDstSizeROI Size of the destination region-of-interest.

nTopBorderHeight Height of top border.

nLeftBorderWidth Width of left border.

aValue Vector of the RGBA values of the border pixels to be set.

Returns:

[Image Data Related Error Codes](#), [ROI Related Error Codes](#)

7.10.1.5 NppStatus nppiCopyConstBorder_16u_AC4R (const Npp16u * pSrc, int nSrcStep, NppiSize oSrcSizeROI, Npp16u * pDst, int nDstStep, NppiSize oDstSizeROI, int nTopBorderHeight, int nLeftBorderWidth, const Npp16u aValue[3])

4 channel 16-bit unsigned integer image copy with constant border color with alpha channel unaffected.

See [nppiCopyConstBorder_16u_C1R\(\)](#) for detailed documentation.

Parameters:

pSrc Source-Image Pointer.

nSrcStep Source-Image Line Step.

oSrcSizeROI Size of the source region-of-interest.

pDst Destination-Image Pointer.

nDstStep Destination-Image Line Step.

oDstSizeROI Size of the destination region-of-interest.

nTopBorderHeight Height of top border.

nLeftBorderWidth Width of left border.

aValue Vector of the RGB values of the border pixels. Because this method does not affect the destination image's alpha channel, only three components of the border color are needed.

Returns:

[Image Data Related Error Codes](#), [ROI Related Error Codes](#)

7.10.1.6 NppStatus nppiCopyConstBorder_16u_C1R (const Npp16u * pSrc, int nSrcStep, NppiSize oSrcSizeROI, Npp16u * pDst, int nDstStep, NppiSize oDstSizeROI, int nTopBorderHeight, int nLeftBorderWidth, Npp16u nValue)

1 channel 16-bit unsigned integer image copy with constant border color.

Parameters:

pSrc Source-Image Pointer.

nSrcStep Source-Image Line Step.

oSrcSizeROI Size of the source region of pixels.

pDst Destination-Image Pointer.

nDstStep Destination-Image Line Step.

oDstSizeROI Size (width, height) of the destination region, i.e. the region that gets filled with data from the source image (inner part) and constant border color (outer part).

nTopBorderHeight Height (in pixels) of the top border. The number of pixel rows at the top of the destination ROI that will be filled with the constant border color. $nBottomBorderHeight = oDstSizeROI.height - nTopBorderHeight - oSrcSizeROI.height$.

nLeftBorderWidth Width (in pixels) of the left border. The width of the border at the right side of the destination ROI is implicitly defined by the size of the source ROI: $nRightBorderWidth = oDstSizeROI.width - nLeftBorderWidth - oSrcSizeROI.width$.

nValue The pixel value to be set for border pixels.

Returns:

[Image Data Related Error Codes](#), [ROI Related Error Codes](#)

7.10.1.7 NppStatus nppiCopyConstBorder_16u_C3R (const Npp16u * pSrc, int nSrcStep, NppiSize oSrcSizeROI, Npp16u * pDst, int nDstStep, NppiSize oDstSizeROI, int nTopBorderHeight, int nLeftBorderWidth, const Npp16u aValue[3])

3 channel 16-bit unsigned integer image copy with constant border color.

See [nppiCopyConstBorder_16u_C1R\(\)](#) for detailed documentation.

Parameters:

pSrc [Source-Image Pointer](#).

nSrcStep [Source-Image Line Step](#).

oSrcSizeROI Size of the source region-of-interest.

pDst [Destination-Image Pointer](#).

nDstStep [Destination-Image Line Step](#).

oDstSizeROI Size of the destination region-of-interest.

nTopBorderHeight Height of top border.

nLeftBorderWidth Width of left border.

aValue Vector of the RGBA values of the border pixels to be set.

Returns:

[Image Data Related Error Codes](#), [ROI Related Error Codes](#)

7.10.1.8 NppStatus nppiCopyConstBorder_16u_C4R (const Npp16u * pSrc, int nSrcStep, NppiSize oSrcSizeROI, Npp16u * pDst, int nDstStep, NppiSize oDstSizeROI, int nTopBorderHeight, int nLeftBorderWidth, const Npp16u aValue[4])

4 channel 16-bit unsigned integer image copy with constant border color.

See [nppiCopyConstBorder_16u_C1R\(\)](#) for detailed documentation.

Parameters:

pSrc [Source-Image Pointer](#).

nSrcStep [Source-Image Line Step](#).

oSrcSizeROI Size of the source region-of-interest.
pDst Destination-Image Pointer.
nDstStep Destination-Image Line Step.
oDstSizeROI Size of the destination region-of-interest.
nTopBorderHeight Height of top border.
nLeftBorderWidth Width of left border.
aValue Vector of the RGBA values of the border pixels to be set.

Returns:

[Image Data Related Error Codes](#), [ROI Related Error Codes](#)

7.10.1.9 **NppStatus nppiCopyConstBorder_32f_AC4R** (const Npp32f * *pSrc*, int *nSrcStep*, NppiSize *oSrcSizeROI*, Npp32f * *pDst*, int *nDstStep*, NppiSize *oDstSizeROI*, int *nTopBorderHeight*, int *nLeftBorderWidth*, const Npp32f *aValue*[3])

4 channel 32-bit floating point image copy with constant border color with alpha channel unaffected.

See [nppiCopyConstBorder_32f_C1R\(\)](#) for detailed documentation.

Parameters:

pSrc Source-Image Pointer.
nSrcStep Source-Image Line Step.
oSrcSizeROI Size of the source region-of-interest.
pDst Destination-Image Pointer.
nDstStep Destination-Image Line Step.
oDstSizeROI Size of the destination region-of-interest.
nTopBorderHeight Height of top border.
nLeftBorderWidth Width of left border.
aValue Vector of the RGB values of the border pixels. Because this method does not affect the destination image's alpha channel, only three components of the border color are needed.

Returns:

[Image Data Related Error Codes](#), [ROI Related Error Codes](#)

7.10.1.10 **NppStatus nppiCopyConstBorder_32f_C1R** (const Npp32f * *pSrc*, int *nSrcStep*, NppiSize *oSrcSizeROI*, Npp32f * *pDst*, int *nDstStep*, NppiSize *oDstSizeROI*, int *nTopBorderHeight*, int *nLeftBorderWidth*, Npp32f *nValue*)

1 channel 32-bit floating point image copy with constant border color.

Parameters:

pSrc Source-Image Pointer.
nSrcStep Source-Image Line Step.
oSrcSizeROI Size of the source region of pixels.

pDst Destination-Image Pointer.

nDstStep Destination-Image Line Step.

oDstSizeROI Size (width, height) of the destination region, i.e. the region that gets filled with data from the source image (inner part) and constant border color (outer part).

nTopBorderHeight Height (in pixels) of the top border. The number of pixel rows at the top of the destination ROI that will be filled with the constant border color. $nBottomBorderHeight = oDstSizeROI.height - nTopBorderHeight - oSrcSizeROI.height$.

nLeftBorderWidth Width (in pixels) of the left border. The width of the border at the right side of the destination ROI is implicitly defined by the size of the source ROI: $nRightBorderWidth = oDstSizeROI.width - nLeftBorderWidth - oSrcSizeROI.width$.

nValue The pixel value to be set for border pixels.

Returns:

[Image Data Related Error Codes](#), [ROI Related Error Codes](#)

7.10.1.11 `NppStatus nppiCopyConstBorder_32f_C3R (const Npp32f * pSrc, int nSrcStep, NppiSize oSrcSizeROI, Npp32f * pDst, int nDstStep, NppiSize oDstSizeROI, int nTopBorderHeight, int nLeftBorderWidth, const Npp32f aValue[3])`

3 channel 32-bit floating point image copy with constant border color.

See [nppiCopyConstBorder_32f_C1R\(\)](#) for detailed documentation.

Parameters:

pSrc Source-Image Pointer.

nSrcStep Source-Image Line Step.

oSrcSizeROI Size of the source region-of-interest.

pDst Destination-Image Pointer.

nDstStep Destination-Image Line Step.

oDstSizeROI Size of the destination region-of-interest.

nTopBorderHeight Height of top border.

nLeftBorderWidth Width of left border.

aValue Vector of the RGBA values of the border pixels to be set.

Returns:

[Image Data Related Error Codes](#), [ROI Related Error Codes](#)

7.10.1.12 `NppStatus nppiCopyConstBorder_32f_C4R (const Npp32f * pSrc, int nSrcStep, NppiSize oSrcSizeROI, Npp32f * pDst, int nDstStep, NppiSize oDstSizeROI, int nTopBorderHeight, int nLeftBorderWidth, const Npp32f aValue[4])`

4 channel 32-bit floating point image copy with constant border color.

See [nppiCopyConstBorder_32f_C1R\(\)](#) for detailed documentation.

Parameters:

pSrc Source-Image Pointer.

nSrcStep Source-Image Line Step.
oSrcSizeROI Size of the source region-of-interest.
pDst Destination-Image Pointer.
nDstStep Destination-Image Line Step.
oDstSizeROI Size of the destination region-of-interest.
nTopBorderHeight Height of top border.
nLeftBorderWidth Width of left border.
aValue Vector of the RGBA values of the border pixels to be set.

Returns:

[Image Data Related Error Codes](#), [ROI Related Error Codes](#)

7.10.1.13 NppStatus nppiCopyConstBorder_32s_AC4R (const Npp32s * pSrc, int nSrcStep, NppiSize oSrcSizeROI, Npp32s * pDst, int nDstStep, NppiSize oDstSizeROI, int nTopBorderHeight, int nLeftBorderWidth, const Npp32s aValue[3])

4 channel 32-bit signed integer image copy with constant border color with alpha channel unaffected.
 See [nppiCopyConstBorder_32s_C1R\(\)](#) for detailed documentation.

Parameters:

pSrc Source-Image Pointer.
nSrcStep Source-Image Line Step.
oSrcSizeROI Size of the source region-of-interest.
pDst Destination-Image Pointer.
nDstStep Destination-Image Line Step.
oDstSizeROI Size of the destination region-of-interest.
nTopBorderHeight Height of top border.
nLeftBorderWidth Width of left border.
aValue Vector of the RGB values of the border pixels. Because this method does not affect the destination image's alpha channel, only three components of the border color are needed.

Returns:

[Image Data Related Error Codes](#), [ROI Related Error Codes](#)

7.10.1.14 NppStatus nppiCopyConstBorder_32s_C1R (const Npp32s * pSrc, int nSrcStep, NppiSize oSrcSizeROI, Npp32s * pDst, int nDstStep, NppiSize oDstSizeROI, int nTopBorderHeight, int nLeftBorderWidth, Npp32s nValue)

1 channel 32-bit signed integer image copy with constant border color.

Parameters:

pSrc Source-Image Pointer.
nSrcStep Source-Image Line Step.

oSrcSizeROI Size of the source region of pixels.

pDst Destination-Image Pointer.

nDstStep Destination-Image Line Step.

oDstSizeROI Size (width, height) of the destination region, i.e. the region that gets filled with data from the source image (inner part) and constant border color (outer part).

nTopBorderHeight Height (in pixels) of the top border. The number of pixel rows at the top of the destination ROI that will be filled with the constant border color. $nBottomBorderHeight = oDstSizeROI.height - nTopBorderHeight - oSrcSizeROI.height$.

nLeftBorderWidth Width (in pixels) of the left border. The width of the border at the right side of the destination ROI is implicitly defined by the size of the source ROI: $nRightBorderWidth = oDstSizeROI.width - nLeftBorderWidth - oSrcSizeROI.width$.

nValue The pixel value to be set for border pixels.

Returns:

[Image Data Related Error Codes](#), [ROI Related Error Codes](#)

7.10.1.15 `NppStatus nppiCopyConstBorder_32s_C3R (const Npp32s * pSrc, int nSrcStep, NppiSize oSrcSizeROI, Npp32s * pDst, int nDstStep, NppiSize oDstSizeROI, int nTopBorderHeight, int nLeftBorderWidth, const Npp32s aValue[3])`

3 channel 32-bit signed integer image copy with constant border color.

See [nppiCopyConstBorder_32s_C1R\(\)](#) for detailed documentation.

Parameters:

pSrc Source-Image Pointer.

nSrcStep Source-Image Line Step.

oSrcSizeROI Size of the source region-of-interest.

pDst Destination-Image Pointer.

nDstStep Destination-Image Line Step.

oDstSizeROI Size of the destination region-of-interest.

nTopBorderHeight Height of top border.

nLeftBorderWidth Width of left border.

aValue Vector of the RGBA values of the border pixels to be set.

Returns:

[Image Data Related Error Codes](#), [ROI Related Error Codes](#)

7.10.1.16 `NppStatus nppiCopyConstBorder_32s_C4R (const Npp32s * pSrc, int nSrcStep, NppiSize oSrcSizeROI, Npp32s * pDst, int nDstStep, NppiSize oDstSizeROI, int nTopBorderHeight, int nLeftBorderWidth, const Npp32s aValue[4])`

4 channel 32-bit signed integer image copy with constant border color.

See [nppiCopyConstBorder_32s_C1R\(\)](#) for detailed documentation.

Parameters:

pSrc Source-Image Pointer.
nSrcStep Source-Image Line Step.
oSrcSizeROI Size of the source region-of-interest.
pDst Destination-Image Pointer.
nDstStep Destination-Image Line Step.
oDstSizeROI Size of the destination region-of-interest.
nTopBorderHeight Height of top border.
nLeftBorderWidth Width of left border.
aValue Vector of the RGBA values of the border pixels to be set.

Returns:

[Image Data Related Error Codes](#), [ROI Related Error Codes](#)

7.10.1.17 `NppStatus nppiCopyConstBorder_8u_AC4R (const Npp8u * pSrc, int nSrcStep, NppiSize oSrcSizeROI, Npp8u * pDst, int nDstStep, NppiSize oDstSizeROI, int nTopBorderHeight, int nLeftBorderWidth, const Npp8u aValue[3])`

4 channel 8-bit unsigned integer image copy with constant border color with alpha channel unaffected.

See [nppiCopyConstBorder_8u_C1R\(\)](#) for detailed documentation.

Parameters:

pSrc Source-Image Pointer.
nSrcStep Source-Image Line Step.
oSrcSizeROI Size of the source region-of-interest.
pDst Destination-Image Pointer.
nDstStep Destination-Image Line Step.
oDstSizeROI Size of the destination region-of-interest.
nTopBorderHeight Height of top border.
nLeftBorderWidth Width of left border.
aValue Vector of the RGB values of the border pixels. Because this method does not affect the destination image's alpha channel, only three components of the border color are needed.

Returns:

[Image Data Related Error Codes](#), [ROI Related Error Codes](#)

7.10.1.18 `NppStatus nppiCopyConstBorder_8u_C1R (const Npp8u * pSrc, int nSrcStep, NppiSize oSrcSizeROI, Npp8u * pDst, int nDstStep, NppiSize oDstSizeROI, int nTopBorderHeight, int nLeftBorderWidth, Npp8u nValue)`

1 channel 8-bit unsigned integer image copy with constant border color.

Parameters:

pSrc Source-Image Pointer.

nSrcStep Source-Image Line Step.

oSrcSizeROI Size of the source region of pixels.

pDst Destination-Image Pointer.

nDstStep Destination-Image Line Step.

oDstSizeROI Size (width, height) of the destination region, i.e. the region that gets filled with data from the source image (inner part) and constant border color (outer part).

nTopBorderHeight Height (in pixels) of the top border. The number of pixel rows at the top of the destination ROI that will be filled with the constant border color. $nBottomBorderHeight = oDstSizeROI.height - nTopBorderHeight - oSrcSizeROI.height$.

nLeftBorderWidth Width (in pixels) of the left border. The width of the border at the right side of the destination ROI is implicitly defined by the size of the source ROI: $nRightBorderWidth = oDstSizeROI.width - nLeftBorderWidth - oSrcSizeROI.width$.

nValue The pixel value to be set for border pixels.

Returns:

[Image Data Related Error Codes](#), [ROI Related Error Codes](#)

7.10.1.19 `NppStatus nppiCopyConstBorder_8u_C3R (const Npp8u * pSrc, int nSrcStep, NppiSize oSrcSizeROI, Npp8u * pDst, int nDstStep, NppiSize oDstSizeROI, int nTopBorderHeight, int nLeftBorderWidth, const Npp8u aValue[3])`

3 channel 8-bit unsigned integer image copy with constant border color.

See [nppiCopyConstBorder_8u_C1R\(\)](#) for detailed documentation.

Parameters:

pSrc Source-Image Pointer.

nSrcStep Source-Image Line Step.

oSrcSizeROI Size of the source region-of-interest.

pDst Destination-Image Pointer.

nDstStep Destination-Image Line Step.

oDstSizeROI Size of the destination region-of-interest.

nTopBorderHeight Height of top border.

nLeftBorderWidth Width of left border.

aValue Vector of the RGBA values of the border pixels to be set.

Returns:

[Image Data Related Error Codes](#), [ROI Related Error Codes](#)

7.10.1.20 `NppStatus nppiCopyConstBorder_8u_C4R (const Npp8u * pSrc, int nSrcStep, NppiSize oSrcSizeROI, Npp8u * pDst, int nDstStep, NppiSize oDstSizeROI, int nTopBorderHeight, int nLeftBorderWidth, const Npp8u aValue[4])`

4 channel 8-bit unsigned integer image copy with constant border color.

See [nppiCopyConstBorder_8u_C1R\(\)](#) for detailed documentation.

Parameters:

pSrc Source-Image Pointer.

nSrcStep Source-Image Line Step.

oSrcSizeROI Size of the source region-of-interest.

pDst Destination-Image Pointer.

nDstStep Destination-Image Line Step.

oDstSizeROI Size of the destination region-of-interest.

nTopBorderHeight Height of top border.

nLeftBorderWidth Width of left border.

aValue Vector of the RGBA values of the border pixels to be set.

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.11 Copy Replicate Border

CopyReplicateBorder

Methods for copying images and padding borders with a replicates of the nearest source image pixel color.

- `NppStatus nppiCopyReplicateBorder_8u_C1R` (const `Npp8u` *pSrc, int nSrcStep, `NppiSize` oSrcSizeROI, `Npp8u` *pDst, int nDstStep, `NppiSize` oDstSizeROI, int nTopBorderHeight, int nLeftBorderWidth)

1 channel 8-bit unsigned integer image copy with nearest source image pixel color.
- `NppStatus nppiCopyReplicateBorder_8u_C3R` (const `Npp8u` *pSrc, int nSrcStep, `NppiSize` oSrcSizeROI, `Npp8u` *pDst, int nDstStep, `NppiSize` oDstSizeROI, int nTopBorderHeight, int nLeftBorderWidth)

3 channel 8-bit unsigned integer image copy with nearest source image pixel color.
- `NppStatus nppiCopyReplicateBorder_8u_C4R` (const `Npp8u` *pSrc, int nSrcStep, `NppiSize` oSrcSizeROI, `Npp8u` *pDst, int nDstStep, `NppiSize` oDstSizeROI, int nTopBorderHeight, int nLeftBorderWidth)

4 channel 8-bit unsigned integer image copy with nearest source image pixel color.
- `NppStatus nppiCopyReplicateBorder_8u_AC4R` (const `Npp8u` *pSrc, int nSrcStep, `NppiSize` oSrcSizeROI, `Npp8u` *pDst, int nDstStep, `NppiSize` oDstSizeROI, int nTopBorderHeight, int nLeftBorderWidth)

4 channel 8-bit unsigned integer image copy with nearest source image pixel color with alpha channel unaffected.
- `NppStatus nppiCopyReplicateBorder_16u_C1R` (const `Npp16u` *pSrc, int nSrcStep, `NppiSize` oSrcSizeROI, `Npp16u` *pDst, int nDstStep, `NppiSize` oDstSizeROI, int nTopBorderHeight, int nLeftBorderWidth)

1 channel 16-bit unsigned integer image copy with nearest source image pixel color.
- `NppStatus nppiCopyReplicateBorder_16u_C3R` (const `Npp16u` *pSrc, int nSrcStep, `NppiSize` oSrcSizeROI, `Npp16u` *pDst, int nDstStep, `NppiSize` oDstSizeROI, int nTopBorderHeight, int nLeftBorderWidth)

3 channel 16-bit unsigned integer image copy with nearest source image pixel color.
- `NppStatus nppiCopyReplicateBorder_16u_C4R` (const `Npp16u` *pSrc, int nSrcStep, `NppiSize` oSrcSizeROI, `Npp16u` *pDst, int nDstStep, `NppiSize` oDstSizeROI, int nTopBorderHeight, int nLeftBorderWidth)

4 channel 16-bit unsigned integer image copy with nearest source image pixel color.
- `NppStatus nppiCopyReplicateBorder_16u_AC4R` (const `Npp16u` *pSrc, int nSrcStep, `NppiSize` oSrcSizeROI, `Npp16u` *pDst, int nDstStep, `NppiSize` oDstSizeROI, int nTopBorderHeight, int nLeftBorderWidth)

4 channel 16-bit unsigned image copy with nearest source image pixel color with alpha channel unaffected.
- `NppStatus nppiCopyReplicateBorder_16s_C1R` (const `Npp16s` *pSrc, int nSrcStep, `NppiSize` oSrcSizeROI, `Npp16s` *pDst, int nDstStep, `NppiSize` oDstSizeROI, int nTopBorderHeight, int nLeftBorderWidth)

1 channel 16-bit signed integer image copy with nearest source image pixel color.

- **NppStatus nppiCopyReplicateBorder_16s_C3R** (const **Npp16s** *pSrc, int nSrcStep, **NppiSize** oSrcSizeROI, **Npp16s** *pDst, int nDstStep, **NppiSize** oDstSizeROI, int nTopBorderHeight, int nLeftBorderWidth)

3 channel 16-bit signed integer image copy with nearest source image pixel color.
- **NppStatus nppiCopyReplicateBorder_16s_C4R** (const **Npp16s** *pSrc, int nSrcStep, **NppiSize** oSrcSizeROI, **Npp16s** *pDst, int nDstStep, **NppiSize** oDstSizeROI, int nTopBorderHeight, int nLeftBorderWidth)

4 channel 16-bit signed integer image copy with nearest source image pixel color.
- **NppStatus nppiCopyReplicateBorder_16s_AC4R** (const **Npp16s** *pSrc, int nSrcStep, **NppiSize** oSrcSizeROI, **Npp16s** *pDst, int nDstStep, **NppiSize** oDstSizeROI, int nTopBorderHeight, int nLeftBorderWidth)

4 channel 16-bit signed integer image copy with nearest source image pixel color with alpha channel unaffected.
- **NppStatus nppiCopyReplicateBorder_32s_C1R** (const **Npp32s** *pSrc, int nSrcStep, **NppiSize** oSrcSizeROI, **Npp32s** *pDst, int nDstStep, **NppiSize** oDstSizeROI, int nTopBorderHeight, int nLeftBorderWidth)

1 channel 32-bit signed integer image copy with nearest source image pixel color.
- **NppStatus nppiCopyReplicateBorder_32s_C3R** (const **Npp32s** *pSrc, int nSrcStep, **NppiSize** oSrcSizeROI, **Npp32s** *pDst, int nDstStep, **NppiSize** oDstSizeROI, int nTopBorderHeight, int nLeftBorderWidth)

3 channel 32-bit signed image copy with nearest source image pixel color.
- **NppStatus nppiCopyReplicateBorder_32s_C4R** (const **Npp32s** *pSrc, int nSrcStep, **NppiSize** oSrcSizeROI, **Npp32s** *pDst, int nDstStep, **NppiSize** oDstSizeROI, int nTopBorderHeight, int nLeftBorderWidth)

4 channel 32-bit signed integer image copy with nearest source image pixel color.
- **NppStatus nppiCopyReplicateBorder_32s_AC4R** (const **Npp32s** *pSrc, int nSrcStep, **NppiSize** oSrcSizeROI, **Npp32s** *pDst, int nDstStep, **NppiSize** oDstSizeROI, int nTopBorderHeight, int nLeftBorderWidth)

4 channel 32-bit signed integer image copy with nearest source image pixel color with alpha channel unaffected.
- **NppStatus nppiCopyReplicateBorder_32f_C1R** (const **Npp32f** *pSrc, int nSrcStep, **NppiSize** oSrcSizeROI, **Npp32f** *pDst, int nDstStep, **NppiSize** oDstSizeROI, int nTopBorderHeight, int nLeftBorderWidth)

1 channel 32-bit floating point image copy with nearest source image pixel color.
- **NppStatus nppiCopyReplicateBorder_32f_C3R** (const **Npp32f** *pSrc, int nSrcStep, **NppiSize** oSrcSizeROI, **Npp32f** *pDst, int nDstStep, **NppiSize** oDstSizeROI, int nTopBorderHeight, int nLeftBorderWidth)

3 channel 32-bit floating point image copy with nearest source image pixel color.
- **NppStatus nppiCopyReplicateBorder_32f_C4R** (const **Npp32f** *pSrc, int nSrcStep, **NppiSize** oSrcSizeROI, **Npp32f** *pDst, int nDstStep, **NppiSize** oDstSizeROI, int nTopBorderHeight, int nLeftBorderWidth)

4 channel 32-bit floating point image copy with nearest source image pixel color.

- [NppStatus nppiCopyReplicateBorder_32f_AC4R](#) (const [Npp32f](#) *pSrc, int nSrcStep, [NppiSize](#) oSrcSizeROI, [Npp32f](#) *pDst, int nDstStep, [NppiSize](#) oDstSizeROI, int nTopBorderHeight, int nLeftBorderWidth)

4 channel 32-bit floating point image copy with nearest source image pixel color with alpha channel unaffected.

7.11.1 Function Documentation

7.11.1.1 [NppStatus nppiCopyReplicateBorder_16s_AC4R](#) (const [Npp16s](#) * pSrc, int nSrcStep, [NppiSize](#) oSrcSizeROI, [Npp16s](#) * pDst, int nDstStep, [NppiSize](#) oDstSizeROI, int nTopBorderHeight, int nLeftBorderWidth)

4 channel 16-bit signed integer image copy with nearest source image pixel color with alpha channel unaffected.

See [nppiCopyReplicateBorder_16s_C1R\(\)](#) for detailed documentation.

Parameters:

pSrc [Source-Image Pointer](#).
nSrcStep [Source-Image Line Step](#).
oSrcSizeROI Size of the source region-of-interest.
pDst [Destination-Image Pointer](#).
nDstStep [Destination-Image Line Step](#).
oDstSizeROI Size of the destination region-of-interest.
nTopBorderHeight Height of top border.
nLeftBorderWidth Width of left border.

Returns:

[Image Data Related Error Codes](#), [ROI Related Error Codes](#)

7.11.1.2 [NppStatus nppiCopyReplicateBorder_16s_C1R](#) (const [Npp16s](#) * pSrc, int nSrcStep, [NppiSize](#) oSrcSizeROI, [Npp16s](#) * pDst, int nDstStep, [NppiSize](#) oDstSizeROI, int nTopBorderHeight, int nLeftBorderWidth)

1 channel 16-bit signed integer image copy with nearest source image pixel color.

Parameters:

pSrc [Source-Image Pointer](#).
nSrcStep [Source-Image Line Step](#).
oSrcSizeROI Size of the source region of pixels.
pDst [Destination-Image Pointer](#).
nDstStep [Destination-Image Line Step](#).
oDstSizeROI Size (width, height) of the destination region, i.e. the region that gets filled with data from the source image (inner part) and nearest source image pixel color (outer part).

nTopBorderHeight Height (in pixels) of the top border. The number of pixel rows at the top of the destination ROI that will be filled with the nearest source image pixel color. $nBottomBorderHeight = oDstSizeROI.height - nTopBorderHeight - oSrcSizeROI.height$.

nLeftBorderWidth Width (in pixels) of the left border. The width of the border at the right side of the destination ROI is implicitly defined by the size of the source ROI: $nRightBorderWidth = oDstSizeROI.width - nLeftBorderWidth - oSrcSizeROI.width$.

Returns:

[Image Data Related Error Codes](#), [ROI Related Error Codes](#)

7.11.1.3 NppStatus nppiCopyReplicateBorder_16s_C3R (const Npp16s * pSrc, int nSrcStep, NppiSize oSrcSizeROI, Npp16s * pDst, int nDstStep, NppiSize oDstSizeROI, int nTopBorderHeight, int nLeftBorderWidth)

3 channel 16-bit signed integer image copy with nearest source image pixel color.

See [nppiCopyReplicateBorder_16s_C1R\(\)](#) for detailed documentation.

Parameters:

pSrc Source-Image Pointer.

nSrcStep Source-Image Line Step.

oSrcSizeROI Size of the source region-of-interest.

pDst Destination-Image Pointer.

nDstStep Destination-Image Line Step.

oDstSizeROI Size of the destination region-of-interest.

nTopBorderHeight Height of top border.

nLeftBorderWidth Width of left border.

Returns:

[Image Data Related Error Codes](#), [ROI Related Error Codes](#)

7.11.1.4 NppStatus nppiCopyReplicateBorder_16s_C4R (const Npp16s * pSrc, int nSrcStep, NppiSize oSrcSizeROI, Npp16s * pDst, int nDstStep, NppiSize oDstSizeROI, int nTopBorderHeight, int nLeftBorderWidth)

4 channel 16-bit signed integer image copy with nearest source image pixel color.

See [nppiCopyReplicateBorder_16s_C1R\(\)](#) for detailed documentation.

Parameters:

pSrc Source-Image Pointer.

nSrcStep Source-Image Line Step.

oSrcSizeROI Size of the source region-of-interest.

pDst Destination-Image Pointer.

nDstStep Destination-Image Line Step.

oDstSizeROI Size of the destination region-of-interest.

nTopBorderHeight Height of top border.

nLeftBorderWidth Width of left border.

Returns:

[Image Data Related Error Codes](#), [ROI Related Error Codes](#)

7.11.1.5 NppStatus nppiCopyReplicateBorder_16u_AC4R (const Npp16u * pSrc, int nSrcStep, NppiSize oSrcSizeROI, Npp16u * pDst, int nDstStep, NppiSize oDstSizeROI, int nTopBorderHeight, int nLeftBorderWidth)

4 channel 16-bit unsigned image copy with nearest source image pixel color with alpha channel unaffected.

See [nppiCopyReplicateBorder_16u_C1R\(\)](#) for detailed documentation.

Parameters:

pSrc [Source-Image Pointer](#).

nSrcStep [Source-Image Line Step](#).

oSrcSizeROI Size of the source region-of-interest.

pDst [Destination-Image Pointer](#).

nDstStep [Destination-Image Line Step](#).

oDstSizeROI Size of the destination region-of-interest.

nTopBorderHeight Height of top border.

nLeftBorderWidth Width of left border.

Returns:

[Image Data Related Error Codes](#), [ROI Related Error Codes](#)

7.11.1.6 NppStatus nppiCopyReplicateBorder_16u_C1R (const Npp16u * pSrc, int nSrcStep, NppiSize oSrcSizeROI, Npp16u * pDst, int nDstStep, NppiSize oDstSizeROI, int nTopBorderHeight, int nLeftBorderWidth)

1 channel 16-bit unsigned integer image copy with nearest source image pixel color.

Parameters:

pSrc [Source-Image Pointer](#).

nSrcStep [Source-Image Line Step](#).

oSrcSizeROI Size of the source region of pixels.

pDst [Destination-Image Pointer](#).

nDstStep [Destination-Image Line Step](#).

oDstSizeROI Size (width, height) of the destination region, i.e. the region that gets filled with data from the source image (inner part) and nearest source image pixel color (outer part).

nTopBorderHeight Height (in pixels) of the top border. The number of pixel rows at the top of the destination ROI that will be filled with the nearest source image pixel color. $nBottomBorderHeight = oDstSizeROI.height - nTopBorderHeight - oSrcSizeROI.height$.

nLeftBorderWidth Width (in pixels) of the left border. The width of the border at the right side of the destination ROI is implicitly defined by the size of the source ROI: $nRightBorderWidth = oDstSizeROI.width - nLeftBorderWidth - oSrcSizeROI.width$.

Returns:

[Image Data Related Error Codes](#), [ROI Related Error Codes](#)

7.11.1.7 NppStatus nppiCopyReplicateBorder_16u_C3R (const Npp16u * pSrc, int nSrcStep, NppiSize oSrcSizeROI, Npp16u * pDst, int nDstStep, NppiSize oDstSizeROI, int nTopBorderHeight, int nLeftBorderWidth)

3 channel 16-bit unsigned integer image copy with nearest source image pixel color.

See [nppiCopyReplicateBorder_16u_C1R\(\)](#) for detailed documentation.

Parameters:

pSrc [Source-Image Pointer](#).

nSrcStep [Source-Image Line Step](#).

oSrcSizeROI Size of the source region-of-interest.

pDst [Destination-Image Pointer](#).

nDstStep [Destination-Image Line Step](#).

oDstSizeROI Size of the destination region-of-interest.

nTopBorderHeight Height of top border.

nLeftBorderWidth Width of left border.

Returns:

[Image Data Related Error Codes](#), [ROI Related Error Codes](#)

7.11.1.8 NppStatus nppiCopyReplicateBorder_16u_C4R (const Npp16u * pSrc, int nSrcStep, NppiSize oSrcSizeROI, Npp16u * pDst, int nDstStep, NppiSize oDstSizeROI, int nTopBorderHeight, int nLeftBorderWidth)

4 channel 16-bit unsigned integer image copy with nearest source image pixel color.

See [nppiCopyReplicateBorder_16u_C1R\(\)](#) for detailed documentation.

Parameters:

pSrc [Source-Image Pointer](#).

nSrcStep [Source-Image Line Step](#).

oSrcSizeROI Size of the source region-of-interest.

pDst [Destination-Image Pointer](#).

nDstStep [Destination-Image Line Step](#).

oDstSizeROI Size of the destination region-of-interest.

nTopBorderHeight Height of top border.

nLeftBorderWidth Width of left border.

Returns:

[Image Data Related Error Codes](#), [ROI Related Error Codes](#)

7.11.1.9 NppStatus nppiCopyReplicateBorder_32f_AC4R (const Npp32f * pSrc, int nSrcStep, NppiSize oSrcSizeROI, Npp32f * pDst, int nDstStep, NppiSize oDstSizeROI, int nTopBorderHeight, int nLeftBorderWidth)

4 channel 32-bit floating point image copy with nearest source image pixel color with alpha channel unaffected.

See [nppiCopyReplicateBorder_32f_C1R\(\)](#) for detailed documentation.

Parameters:

- pSrc* Source-Image Pointer.
- nSrcStep* Source-Image Line Step.
- oSrcSizeROI* Size of the source region-of-interest.
- pDst* Destination-Image Pointer.
- nDstStep* Destination-Image Line Step.
- oDstSizeROI* Size of the destination region-of-interest.
- nTopBorderHeight* Height of top border.
- nLeftBorderWidth* Width of left border.

Returns:

[Image Data Related Error Codes](#), [ROI Related Error Codes](#)

7.11.1.10 NppStatus nppiCopyReplicateBorder_32f_C1R (const Npp32f * pSrc, int nSrcStep, NppiSize oSrcSizeROI, Npp32f * pDst, int nDstStep, NppiSize oDstSizeROI, int nTopBorderHeight, int nLeftBorderWidth)

1 channel 32-bit floating point image copy with nearest source image pixel color.

Parameters:

- pSrc* Source-Image Pointer.
- nSrcStep* Source-Image Line Step.
- oSrcSizeROI* Size of the source region of pixels.
- pDst* Destination-Image Pointer.
- nDstStep* Destination-Image Line Step.
- oDstSizeROI* Size (width, height) of the destination region, i.e. the region that gets filled with data from the source image (inner part) and nearest source image pixel color (outer part).
- nTopBorderHeight* Height (in pixels) of the top border. The number of pixel rows at the top of the destination ROI that will be filled with the nearest source image pixel color. $nBottomBorderHeight = oDstSizeROI.height - nTopBorderHeight - oSrcSizeROI.height$.
- nLeftBorderWidth* Width (in pixels) of the left border. The width of the border at the right side of the destination ROI is implicitly defined by the size of the source ROI: $nRightBorderWidth = oDstSizeROI.width - nLeftBorderWidth - oSrcSizeROI.width$.

Returns:

[Image Data Related Error Codes](#), [ROI Related Error Codes](#)

7.11.1.11 NppStatus nppiCopyReplicateBorder_32f_C3R (const Npp32f * pSrc, int nSrcStep, NppiSize oSrcSizeROI, Npp32f * pDst, int nDstStep, NppiSize oDstSizeROI, int nTopBorderHeight, int nLeftBorderWidth)

3 channel 32-bit floating point image copy with nearest source image pixel color.

See [nppiCopyReplicateBorder_32f_C1R\(\)](#) for detailed documentation.

Parameters:

pSrc Source-Image Pointer.

nSrcStep Source-Image Line Step.

oSrcSizeROI Size of the source region-of-interest.

pDst Destination-Image Pointer.

nDstStep Destination-Image Line Step.

oDstSizeROI Size of the destination region-of-interest.

nTopBorderHeight Height of top border.

nLeftBorderWidth Width of left border.

Returns:

[Image Data Related Error Codes](#), [ROI Related Error Codes](#)

7.11.1.12 NppStatus nppiCopyReplicateBorder_32f_C4R (const Npp32f * pSrc, int nSrcStep, NppiSize oSrcSizeROI, Npp32f * pDst, int nDstStep, NppiSize oDstSizeROI, int nTopBorderHeight, int nLeftBorderWidth)

4 channel 32-bit floating point image copy with nearest source image pixel color.

See [nppiCopyReplicateBorder_32f_C1R\(\)](#) for detailed documentation.

Parameters:

pSrc Source-Image Pointer.

nSrcStep Source-Image Line Step.

oSrcSizeROI Size of the source region-of-interest.

pDst Destination-Image Pointer.

nDstStep Destination-Image Line Step.

oDstSizeROI Size of the destination region-of-interest.

nTopBorderHeight Height of top border.

nLeftBorderWidth Width of left border.

Returns:

[Image Data Related Error Codes](#), [ROI Related Error Codes](#)

7.11.1.13 NppStatus nppiCopyReplicateBorder_32s_AC4R (const Npp32s * pSrc, int nSrcStep, NppiSize oSrcSizeROI, Npp32s * pDst, int nDstStep, NppiSize oDstSizeROI, int nTopBorderHeight, int nLeftBorderWidth)

4 channel 32-bit signed integer image copy with nearest source image pixel color with alpha channel unaffected.

See [nppiCopyReplicateBorder_32s_C1R\(\)](#) for detailed documentation.

Parameters:

- pSrc* Source-Image Pointer.
- nSrcStep* Source-Image Line Step.
- oSrcSizeROI* Size of the source region-of-interest.
- pDst* Destination-Image Pointer.
- nDstStep* Destination-Image Line Step.
- oDstSizeROI* Size of the destination region-of-interest.
- nTopBorderHeight* Height of top border.
- nLeftBorderWidth* Width of left border.

Returns:

[Image Data Related Error Codes](#), [ROI Related Error Codes](#)

7.11.1.14 NppStatus nppiCopyReplicateBorder_32s_C1R (const Npp32s * pSrc, int nSrcStep, NppiSize oSrcSizeROI, Npp32s * pDst, int nDstStep, NppiSize oDstSizeROI, int nTopBorderHeight, int nLeftBorderWidth)

1 channel 32-bit signed integer image copy with nearest source image pixel color.

Parameters:

- pSrc* Source-Image Pointer.
- nSrcStep* Source-Image Line Step.
- oSrcSizeROI* Size of the source region of pixels.
- pDst* Destination-Image Pointer.
- nDstStep* Destination-Image Line Step.
- oDstSizeROI* Size (width, height) of the destination region, i.e. the region that gets filled with data from the source image (inner part) and nearest source image pixel color (outer part).
- nTopBorderHeight* Height (in pixels) of the top border. The number of pixel rows at the top of the destination ROI that will be filled with the nearest source image pixel color. $nBottomBorderHeight = oDstSizeROI.height - nTopBorderHeight - oSrcSizeROI.height$.
- nLeftBorderWidth* Width (in pixels) of the left border. The width of the border at the right side of the destination ROI is implicitly defined by the size of the source ROI: $nRightBorderWidth = oDstSizeROI.width - nLeftBorderWidth - oSrcSizeROI.width$.

Returns:

[Image Data Related Error Codes](#), [ROI Related Error Codes](#)

7.11.1.15 NppStatus nppiCopyReplicateBorder_32s_C3R (const Npp32s * pSrc, int nSrcStep, NppiSize oSrcSizeROI, Npp32s * pDst, int nDstStep, NppiSize oDstSizeROI, int nTopBorderHeight, int nLeftBorderWidth)

3 channel 32-bit signed image copy with nearest source image pixel color.

See [nppiCopyReplicateBorder_32s_C1R\(\)](#) for detailed documentation.

Parameters:

pSrc Source-Image Pointer.

nSrcStep Source-Image Line Step.

oSrcSizeROI Size of the source region-of-interest.

pDst Destination-Image Pointer.

nDstStep Destination-Image Line Step.

oDstSizeROI Size of the destination region-of-interest.

nTopBorderHeight Height of top border.

nLeftBorderWidth Width of left border.

Returns:

[Image Data Related Error Codes](#), [ROI Related Error Codes](#)

7.11.1.16 NppStatus nppiCopyReplicateBorder_32s_C4R (const Npp32s * pSrc, int nSrcStep, NppiSize oSrcSizeROI, Npp32s * pDst, int nDstStep, NppiSize oDstSizeROI, int nTopBorderHeight, int nLeftBorderWidth)

4 channel 32-bit signed integer image copy with nearest source image pixel color.

See [nppiCopyReplicateBorder_32s_C1R\(\)](#) for detailed documentation.

Parameters:

pSrc Source-Image Pointer.

nSrcStep Source-Image Line Step.

oSrcSizeROI Size of the source region-of-interest.

pDst Destination-Image Pointer.

nDstStep Destination-Image Line Step.

oDstSizeROI Size of the destination region-of-interest.

nTopBorderHeight Height of top border.

nLeftBorderWidth Width of left border.

Returns:

[Image Data Related Error Codes](#), [ROI Related Error Codes](#)

7.11.1.17 NppStatus nppiCopyReplicateBorder_8u_AC4R (const Npp8u * pSrc, int nSrcStep, NppiSize oSrcSizeROI, Npp8u * pDst, int nDstStep, NppiSize oDstSizeROI, int nTopBorderHeight, int nLeftBorderWidth)

4 channel 8-bit unsigned integer image copy with nearest source image pixel color with alpha channel unaffected.

See [nppiCopyReplicateBorder_8u_C1R\(\)](#) for detailed documentation.

Parameters:

- pSrc* Source-Image Pointer.
- nSrcStep* Source-Image Line Step.
- oSrcSizeROI* Size of the source region-of-interest.
- pDst* Destination-Image Pointer.
- nDstStep* Destination-Image Line Step.
- oDstSizeROI* Size of the destination region-of-interest.
- nTopBorderHeight* Height of top border.
- nLeftBorderWidth* Width of left border.

Returns:

[Image Data Related Error Codes](#), [ROI Related Error Codes](#)

7.11.1.18 NppStatus nppiCopyReplicateBorder_8u_C1R (const Npp8u * pSrc, int nSrcStep, NppiSize oSrcSizeROI, Npp8u * pDst, int nDstStep, NppiSize oDstSizeROI, int nTopBorderHeight, int nLeftBorderWidth)

1 channel 8-bit unsigned integer image copy with nearest source image pixel color.

Parameters:

- pSrc* Source-Image Pointer.
- nSrcStep* Source-Image Line Step.
- oSrcSizeROI* Size of the source region of pixels.
- pDst* Destination-Image Pointer.
- nDstStep* Destination-Image Line Step.
- oDstSizeROI* Size (width, height) of the destination region, i.e. the region that gets filled with data from the source image (inner part) and nearest source image pixel color (outer part).
- nTopBorderHeight* Height (in pixels) of the top border. The number of pixel rows at the top of the destination ROI that will be filled with the nearest source image pixel color. $nBottomBorderHeight = oDstSizeROI.height - nTopBorderHeight - oSrcSizeROI.height$.
- nLeftBorderWidth* Width (in pixels) of the left border. The width of the border at the right side of the destination ROI is implicitly defined by the size of the source ROI: $nRightBorderWidth = oDstSizeROI.width - nLeftBorderWidth - oSrcSizeROI.width$.

Returns:

[Image Data Related Error Codes](#), [ROI Related Error Codes](#)

7.11.1.19 `NppStatus nppiCopyReplicateBorder_8u_C3R (const Npp8u * pSrc, int nSrcStep, NppiSize oSrcSizeROI, Npp8u * pDst, int nDstStep, NppiSize oDstSizeROI, int nTopBorderHeight, int nLeftBorderWidth)`

3 channel 8-bit unsigned integer image copy with nearest source image pixel color.

See [nppiCopyReplicateBorder_8u_C1R\(\)](#) for detailed documentation.

Parameters:

- pSrc* Source-Image Pointer.
- nSrcStep* Source-Image Line Step.
- oSrcSizeROI* Size of the source region-of-interest.
- pDst* Destination-Image Pointer.
- nDstStep* Destination-Image Line Step.
- oDstSizeROI* Size of the destination region-of-interest.
- nTopBorderHeight* Height of top border.
- nLeftBorderWidth* Width of left border.

Returns:

[Image Data Related Error Codes](#), [ROI Related Error Codes](#)

7.11.1.20 `NppStatus nppiCopyReplicateBorder_8u_C4R (const Npp8u * pSrc, int nSrcStep, NppiSize oSrcSizeROI, Npp8u * pDst, int nDstStep, NppiSize oDstSizeROI, int nTopBorderHeight, int nLeftBorderWidth)`

4 channel 8-bit unsigned integer image copy with nearest source image pixel color.

See [nppiCopyReplicateBorder_8u_C1R\(\)](#) for detailed documentation.

Parameters:

- pSrc* Source-Image Pointer.
- nSrcStep* Source-Image Line Step.
- oSrcSizeROI* Size of the source region-of-interest.
- pDst* Destination-Image Pointer.
- nDstStep* Destination-Image Line Step.
- oDstSizeROI* Size of the destination region-of-interest.
- nTopBorderHeight* Height of top border.
- nLeftBorderWidth* Width of left border.

Returns:

[Image Data Related Error Codes](#), [ROI Related Error Codes](#)

7.12 Copy Wrap Border

CopyWrapBorder

Methods for copying images and padding borders with wrapped replications of the source image pixel colors.

- **NppStatus nppiCopyWrapBorder_8u_C1R** (const **Npp8u** *pSrc, int nSrcStep, **NppiSize** oSrcSizeROI, **Npp8u** *pDst, int nDstStep, **NppiSize** oDstSizeROI, int nTopBorderHeight, int nLeftBorderWidth)

1 channel 8-bit unsigned integer image copy with the borders wrapped by replication of source image pixel colors.
- **NppStatus nppiCopyWrapBorder_8u_C3R** (const **Npp8u** *pSrc, int nSrcStep, **NppiSize** oSrcSizeROI, **Npp8u** *pDst, int nDstStep, **NppiSize** oDstSizeROI, int nTopBorderHeight, int nLeftBorderWidth)

3 channel 8-bit unsigned integer image copy with the borders wrapped by replication of source image pixel colors.
- **NppStatus nppiCopyWrapBorder_8u_C4R** (const **Npp8u** *pSrc, int nSrcStep, **NppiSize** oSrcSizeROI, **Npp8u** *pDst, int nDstStep, **NppiSize** oDstSizeROI, int nTopBorderHeight, int nLeftBorderWidth)

4 channel 8-bit unsigned integer image copy with the borders wrapped by replication of source image pixel colors.
- **NppStatus nppiCopyWrapBorder_8u_AC4R** (const **Npp8u** *pSrc, int nSrcStep, **NppiSize** oSrcSizeROI, **Npp8u** *pDst, int nDstStep, **NppiSize** oDstSizeROI, int nTopBorderHeight, int nLeftBorderWidth)

4 channel 8-bit unsigned integer image copy with the borders wrapped by replication of source image pixel colors with alpha channel unaffected.
- **NppStatus nppiCopyWrapBorder_16u_C1R** (const **Npp16u** *pSrc, int nSrcStep, **NppiSize** oSrcSizeROI, **Npp16u** *pDst, int nDstStep, **NppiSize** oDstSizeROI, int nTopBorderHeight, int nLeftBorderWidth)

1 channel 16-bit unsigned integer image copy with the borders wrapped by replication of source image pixel colors.
- **NppStatus nppiCopyWrapBorder_16u_C3R** (const **Npp16u** *pSrc, int nSrcStep, **NppiSize** oSrcSizeROI, **Npp16u** *pDst, int nDstStep, **NppiSize** oDstSizeROI, int nTopBorderHeight, int nLeftBorderWidth)

3 channel 16-bit unsigned integer image copy with the borders wrapped by replication of source image pixel colors.
- **NppStatus nppiCopyWrapBorder_16u_C4R** (const **Npp16u** *pSrc, int nSrcStep, **NppiSize** oSrcSizeROI, **Npp16u** *pDst, int nDstStep, **NppiSize** oDstSizeROI, int nTopBorderHeight, int nLeftBorderWidth)

4 channel 16-bit unsigned integer image copy with the borders wrapped by replication of source image pixel colors.
- **NppStatus nppiCopyWrapBorder_16u_AC4R** (const **Npp16u** *pSrc, int nSrcStep, **NppiSize** oSrcSizeROI, **Npp16u** *pDst, int nDstStep, **NppiSize** oDstSizeROI, int nTopBorderHeight, int nLeftBorderWidth)

4 channel 16-bit unsigned integer image copy with the borders wrapped by replication of source image pixel colors with alpha channel unaffected.

- **NppStatus nppiCopyWrapBorder_16s_C1R** (const **Npp16s** *pSrc, int nSrcStep, **NppiSize** oSrcSizeROI, **Npp16s** *pDst, int nDstStep, **NppiSize** oDstSizeROI, int nTopBorderHeight, int nLeftBorderWidth)

1 channel 16-bit signed integer image copy with the borders wrapped by replication of source image pixel colors.

- **NppStatus nppiCopyWrapBorder_16s_C3R** (const **Npp16s** *pSrc, int nSrcStep, **NppiSize** oSrcSizeROI, **Npp16s** *pDst, int nDstStep, **NppiSize** oDstSizeROI, int nTopBorderHeight, int nLeftBorderWidth)

3 channel 16-bit signed integer image copy with the borders wrapped by replication of source image pixel colors.

- **NppStatus nppiCopyWrapBorder_16s_C4R** (const **Npp16s** *pSrc, int nSrcStep, **NppiSize** oSrcSizeROI, **Npp16s** *pDst, int nDstStep, **NppiSize** oDstSizeROI, int nTopBorderHeight, int nLeftBorderWidth)

4 channel 16-bit signed integer image copy with the borders wrapped by replication of source image pixel colors.

- **NppStatus nppiCopyWrapBorder_16s_AC4R** (const **Npp16s** *pSrc, int nSrcStep, **NppiSize** oSrcSizeROI, **Npp16s** *pDst, int nDstStep, **NppiSize** oDstSizeROI, int nTopBorderHeight, int nLeftBorderWidth)

4 channel 16-bit signed integer image copy with the borders wrapped by replication of source image pixel colors with alpha channel unaffected.

- **NppStatus nppiCopyWrapBorder_32s_C1R** (const **Npp32s** *pSrc, int nSrcStep, **NppiSize** oSrcSizeROI, **Npp32s** *pDst, int nDstStep, **NppiSize** oDstSizeROI, int nTopBorderHeight, int nLeftBorderWidth)

1 channel 32-bit signed integer image copy with the borders wrapped by replication of source image pixel colors.

- **NppStatus nppiCopyWrapBorder_32s_C3R** (const **Npp32s** *pSrc, int nSrcStep, **NppiSize** oSrcSizeROI, **Npp32s** *pDst, int nDstStep, **NppiSize** oDstSizeROI, int nTopBorderHeight, int nLeftBorderWidth)

3 channel 32-bit signed integer image copy with the borders wrapped by replication of source image pixel colors.

- **NppStatus nppiCopyWrapBorder_32s_C4R** (const **Npp32s** *pSrc, int nSrcStep, **NppiSize** oSrcSizeROI, **Npp32s** *pDst, int nDstStep, **NppiSize** oDstSizeROI, int nTopBorderHeight, int nLeftBorderWidth)

4 channel 32-bit signed integer image copy with the borders wrapped by replication of source image pixel colors.

- **NppStatus nppiCopyWrapBorder_32s_AC4R** (const **Npp32s** *pSrc, int nSrcStep, **NppiSize** oSrcSizeROI, **Npp32s** *pDst, int nDstStep, **NppiSize** oDstSizeROI, int nTopBorderHeight, int nLeftBorderWidth)

4 channel 32-bit signed integer image copy with the borders wrapped by replication of source image pixel colors with alpha channel unaffected.

- [NppStatus nppiCopyWrapBorder_32f_C1R](#) (const [Npp32f](#) *pSrc, int nSrcStep, [NppiSize](#) oSrcSizeROI, [Npp32f](#) *pDst, int nDstStep, [NppiSize](#) oDstSizeROI, int nTopBorderHeight, int nLeftBorderWidth)
1 channel 32-bit floating point image copy with the borders wrapped by replication of source image pixel colors.
- [NppStatus nppiCopyWrapBorder_32f_C3R](#) (const [Npp32f](#) *pSrc, int nSrcStep, [NppiSize](#) oSrcSizeROI, [Npp32f](#) *pDst, int nDstStep, [NppiSize](#) oDstSizeROI, int nTopBorderHeight, int nLeftBorderWidth)
3 channel 32-bit floating point image copy with the borders wrapped by replication of source image pixel colors.
- [NppStatus nppiCopyWrapBorder_32f_C4R](#) (const [Npp32f](#) *pSrc, int nSrcStep, [NppiSize](#) oSrcSizeROI, [Npp32f](#) *pDst, int nDstStep, [NppiSize](#) oDstSizeROI, int nTopBorderHeight, int nLeftBorderWidth)
4 channel 32-bit floating point image copy with the borders wrapped by replication of source image pixel colors.
- [NppStatus nppiCopyWrapBorder_32f_AC4R](#) (const [Npp32f](#) *pSrc, int nSrcStep, [NppiSize](#) oSrcSizeROI, [Npp32f](#) *pDst, int nDstStep, [NppiSize](#) oDstSizeROI, int nTopBorderHeight, int nLeftBorderWidth)
1 channel 32-bit floating point image copy with the borders wrapped by replication of source image pixel colors with alpha channel unaffected.

7.12.1 Function Documentation

7.12.1.1 [NppStatus nppiCopyWrapBorder_16s_AC4R](#) (const [Npp16s](#) *pSrc, int nSrcStep, [NppiSize](#) oSrcSizeROI, [Npp16s](#) *pDst, int nDstStep, [NppiSize](#) oDstSizeROI, int nTopBorderHeight, int nLeftBorderWidth)

4 channel 16-bit signed integer image copy with the borders wrapped by replication of source image pixel colors with alpha channel unaffected.

See [nppiCopyWrapBorder_16s_C1R\(\)](#) for detailed documentation.

Parameters:

- pSrc* [Source-Image Pointer](#).
- nSrcStep* [Source-Image Line Step](#).
- oSrcSizeROI* Size of the source region-of-interest.
- pDst* [Destination-Image Pointer](#).
- nDstStep* [Destination-Image Line Step](#).
- oDstSizeROI* Size of the destination region-of-interest.
- nTopBorderHeight* Height of top border.
- nLeftBorderWidth* Width of left border.

Returns:

[Image Data Related Error Codes](#), [ROI Related Error Codes](#)

7.12.1.2 NppStatus nppiCopyWrapBorder_16s_C1R (const Npp16s * pSrc, int nSrcStep, NppiSize oSrcSizeROI, Npp16s * pDst, int nDstStep, NppiSize oDstSizeROI, int nTopBorderHeight, int nLeftBorderWidth)

1 channel 16-bit signed integer image copy with the borders wrapped by replication of source image pixel colors.

Parameters:

pSrc Source-Image Pointer.

nSrcStep Source-Image Line Step.

oSrcSizeROI Size of the source region of pixels.

pDst Destination-Image Pointer.

nDstStep Destination-Image Line Step.

oDstSizeROI Size (width, height) of the destination region, i.e. the region that gets filled with data from the source image (inner part) and a border consisting of wrapped replication of the source image pixel colors (outer part).

nTopBorderHeight Height (in pixels) of the top border. The number of pixel rows at the top of the destination ROI that will be filled with the wrapped replication of the corresponding column of source image pixels colors. $nBottomBorderHeight = oDstSizeROI.height - nTopBorderHeight - oSrcSizeROI.height$.

nLeftBorderWidth Width (in pixels) of the left border. The width of the border at the right side of the destination ROI is implicitly defined by the size of the source ROI: $nRightBorderWidth = oDstSizeROI.width - nLeftBorderWidth - oSrcSizeROI.width$.

Returns:

[Image Data Related Error Codes](#), [ROI Related Error Codes](#)

7.12.1.3 NppStatus nppiCopyWrapBorder_16s_C3R (const Npp16s * pSrc, int nSrcStep, NppiSize oSrcSizeROI, Npp16s * pDst, int nDstStep, NppiSize oDstSizeROI, int nTopBorderHeight, int nLeftBorderWidth)

3 channel 16-bit signed integer image copy with the borders wrapped by replication of source image pixel colors.

See [nppiCopyWrapBorder_16s_C1R\(\)](#) for detailed documentation.

Parameters:

pSrc Source-Image Pointer.

nSrcStep Source-Image Line Step.

oSrcSizeROI Size of the source region-of-interest.

pDst Destination-Image Pointer.

nDstStep Destination-Image Line Step.

oDstSizeROI Size of the destination region-of-interest.

nTopBorderHeight Height of top border.

nLeftBorderWidth Width of left border.

Returns:

[Image Data Related Error Codes](#), [ROI Related Error Codes](#)

7.12.1.4 NppStatus nppiCopyWrapBorder_16s_C4R (const Npp16s * pSrc, int nSrcStep, NppiSize oSrcSizeROI, Npp16s * pDst, int nDstStep, NppiSize oDstSizeROI, int nTopBorderHeight, int nLeftBorderWidth)

4 channel 16-bit signed integer image copy with the borders wrapped by replication of source image pixel colors.

See [nppiCopyWrapBorder_16s_C1R\(\)](#) for detailed documentation.

Parameters:

- pSrc* Source-Image Pointer.
- nSrcStep* Source-Image Line Step.
- oSrcSizeROI* Size of the source region-of-interest.
- pDst* Destination-Image Pointer.
- nDstStep* Destination-Image Line Step.
- oDstSizeROI* Size of the destination region-of-interest.
- nTopBorderHeight* Height of top border.
- nLeftBorderWidth* Width of left border.

Returns:

[Image Data Related Error Codes](#), [ROI Related Error Codes](#)

7.12.1.5 NppStatus nppiCopyWrapBorder_16u_AC4R (const Npp16u * pSrc, int nSrcStep, NppiSize oSrcSizeROI, Npp16u * pDst, int nDstStep, NppiSize oDstSizeROI, int nTopBorderHeight, int nLeftBorderWidth)

4 channel 16-bit unsigned integer image copy with the borders wrapped by replication of source image pixel colors with alpha channel unaffected.

See [nppiCopyWrapBorder_16u_C1R\(\)](#) for detailed documentation.

Parameters:

- pSrc* Source-Image Pointer.
- nSrcStep* Source-Image Line Step.
- oSrcSizeROI* Size of the source region-of-interest.
- pDst* Destination-Image Pointer.
- nDstStep* Destination-Image Line Step.
- oDstSizeROI* Size of the destination region-of-interest.
- nTopBorderHeight* Height of top border.
- nLeftBorderWidth* Width of left border.

Returns:

[Image Data Related Error Codes](#), [ROI Related Error Codes](#)

7.12.1.6 NppStatus nppiCopyWrapBorder_16u_C1R (const Npp16u * pSrc, int nSrcStep, NppiSize oSrcSizeROI, Npp16u * pDst, int nDstStep, NppiSize oDstSizeROI, int nTopBorderHeight, int nLeftBorderWidth)

1 channel 16-bit unsigned integer image copy with the borders wrapped by replication of source image pixel colors.

Parameters:

pSrc Source-Image Pointer.

nSrcStep Source-Image Line Step.

oSrcSizeROI Size of the source region of pixels.

pDst Destination-Image Pointer.

nDstStep Destination-Image Line Step.

oDstSizeROI Size (width, height) of the destination region, i.e. the region that gets filled with data from the source image (inner part) and a border consisting of wrapped replication of the source image pixel colors (outer part).

nTopBorderHeight Height (in pixels) of the top border. The number of pixel rows at the top of the destination ROI that will be filled with the wrapped replication of the corresponding column of source image pixels colors. $nBottomBorderHeight = oDstSizeROI.height - nTopBorderHeight - oSrcSizeROI.height$.

nLeftBorderWidth Width (in pixels) of the left border. The width of the border at the right side of the destination ROI is implicitly defined by the size of the source ROI: $nRightBorderWidth = oDstSizeROI.width - nLeftBorderWidth - oSrcSizeROI.width$.

Returns:

[Image Data Related Error Codes](#), [ROI Related Error Codes](#)

7.12.1.7 NppStatus nppiCopyWrapBorder_16u_C3R (const Npp16u * pSrc, int nSrcStep, NppiSize oSrcSizeROI, Npp16u * pDst, int nDstStep, NppiSize oDstSizeROI, int nTopBorderHeight, int nLeftBorderWidth)

3 channel 16-bit unsigned integer image copy with the borders wrapped by replication of source image pixel colors.

See [nppiCopyWrapBorder_16u_C1R\(\)](#) for detailed documentation.

Parameters:

pSrc Source-Image Pointer.

nSrcStep Source-Image Line Step.

oSrcSizeROI Size of the source region-of-interest.

pDst Destination-Image Pointer.

nDstStep Destination-Image Line Step.

oDstSizeROI Size of the destination region-of-interest.

nTopBorderHeight Height of top border.

nLeftBorderWidth Width of left border.

Returns:

[Image Data Related Error Codes](#), [ROI Related Error Codes](#)

7.12.1.8 NppStatus nppiCopyWrapBorder_16u_C4R (const Npp16u * pSrc, int nSrcStep, NppiSize oSrcSizeROI, Npp16u * pDst, int nDstStep, NppiSize oDstSizeROI, int nTopBorderHeight, int nLeftBorderWidth)

4 channel 16-bit unsigned integer image copy with the borders wrapped by replication of source image pixel colors.

See [nppiCopyWrapBorder_16u_C1R\(\)](#) for detailed documentation.

Parameters:

- pSrc* Source-Image Pointer.
- nSrcStep* Source-Image Line Step.
- oSrcSizeROI* Size of the source region-of-interest.
- pDst* Destination-Image Pointer.
- nDstStep* Destination-Image Line Step.
- oDstSizeROI* Size of the destination region-of-interest.
- nTopBorderHeight* Height of top border.
- nLeftBorderWidth* Width of left border.

Returns:

[Image Data Related Error Codes](#), [ROI Related Error Codes](#)

7.12.1.9 NppStatus nppiCopyWrapBorder_32f_AC4R (const Npp32f * pSrc, int nSrcStep, NppiSize oSrcSizeROI, Npp32f * pDst, int nDstStep, NppiSize oDstSizeROI, int nTopBorderHeight, int nLeftBorderWidth)

1 channel 32-bit floating point image copy with the borders wrapped by replication of source image pixel colors with alpha channel unaffected.

See [nppiCopyWrapBorder_32f_C1R\(\)](#) for detailed documentation.

Parameters:

- pSrc* Source-Image Pointer.
- nSrcStep* Source-Image Line Step.
- oSrcSizeROI* Size of the source region-of-interest.
- pDst* Destination-Image Pointer.
- nDstStep* Destination-Image Line Step.
- oDstSizeROI* Size of the destination region-of-interest.
- nTopBorderHeight* Height of top border.
- nLeftBorderWidth* Width of left border.

Returns:

[Image Data Related Error Codes](#), [ROI Related Error Codes](#)

7.12.1.10 NppStatus nppiCopyWrapBorder_32f_C1R (const Npp32f * pSrc, int nSrcStep, NppiSize oSrcSizeROI, Npp32f * pDst, int nDstStep, NppiSize oDstSizeROI, int nTopBorderHeight, int nLeftBorderWidth)

1 channel 32-bit floating point image copy with the borders wrapped by replication of source image pixel colors.

Parameters:

pSrc Source-Image Pointer.

nSrcStep Source-Image Line Step.

oSrcSizeROI Size of the source region of pixels.

pDst Destination-Image Pointer.

nDstStep Destination-Image Line Step.

oDstSizeROI Size (width, height) of the destination region, i.e. the region that gets filled with data from the source image (inner part) and a border consisting of wrapped replication of the source image pixel colors (outer part).

nTopBorderHeight Height (in pixels) of the top border. The number of pixel rows at the top of the destination ROI that will be filled with the wrapped replication of the corresponding column of source image pixels colors. $nBottomBorderHeight = oDstSizeROI.height - nTopBorderHeight - oSrcSizeROI.height$.

nLeftBorderWidth Width (in pixels) of the left border. The width of the border at the right side of the destination ROI is implicitly defined by the size of the source ROI: $nRightBorderWidth = oDstSizeROI.width - nLeftBorderWidth - oSrcSizeROI.width$.

Returns:

[Image Data Related Error Codes](#), [ROI Related Error Codes](#)

7.12.1.11 NppStatus nppiCopyWrapBorder_32f_C3R (const Npp32f * pSrc, int nSrcStep, NppiSize oSrcSizeROI, Npp32f * pDst, int nDstStep, NppiSize oDstSizeROI, int nTopBorderHeight, int nLeftBorderWidth)

3 channel 32-bit floating point image copy with the borders wrapped by replication of source image pixel colors.

See [nppiCopyWrapBorder_32f_C1R\(\)](#) for detailed documentation.

Parameters:

pSrc Source-Image Pointer.

nSrcStep Source-Image Line Step.

oSrcSizeROI Size of the source region-of-interest.

pDst Destination-Image Pointer.

nDstStep Destination-Image Line Step.

oDstSizeROI Size of the destination region-of-interest.

nTopBorderHeight Height of top border.

nLeftBorderWidth Width of left border.

Returns:

[Image Data Related Error Codes](#), [ROI Related Error Codes](#)

7.12.1.12 NppStatus nppiCopyWrapBorder_32f_C4R (const Npp32f * pSrc, int nSrcStep, NppiSize oSrcSizeROI, Npp32f * pDst, int nDstStep, NppiSize oDstSizeROI, int nTopBorderHeight, int nLeftBorderWidth)

4 channel 32-bit floating point image copy with the borders wrapped by replication of source image pixel colors.

See [nppiCopyWrapBorder_32f_C1R\(\)](#) for detailed documentation.

Parameters:

- pSrc* Source-Image Pointer.
- nSrcStep* Source-Image Line Step.
- oSrcSizeROI* Size of the source region-of-interest.
- pDst* Destination-Image Pointer.
- nDstStep* Destination-Image Line Step.
- oDstSizeROI* Size of the destination region-of-interest.
- nTopBorderHeight* Height of top border.
- nLeftBorderWidth* Width of left border.

Returns:

[Image Data Related Error Codes](#), [ROI Related Error Codes](#)

7.12.1.13 NppStatus nppiCopyWrapBorder_32s_AC4R (const Npp32s * pSrc, int nSrcStep, NppiSize oSrcSizeROI, Npp32s * pDst, int nDstStep, NppiSize oDstSizeROI, int nTopBorderHeight, int nLeftBorderWidth)

4 channel 32-bit signed integer image copy with the borders wrapped by replication of source image pixel colors with alpha channel unaffected.

See [nppiCopyWrapBorder_32s_C1R\(\)](#) for detailed documentation.

Parameters:

- pSrc* Source-Image Pointer.
- nSrcStep* Source-Image Line Step.
- oSrcSizeROI* Size of the source region-of-interest.
- pDst* Destination-Image Pointer.
- nDstStep* Destination-Image Line Step.
- oDstSizeROI* Size of the destination region-of-interest.
- nTopBorderHeight* Height of top border.
- nLeftBorderWidth* Width of left border.

Returns:

[Image Data Related Error Codes](#), [ROI Related Error Codes](#)

7.12.1.14 NppStatus nppiCopyWrapBorder_32s_C1R (const Npp32s * pSrc, int nSrcStep, NppiSize oSrcSizeROI, Npp32s * pDst, int nDstStep, NppiSize oDstSizeROI, int nTopBorderHeight, int nLeftBorderWidth)

1 channel 32-bit signed integer image copy with the borders wrapped by replication of source image pixel colors.

Parameters:

pSrc Source-Image Pointer.

nSrcStep Source-Image Line Step.

oSrcSizeROI Size of the source region of pixels.

pDst Destination-Image Pointer.

nDstStep Destination-Image Line Step.

oDstSizeROI Size (width, height) of the destination region, i.e. the region that gets filled with data from the source image (inner part) and a border consisting of wrapped replication of the source image pixel colors (outer part).

nTopBorderHeight Height (in pixels) of the top border. The number of pixel rows at the top of the destination ROI that will be filled with the wrapped replication of the corresponding column of source image pixels colors. $nBottomBorderHeight = oDstSizeROI.height - nTopBorderHeight - oSrcSizeROI.height$.

nLeftBorderWidth Width (in pixels) of the left border. The width of the border at the right side of the destination ROI is implicitly defined by the size of the source ROI: $nRightBorderWidth = oDstSizeROI.width - nLeftBorderWidth - oSrcSizeROI.width$.

Returns:

[Image Data Related Error Codes](#), [ROI Related Error Codes](#)

7.12.1.15 NppStatus nppiCopyWrapBorder_32s_C3R (const Npp32s * pSrc, int nSrcStep, NppiSize oSrcSizeROI, Npp32s * pDst, int nDstStep, NppiSize oDstSizeROI, int nTopBorderHeight, int nLeftBorderWidth)

3 channel 32-bit signed integer image copy with the borders wrapped by replication of source image pixel colors.

See [nppiCopyWrapBorder_32s_C1R\(\)](#) for detailed documentation.

Parameters:

pSrc Source-Image Pointer.

nSrcStep Source-Image Line Step.

oSrcSizeROI Size of the source region-of-interest.

pDst Destination-Image Pointer.

nDstStep Destination-Image Line Step.

oDstSizeROI Size of the destination region-of-interest.

nTopBorderHeight Height of top border.

nLeftBorderWidth Width of left border.

Returns:

[Image Data Related Error Codes](#), [ROI Related Error Codes](#)

7.12.1.16 NppStatus nppiCopyWrapBorder_32s_C4R (const Npp32s * pSrc, int nSrcStep, NppiSize oSrcSizeROI, Npp32s * pDst, int nDstStep, NppiSize oDstSizeROI, int nTopBorderHeight, int nLeftBorderWidth)

4 channel 32-bit signed integer image copy with the borders wrapped by replication of source image pixel colors.

See [nppiCopyWrapBorder_32s_C1R\(\)](#) for detailed documentation.

Parameters:

- pSrc* Source-Image Pointer.
- nSrcStep* Source-Image Line Step.
- oSrcSizeROI* Size of the source region-of-interest.
- pDst* Destination-Image Pointer.
- nDstStep* Destination-Image Line Step.
- oDstSizeROI* Size of the destination region-of-interest.
- nTopBorderHeight* Height of top border.
- nLeftBorderWidth* Width of left border.

Returns:

[Image Data Related Error Codes](#), [ROI Related Error Codes](#)

7.12.1.17 NppStatus nppiCopyWrapBorder_8u_AC4R (const Npp8u * pSrc, int nSrcStep, NppiSize oSrcSizeROI, Npp8u * pDst, int nDstStep, NppiSize oDstSizeROI, int nTopBorderHeight, int nLeftBorderWidth)

4 channel 8-bit unsigned integer image copy with the borders wrapped by replication of source image pixel colors with alpha channel unaffected.

See [nppiCopyWrapBorder_8u_C1R\(\)](#) for detailed documentation.

Parameters:

- pSrc* Source-Image Pointer.
- nSrcStep* Source-Image Line Step.
- oSrcSizeROI* Size of the source region-of-interest.
- pDst* Destination-Image Pointer.
- nDstStep* Destination-Image Line Step.
- oDstSizeROI* Size of the destination region-of-interest.
- nTopBorderHeight* Height of top border.
- nLeftBorderWidth* Width of left border.

Returns:

[Image Data Related Error Codes](#), [ROI Related Error Codes](#)

7.12.1.18 NppStatus nppiCopyWrapBorder_8u_C1R (const Npp8u * pSrc, int nSrcStep, NppiSize oSrcSizeROI, Npp8u * pDst, int nDstStep, NppiSize oDstSizeROI, int nTopBorderHeight, int nLeftBorderWidth)

1 channel 8-bit unsigned integer image copy with the borders wrapped by replication of source image pixel colors.

Parameters:

pSrc Source-Image Pointer.

nSrcStep Source-Image Line Step.

oSrcSizeROI Size of the source region of pixels.

pDst Destination-Image Pointer.

nDstStep Destination-Image Line Step.

oDstSizeROI Size (width, height) of the destination region, i.e. the region that gets filled with data from the source image (inner part) and a border consisting of wrapped replication of the source image pixel colors (outer part).

nTopBorderHeight Height (in pixels) of the top border. The number of pixel rows at the top of the destination ROI that will be filled with the wrapped replication of the corresponding column of source image pixels colors. $nBottomBorderHeight = oDstSizeROI.height - nTopBorderHeight - oSrcSizeROI.height$.

nLeftBorderWidth Width (in pixels) of the left border. The width of the border at the right side of the destination ROI is implicitly defined by the size of the source ROI: $nRightBorderWidth = oDstSizeROI.width - nLeftBorderWidth - oSrcSizeROI.width$.

Returns:

[Image Data Related Error Codes](#), [ROI Related Error Codes](#)

7.12.1.19 NppStatus nppiCopyWrapBorder_8u_C3R (const Npp8u * pSrc, int nSrcStep, NppiSize oSrcSizeROI, Npp8u * pDst, int nDstStep, NppiSize oDstSizeROI, int nTopBorderHeight, int nLeftBorderWidth)

3 channel 8-bit unsigned integer image copy with the borders wrapped by replication of source image pixel colors.

See [nppiCopyWrapBorder_8u_C1R\(\)](#) for detailed documentation.

Parameters:

pSrc Source-Image Pointer.

nSrcStep Source-Image Line Step.

oSrcSizeROI Size of the source region-of-interest.

pDst Destination-Image Pointer.

nDstStep Destination-Image Line Step.

oDstSizeROI Size of the destination region-of-interest.

nTopBorderHeight Height of top border.

nLeftBorderWidth Width of left border.

Returns:

[Image Data Related Error Codes](#), [ROI Related Error Codes](#)

7.12.1.20 NppStatus nppiCopyWrapBorder_8u_C4R (const Npp8u * pSrc, int nSrcStep, NppiSize oSrcSizeROI, Npp8u * pDst, int nDstStep, NppiSize oDstSizeROI, int nTopBorderHeight, int nLeftBorderWidth)

4 channel 8-bit unsigned integer image copy with the borders wrapped by replication of source image pixel colors.

See [nppiCopyWrapBorder_8u_C1R\(\)](#) for detailed documentation.

Parameters:

pSrc Source-Image Pointer.

nSrcStep Source-Image Line Step.

oSrcSizeROI Size of the source region-of-interest.

pDst Destination-Image Pointer.

nDstStep Destination-Image Line Step.

oDstSizeROI Size of the destination region-of-interest.

nTopBorderHeight Height of top border.

nLeftBorderWidth Width of left border.

Returns:

[Image Data Related Error Codes](#), [ROI Related Error Codes](#)

7.13 Copy Sub-Pixel

CopySubpix

Functions for copying linearly interpolated images using source image subpixel coordinates

- **NppStatus nppiCopySubpix_8u_C1R** (const **Npp8u** *pSrc, int nSrcStep, **Npp8u** *pDst, int nDstStep, **NppiSize** oDstSizeROI, **Npp32f** nDx, **Npp32f** nDy)
1 channel 8-bit unsigned integer linearly interpolated source image subpixel coordinate color copy.
- **NppStatus nppiCopySubpix_8u_C3R** (const **Npp8u** *pSrc, int nSrcStep, **Npp8u** *pDst, int nDstStep, **NppiSize** oDstSizeROI, **Npp32f** nDx, **Npp32f** nDy)
3 channel 8-bit unsigned integer linearly interpolated source image subpixel coordinate color copy.
- **NppStatus nppiCopySubpix_8u_C4R** (const **Npp8u** *pSrc, int nSrcStep, **Npp8u** *pDst, int nDstStep, **NppiSize** oDstSizeROI, **Npp32f** nDx, **Npp32f** nDy)
4 channel 8-bit unsigned integer linearly interpolated source image subpixel coordinate color copy.
- **NppStatus nppiCopySubpix_8u_AC4R** (const **Npp8u** *pSrc, int nSrcStep, **Npp8u** *pDst, int nDstStep, **NppiSize** oDstSizeROI, **Npp32f** nDx, **Npp32f** nDy)
4 channel 8-bit unsigned integer linearly interpolated source image subpixel coordinate color copy with alpha channel unaffected.
- **NppStatus nppiCopySubpix_16u_C1R** (const **Npp16u** *pSrc, int nSrcStep, **Npp16u** *pDst, int nDstStep, **NppiSize** oDstSizeROI, **Npp32f** nDx, **Npp32f** nDy)
1 channel 16-bit unsigned integer linearly interpolated source image subpixel coordinate color copy.
- **NppStatus nppiCopySubpix_16u_C3R** (const **Npp16u** *pSrc, int nSrcStep, **Npp16u** *pDst, int nDstStep, **NppiSize** oDstSizeROI, **Npp32f** nDx, **Npp32f** nDy)
3 channel 16-bit unsigned integer linearly interpolated source image subpixel coordinate color copy.
- **NppStatus nppiCopySubpix_16u_C4R** (const **Npp16u** *pSrc, int nSrcStep, **Npp16u** *pDst, int nDstStep, **NppiSize** oDstSizeROI, **Npp32f** nDx, **Npp32f** nDy)
4 channel 16-bit unsigned integer linearly interpolated source image subpixel coordinate color copy.
- **NppStatus nppiCopySubpix_16u_AC4R** (const **Npp16u** *pSrc, int nSrcStep, **Npp16u** *pDst, int nDstStep, **NppiSize** oDstSizeROI, **Npp32f** nDx, **Npp32f** nDy)
4 channel 16-bit unsigned linearly interpolated source image subpixel coordinate color copy with alpha channel unaffected.
- **NppStatus nppiCopySubpix_16s_C1R** (const **Npp16s** *pSrc, int nSrcStep, **Npp16s** *pDst, int nDstStep, **NppiSize** oDstSizeROI, **Npp32f** nDx, **Npp32f** nDy)
1 channel 16-bit signed integer linearly interpolated source image subpixel coordinate color copy.
- **NppStatus nppiCopySubpix_16s_C3R** (const **Npp16s** *pSrc, int nSrcStep, **Npp16s** *pDst, int nDstStep, **NppiSize** oDstSizeROI, **Npp32f** nDx, **Npp32f** nDy)
3 channel 16-bit signed integer linearly interpolated source image subpixel coordinate color copy.
- **NppStatus nppiCopySubpix_16s_C4R** (const **Npp16s** *pSrc, int nSrcStep, **Npp16s** *pDst, int nDstStep, **NppiSize** oDstSizeROI, **Npp32f** nDx, **Npp32f** nDy)
4 channel 16-bit signed integer linearly interpolated source image subpixel coordinate color copy.

- [NppStatus nppiCopySubpix_16s_AC4R](#) (const [Npp16s](#) *pSrc, int nSrcStep, [Npp16s](#) *pDst, int nDstStep, [NppiSize](#) oDstSizeROI, [Npp32f](#) nDx, [Npp32f](#) nDy)

4 channel 16-bit signed integer linearly interpolated source image subpixel coordinate color copy with alpha channel unaffected.
- [NppStatus nppiCopySubpix_32s_C1R](#) (const [Npp32s](#) *pSrc, int nSrcStep, [Npp32s](#) *pDst, int nDstStep, [NppiSize](#) oDstSizeROI, [Npp32f](#) nDx, [Npp32f](#) nDy)

1 channel 32-bit signed integer linearly interpolated source image subpixel coordinate color copy.
- [NppStatus nppiCopySubpix_32s_C3R](#) (const [Npp32s](#) *pSrc, int nSrcStep, [Npp32s](#) *pDst, int nDstStep, [NppiSize](#) oDstSizeROI, [Npp32f](#) nDx, [Npp32f](#) nDy)

3 channel 32-bit signed linearly interpolated source image subpixel coordinate color copy.
- [NppStatus nppiCopySubpix_32s_C4R](#) (const [Npp32s](#) *pSrc, int nSrcStep, [Npp32s](#) *pDst, int nDstStep, [NppiSize](#) oDstSizeROI, [Npp32f](#) nDx, [Npp32f](#) nDy)

4 channel 32-bit signed integer linearly interpolated source image subpixel coordinate color copy.
- [NppStatus nppiCopySubpix_32s_AC4R](#) (const [Npp32s](#) *pSrc, int nSrcStep, [Npp32s](#) *pDst, int nDstStep, [NppiSize](#) oDstSizeROI, [Npp32f](#) nDx, [Npp32f](#) nDy)

4 channel 32-bit signed integer linearly interpolated source image subpixel coordinate color copy with alpha channel unaffected.
- [NppStatus nppiCopySubpix_32f_C1R](#) (const [Npp32f](#) *pSrc, int nSrcStep, [Npp32f](#) *pDst, int nDstStep, [NppiSize](#) oDstSizeROI, [Npp32f](#) nDx, [Npp32f](#) nDy)

1 channel 32-bit floating point linearly interpolated source image subpixel coordinate color copy.
- [NppStatus nppiCopySubpix_32f_C3R](#) (const [Npp32f](#) *pSrc, int nSrcStep, [Npp32f](#) *pDst, int nDstStep, [NppiSize](#) oDstSizeROI, [Npp32f](#) nDx, [Npp32f](#) nDy)

3 channel 32-bit floating point linearly interpolated source image subpixel coordinate color copy.
- [NppStatus nppiCopySubpix_32f_C4R](#) (const [Npp32f](#) *pSrc, int nSrcStep, [Npp32f](#) *pDst, int nDstStep, [NppiSize](#) oDstSizeROI, [Npp32f](#) nDx, [Npp32f](#) nDy)

4 channel 32-bit floating point linearly interpolated source image subpixel coordinate color copy.
- [NppStatus nppiCopySubpix_32f_AC4R](#) (const [Npp32f](#) *pSrc, int nSrcStep, [Npp32f](#) *pDst, int nDstStep, [NppiSize](#) oDstSizeROI, [Npp32f](#) nDx, [Npp32f](#) nDy)

4 channel 32-bit floating point linearly interpolated source image subpixel coordinate color copy with alpha channel unaffected.

7.13.1 Function Documentation

7.13.1.1 [NppStatus nppiCopySubpix_16s_AC4R](#) (const [Npp16s](#) *pSrc, int nSrcStep, [Npp16s](#) *pDst, int nDstStep, [NppiSize](#) oDstSizeROI, [Npp32f](#) nDx, [Npp32f](#) nDy)

4 channel 16-bit signed integer linearly interpolated source image subpixel coordinate color copy with alpha channel unaffected.

See [nppiCopySubpix_16s_C1R\(\)](#) for detailed documentation.

Parameters:

pSrc Source-Image Pointer.
nSrcStep Source-Image Line Step.
pDst Destination-Image Pointer.
nDstStep Destination-Image Line Step.
oDstSizeROI Size of the destination region-of-interest.
nDx Fractional part of source image X coordinate.
nDy Fractional part of source image Y coordinate.

Returns:

[Image Data Related Error Codes](#), [ROI Related Error Codes](#)

7.13.1.2 `NppStatus nppiCopySubpix_16s_C1R (const Npp16s * pSrc, int nSrcStep, Npp16s * pDst, int nDstStep, NppiSize oDstSizeROI, Npp32f nDx, Npp32f nDy)`

1 channel 16-bit signed integer linearly interpolated source image subpixel coordinate color copy.

Parameters:

pSrc Source-Image Pointer.
nSrcStep Source-Image Line Step.
pDst Destination-Image Pointer.
nDstStep Destination-Image Line Step.
oDstSizeROI Size (width, height) of the destination region, i.e. the region that gets filled with data from the source image, source image ROI is assumed to be same as destination image ROI.
nDx Fractional part of source image X coordinate.
nDy Fractional part of source image Y coordinate.

Returns:

[Image Data Related Error Codes](#), [ROI Related Error Codes](#)

7.13.1.3 `NppStatus nppiCopySubpix_16s_C3R (const Npp16s * pSrc, int nSrcStep, Npp16s * pDst, int nDstStep, NppiSize oDstSizeROI, Npp32f nDx, Npp32f nDy)`

3 channel 16-bit signed integer linearly interpolated source image subpixel coordinate color copy.

See [nppiCopySubpix_16s_C1R\(\)](#) for detailed documentation.

Parameters:

pSrc Source-Image Pointer.
nSrcStep Source-Image Line Step.
pDst Destination-Image Pointer.
nDstStep Destination-Image Line Step.
oDstSizeROI Size of the destination region-of-interest.

nDx Fractional part of source image X coordinate.

nDy Fractional part of source image Y coordinate.

Returns:

[Image Data Related Error Codes](#), [ROI Related Error Codes](#)

7.13.1.4 NppStatus nppiCopySubpix_16s_C4R (const Npp16s * pSrc, int nSrcStep, Npp16s * pDst, int nDstStep, NppiSize oDstSizeROI, Npp32f nDx, Npp32f nDy)

4 channel 16-bit signed integer linearly interpolated source image subpixel coordinate color copy.

See [nppiCopySubpix_16s_C1R\(\)](#) for detailed documentation.

Parameters:

pSrc Source-Image Pointer.

nSrcStep Source-Image Line Step.

pDst Destination-Image Pointer.

nDstStep Destination-Image Line Step.

oDstSizeROI Size of the destination region-of-interest.

nDx Fractional part of source image X coordinate.

nDy Fractional part of source image Y coordinate.

Returns:

[Image Data Related Error Codes](#), [ROI Related Error Codes](#)

7.13.1.5 NppStatus nppiCopySubpix_16u_AC4R (const Npp16u * pSrc, int nSrcStep, Npp16u * pDst, int nDstStep, NppiSize oDstSizeROI, Npp32f nDx, Npp32f nDy)

4 channel 16-bit unsigned linearly interpolated source image subpixel coordinate color copy with alpha channel unaffected.

See [nppiCopySubpix_16u_C1R\(\)](#) for detailed documentation.

Parameters:

pSrc Source-Image Pointer.

nSrcStep Source-Image Line Step.

pDst Destination-Image Pointer.

nDstStep Destination-Image Line Step.

oDstSizeROI Size of the destination region-of-interest.

nDx Fractional part of source image X coordinate.

nDy Fractional part of source image Y coordinate.

Returns:

[Image Data Related Error Codes](#), [ROI Related Error Codes](#)

7.13.1.6 NppStatus nppiCopySubpix_16u_C1R (const Npp16u * pSrc, int nSrcStep, Npp16u * pDst, int nDstStep, NppiSize oDstSizeROI, Npp32f nDx, Npp32f nDy)

1 channel 16-bit unsigned integer linearly interpolated source image subpixel coordinate color copy.

Parameters:

pSrc Source-Image Pointer.

nSrcStep Source-Image Line Step.

pDst Destination-Image Pointer.

nDstStep Destination-Image Line Step.

oDstSizeROI Size (width, height) of the destination region, i.e. the region that gets filled with data from the source image, source image ROI is assumed to be same as destination image ROI.

nDx Fractional part of source image X coordinate.

nDy Fractional part of source image Y coordinate.

Returns:

[Image Data Related Error Codes](#), [ROI Related Error Codes](#)

7.13.1.7 NppStatus nppiCopySubpix_16u_C3R (const Npp16u * pSrc, int nSrcStep, Npp16u * pDst, int nDstStep, NppiSize oDstSizeROI, Npp32f nDx, Npp32f nDy)

3 channel 16-bit unsigned integer linearly interpolated source image subpixel coordinate color copy.

See [nppiCopySubpix_16u_C1R\(\)](#) for detailed documentation.

Parameters:

pSrc Source-Image Pointer.

nSrcStep Source-Image Line Step.

pDst Destination-Image Pointer.

nDstStep Destination-Image Line Step.

oDstSizeROI Size of the destination region-of-interest.

nDx Fractional part of source image X coordinate.

nDy Fractional part of source image Y coordinate.

Returns:

[Image Data Related Error Codes](#), [ROI Related Error Codes](#)

7.13.1.8 NppStatus nppiCopySubpix_16u_C4R (const Npp16u * pSrc, int nSrcStep, Npp16u * pDst, int nDstStep, NppiSize oDstSizeROI, Npp32f nDx, Npp32f nDy)

4 channel 16-bit unsigned integer linearly interpolated source image subpixel coordinate color copy.

See [nppiCopySubpix_16u_C1R\(\)](#) for detailed documentation.

Parameters:

pSrc Source-Image Pointer.

nSrcStep Source-Image Line Step.
pDst Destination-Image Pointer.
nDstStep Destination-Image Line Step.
oDstSizeROI Size of the destination region-of-interest.
nDx Fractional part of source image X coordinate.
nDy Fractional part of source image Y coordinate.

Returns:

[Image Data Related Error Codes](#), [ROI Related Error Codes](#)

7.13.1.9 **NppStatus nppiCopySubpix_32f_AC4R** (const Npp32f * *pSrc*, int *nSrcStep*, Npp32f * *pDst*, int *nDstStep*, NppiSize *oDstSizeROI*, Npp32f *nDx*, Npp32f *nDy*)

4 channel 32-bit floating point linearly interpolated source image subpixel coordinate color copy with alpha channel unaffected.

See [nppiCopySubpix_32f_C1R\(\)](#) for detailed documentation.

Parameters:

pSrc Source-Image Pointer.
nSrcStep Source-Image Line Step.
pDst Destination-Image Pointer.
nDstStep Destination-Image Line Step.
oDstSizeROI Size of the destination region-of-interest.
nDx Fractional part of source image X coordinate.
nDy Fractional part of source image Y coordinate.

Returns:

[Image Data Related Error Codes](#), [ROI Related Error Codes](#)

7.13.1.10 **NppStatus nppiCopySubpix_32f_C1R** (const Npp32f * *pSrc*, int *nSrcStep*, Npp32f * *pDst*, int *nDstStep*, NppiSize *oDstSizeROI*, Npp32f *nDx*, Npp32f *nDy*)

1 channel 32-bit floating point linearly interpolated source image subpixel coordinate color copy.

Parameters:

pSrc Source-Image Pointer.
nSrcStep Source-Image Line Step.
pDst Destination-Image Pointer.
nDstStep Destination-Image Line Step.
oDstSizeROI Size (width, height) of the destination region, i.e. the region that gets filled with data from the source image, source image ROI is assumed to be same as destination image ROI.
nDx Fractional part of source image X coordinate.
nDy Fractional part of source image Y coordinate.

Returns:

[Image Data Related Error Codes](#), [ROI Related Error Codes](#)

7.13.1.11 NppStatus nppiCopySubpix_32f_C3R (const Npp32f * pSrc, int nSrcStep, Npp32f * pDst, int nDstStep, NppiSize oDstSizeROI, Npp32f nDx, Npp32f nDy)

3 channel 32-bit floating point linearly interpolated source image subpixel coordinate color copy.

See [nppiCopySubpix_32f_C1R\(\)](#) for detailed documentation.

Parameters:

pSrc Source-Image Pointer.

nSrcStep Source-Image Line Step.

pDst Destination-Image Pointer.

nDstStep Destination-Image Line Step.

oDstSizeROI Size of the destination region-of-interest.

nDx Fractional part of source image X coordinate.

nDy Fractional part of source image Y coordinate.

Returns:

[Image Data Related Error Codes](#), [ROI Related Error Codes](#)

7.13.1.12 NppStatus nppiCopySubpix_32f_C4R (const Npp32f * pSrc, int nSrcStep, Npp32f * pDst, int nDstStep, NppiSize oDstSizeROI, Npp32f nDx, Npp32f nDy)

4 channel 32-bit floating point linearly interpolated source image subpixel coordinate color copy.

See [nppiCopySubpix_32f_C1R\(\)](#) for detailed documentation.

Parameters:

pSrc Source-Image Pointer.

nSrcStep Source-Image Line Step.

pDst Destination-Image Pointer.

nDstStep Destination-Image Line Step.

oDstSizeROI Size of the destination region-of-interest.

nDx Fractional part of source image X coordinate.

nDy Fractional part of source image Y coordinate.

Returns:

[Image Data Related Error Codes](#), [ROI Related Error Codes](#)

7.13.1.13 NppStatus nppiCopySubpix_32s_AC4R (const Npp32s * pSrc, int nSrcStep, Npp32s * pDst, int nDstStep, NppiSize oDstSizeROI, Npp32f nDx, Npp32f nDy)

4 channel 32-bit signed integer linearly interpolated source image subpixel coordinate color copy with alpha channel unaffected.

See [nppiCopySubpix_32s_C1R\(\)](#) for detailed documentation.

Parameters:

pSrc Source-Image Pointer.
nSrcStep Source-Image Line Step.
pDst Destination-Image Pointer.
nDstStep Destination-Image Line Step.
oDstSizeROI Size of the destination region-of-interest.
nDx Fractional part of source image X coordinate.
nDy Fractional part of source image Y coordinate.

Returns:

[Image Data Related Error Codes](#), [ROI Related Error Codes](#)

7.13.1.14 **NppStatus nppiCopySubpix_32s_C1R** (const Npp32s * *pSrc*, int *nSrcStep*, Npp32s * *pDst*, int *nDstStep*, NppiSize *oDstSizeROI*, Npp32f *nDx*, Npp32f *nDy*)

1 channel 32-bit signed integer linearly interpolated source image subpixel coordinate color copy.

Parameters:

pSrc Source-Image Pointer.
nSrcStep Source-Image Line Step.
pDst Destination-Image Pointer.
nDstStep Destination-Image Line Step.
oDstSizeROI Size (width, height) of the destination region, i.e. the region that gets filled with data from the source image, source image ROI is assumed to be same as destination image ROI.
nDx Fractional part of source image X coordinate.
nDy Fractional part of source image Y coordinate.

Returns:

[Image Data Related Error Codes](#), [ROI Related Error Codes](#)

7.13.1.15 **NppStatus nppiCopySubpix_32s_C3R** (const Npp32s * *pSrc*, int *nSrcStep*, Npp32s * *pDst*, int *nDstStep*, NppiSize *oDstSizeROI*, Npp32f *nDx*, Npp32f *nDy*)

3 channel 32-bit signed integer linearly interpolated source image subpixel coordinate color copy.

See [nppiCopySubpix_32s_C1R\(\)](#) for detailed documentation.

Parameters:

pSrc Source-Image Pointer.
nSrcStep Source-Image Line Step.
pDst Destination-Image Pointer.
nDstStep Destination-Image Line Step.
oDstSizeROI Size of the destination region-of-interest.

nDx Fractional part of source image X coordinate.

nDy Fractional part of source image Y coordinate.

Returns:

[Image Data Related Error Codes](#), [ROI Related Error Codes](#)

7.13.1.16 `NppStatus nppiCopySubpix_32s_C4R (const Npp32s * pSrc, int nSrcStep, Npp32s * pDst, int nDstStep, NppiSize oDstSizeROI, Npp32f nDx, Npp32f nDy)`

4 channel 32-bit signed integer linearly interpolated source image subpixel coordinate color copy.

See [nppiCopySubpix_32s_C1R\(\)](#) for detailed documentation.

Parameters:

pSrc Source-Image Pointer.

nSrcStep Source-Image Line Step.

pDst Destination-Image Pointer.

nDstStep Destination-Image Line Step.

oDstSizeROI Size of the destination region-of-interest.

nDx Fractional part of source image X coordinate.

nDy Fractional part of source image Y coordinate.

Returns:

[Image Data Related Error Codes](#), [ROI Related Error Codes](#)

7.13.1.17 `NppStatus nppiCopySubpix_8u_AC4R (const Npp8u * pSrc, int nSrcStep, Npp8u * pDst, int nDstStep, NppiSize oDstSizeROI, Npp32f nDx, Npp32f nDy)`

4 channel 8-bit unsigned integer linearly interpolated source image subpixel coordinate color copy with alpha channel unaffected.

See [nppiCopySubpix_8u_C1R\(\)](#) for detailed documentation.

Parameters:

pSrc Source-Image Pointer.

nSrcStep Source-Image Line Step.

pDst Destination-Image Pointer.

nDstStep Destination-Image Line Step.

oDstSizeROI Size of the destination region-of-interest.

nDx Fractional part of source image X coordinate.

nDy Fractional part of source image Y coordinate.

Returns:

[Image Data Related Error Codes](#), [ROI Related Error Codes](#)

7.13.1.18 NppStatus nppiCopySubpix_8u_C1R (const Npp8u * pSrc, int nSrcStep, Npp8u * pDst, int nDstStep, NppiSize oDstSizeROI, Npp32f nDx, Npp32f nDy)

1 channel 8-bit unsigned integer linearly interpolated source image subpixel coordinate color copy.

Parameters:

pSrc Source-Image Pointer.

nSrcStep Source-Image Line Step.

pDst Destination-Image Pointer.

nDstStep Destination-Image Line Step.

oDstSizeROI Size (width, height) of the destination region, i.e. the region that gets filled with data from the source image, source image ROI is assumed to be same as destination image ROI.

nDx Fractional part of source image X coordinate.

nDy Fractional part of source image Y coordinate.

Returns:

[Image Data Related Error Codes](#), [ROI Related Error Codes](#)

7.13.1.19 NppStatus nppiCopySubpix_8u_C3R (const Npp8u * pSrc, int nSrcStep, Npp8u * pDst, int nDstStep, NppiSize oDstSizeROI, Npp32f nDx, Npp32f nDy)

3 channel 8-bit unsigned integer linearly interpolated source image subpixel coordinate color copy.

See [nppiCopySubpix_8u_C1R\(\)](#) for detailed documentation.

Parameters:

pSrc Source-Image Pointer.

nSrcStep Source-Image Line Step.

pDst Destination-Image Pointer.

nDstStep Destination-Image Line Step.

oDstSizeROI Size of the destination region-of-interest.

nDx Fractional part of source image X coordinate.

nDy Fractional part of source image Y coordinate.

Returns:

[Image Data Related Error Codes](#), [ROI Related Error Codes](#)

7.13.1.20 NppStatus nppiCopySubpix_8u_C4R (const Npp8u * pSrc, int nSrcStep, Npp8u * pDst, int nDstStep, NppiSize oDstSizeROI, Npp32f nDx, Npp32f nDy)

4 channel 8-bit unsigned integer linearly interpolated source image subpixel coordinate color copy.

See [nppiCopySubpix_8u_C1R\(\)](#) for detailed documentation.

Parameters:

pSrc Source-Image Pointer.

nSrcStep Source-Image Line Step.

pDst Destination-Image Pointer.

nDstStep Destination-Image Line Step.

oDstSizeROI Size of the destination region-of-interest.

nDx Fractional part of source image X coordinate.

nDy Fractional part of source image Y coordinate.

Returns:

[Image Data Related Error Codes](#), [ROI Related Error Codes](#)

7.14 Duplicate Channel

Dup

Functions for duplicating a single channel image in a multiple channel image.

- `NppStatus nppiDup_8u_C1C3R` (const `Npp8u *pSrc`, int `nSrcStep`, `Npp8u *pDst`, int `nDstStep`, `NppiSize oDstSizeROI`)
1 channel 8-bit unsigned integer source image duplicated in all 3 channels of destination image.
- `NppStatus nppiDup_8u_C1C4R` (const `Npp8u *pSrc`, int `nSrcStep`, `Npp8u *pDst`, int `nDstStep`, `NppiSize oDstSizeROI`)
1 channel 8-bit unsigned integer source image duplicated in all 4 channels of destination image.
- `NppStatus nppiDup_8u_C1AC4R` (const `Npp8u *pSrc`, int `nSrcStep`, `Npp8u *pDst`, int `nDstStep`, `NppiSize oDstSizeROI`)
1 channel 8-bit unsigned integer source image duplicated in 3 channels of 4 channel destination image with alpha channel unaffected.
- `NppStatus nppiDup_16u_C1C3R` (const `Npp16u *pSrc`, int `nSrcStep`, `Npp16u *pDst`, int `nDstStep`, `NppiSize oDstSizeROI`)
1 channel 16-bit unsigned integer source image duplicated in all 3 channels of destination image.
- `NppStatus nppiDup_16u_C1C4R` (const `Npp16u *pSrc`, int `nSrcStep`, `Npp16u *pDst`, int `nDstStep`, `NppiSize oDstSizeROI`)
1 channel 16-bit unsigned integer source image duplicated in all 4 channels of destination image.
- `NppStatus nppiDup_16u_C1AC4R` (const `Npp16u *pSrc`, int `nSrcStep`, `Npp16u *pDst`, int `nDstStep`, `NppiSize oDstSizeROI`)
1 channel 16-bit unsigned integer source image duplicated in 3 channels of 4 channel destination image with alpha channel unaffected.
- `NppStatus nppiDup_16s_C1C3R` (const `Npp16s *pSrc`, int `nSrcStep`, `Npp16s *pDst`, int `nDstStep`, `NppiSize oDstSizeROI`)
1 channel 16-bit signed integer source image duplicated in all 3 channels of destination image.
- `NppStatus nppiDup_16s_C1C4R` (const `Npp16s *pSrc`, int `nSrcStep`, `Npp16s *pDst`, int `nDstStep`, `NppiSize oDstSizeROI`)
1 channel 16-bit signed integer source image duplicated in all 4 channels of destination image.
- `NppStatus nppiDup_16s_C1AC4R` (const `Npp16s *pSrc`, int `nSrcStep`, `Npp16s *pDst`, int `nDstStep`, `NppiSize oDstSizeROI`)
1 channel 16-bit signed integer source image duplicated in 3 channels of 4 channel destination image with alpha channel unaffected.
- `NppStatus nppiDup_32s_C1C3R` (const `Npp32s *pSrc`, int `nSrcStep`, `Npp32s *pDst`, int `nDstStep`, `NppiSize oDstSizeROI`)
1 channel 32-bit signed integer source image duplicated in all 3 channels of destination image.
- `NppStatus nppiDup_32s_C1C4R` (const `Npp32s *pSrc`, int `nSrcStep`, `Npp32s *pDst`, int `nDstStep`, `NppiSize oDstSizeROI`)

1 channel 32-bit signed integer source image duplicated in all 4 channels of destination image.

- **NppStatus nppiDup_32s_C1AC4R** (const **Npp32s** *pSrc, int nSrcStep, **Npp32s** *pDst, int nDstStep, **NppiSize** oDstSizeROI)

1 channel 32-bit signed integer source image duplicated in 3 channels of 4 channel destination image with alpha channel unaffected.

- **NppStatus nppiDup_32f_C1C3R** (const **Npp32f** *pSrc, int nSrcStep, **Npp32f** *pDst, int nDstStep, **NppiSize** oDstSizeROI)

1 channel 32-bit floating point source image duplicated in all 3 channels of destination image.

- **NppStatus nppiDup_32f_C1C4R** (const **Npp32f** *pSrc, int nSrcStep, **Npp32f** *pDst, int nDstStep, **NppiSize** oDstSizeROI)

1 channel 32-bit floating point source image duplicated in all 4 channels of destination image.

- **NppStatus nppiDup_32f_C1AC4R** (const **Npp32f** *pSrc, int nSrcStep, **Npp32f** *pDst, int nDstStep, **NppiSize** oDstSizeROI)

1 channel 32-bit floating point source image duplicated in 3 channels of 4 channel destination image with alpha channel unaffected.

7.14.1 Function Documentation

7.14.1.1 NppStatus nppiDup_16s_C1AC4R (const Npp16s * pSrc, int nSrcStep, Npp16s * pDst, int nDstStep, NppiSize oDstSizeROI)

1 channel 16-bit signed integer source image duplicated in 3 channels of 4 channel destination image with alpha channel unaffected.

Parameters:

- pSrc* Source-Image Pointer.
- nSrcStep* Source-Image Line Step.
- pDst* Destination-Image Pointer.
- nDstStep* Destination-Image Line Step.
- oDstSizeROI* Size of the destination region-of-interest.

Returns:

[Image Data Related Error Codes](#), [ROI Related Error Codes](#)

7.14.1.2 NppStatus nppiDup_16s_C1C3R (const Npp16s * pSrc, int nSrcStep, Npp16s * pDst, int nDstStep, NppiSize oDstSizeROI)

1 channel 16-bit signed integer source image duplicated in all 3 channels of destination image.

Parameters:

- pSrc* Source-Image Pointer.
- nSrcStep* Source-Image Line Step.

pDst Destination-Image Pointer.

nDstStep Destination-Image Line Step.

oDstSizeROI Size (width, height) of the destination region, i.e. the region that gets filled with data from the source image, source image ROI is assumed to be same as destination image ROI.

Returns:

[Image Data Related Error Codes](#), [ROI Related Error Codes](#)

7.14.1.3 NppStatus nppiDup_16s_C1C4R (const Npp16s * pSrc, int nSrcStep, Npp16s * pDst, int nDstStep, NppiSize oDstSizeROI)

1 channel 16-bit signed integer source image duplicated in all 4 channels of destination image.

Parameters:

pSrc Source-Image Pointer.

nSrcStep Source-Image Line Step.

pDst Destination-Image Pointer.

nDstStep Destination-Image Line Step.

oDstSizeROI Size of the destination region-of-interest.

Returns:

[Image Data Related Error Codes](#), [ROI Related Error Codes](#)

7.14.1.4 NppStatus nppiDup_16u_C1AC4R (const Npp16u * pSrc, int nSrcStep, Npp16u * pDst, int nDstStep, NppiSize oDstSizeROI)

1 channel 16-bit unsigned integer source image duplicated in 3 channels of 4 channel destination image with alpha channel unaffected.

Parameters:

pSrc Source-Image Pointer.

nSrcStep Source-Image Line Step.

pDst Destination-Image Pointer.

nDstStep Destination-Image Line Step.

oDstSizeROI Size of the destination region-of-interest.

Returns:

[Image Data Related Error Codes](#), [ROI Related Error Codes](#)

7.14.1.5 NppStatus nppiDup_16u_C1C3R (const Npp16u * pSrc, int nSrcStep, Npp16u * pDst, int nDstStep, NppiSize oDstSizeROI)

1 channel 16-bit unsigned integer source image duplicated in all 3 channels of destination image.

Parameters:

pSrc Source-Image Pointer.

nSrcStep Source-Image Line Step.

pDst Destination-Image Pointer.

nDstStep Destination-Image Line Step.

oDstSizeROI Size (width, height) of the destination region, i.e. the region that gets filled with data from the source image, source image ROI is assumed to be same as destination image ROI.

Returns:

[Image Data Related Error Codes](#), [ROI Related Error Codes](#)

7.14.1.6 NppStatus nppiDup_16u_C1C4R (const Npp16u * pSrc, int nSrcStep, Npp16u * pDst, int nDstStep, NppiSize oDstSizeROI)

1 channel 16-bit unsigned integer source image duplicated in all 4 channels of destination image.

Parameters:

pSrc Source-Image Pointer.

nSrcStep Source-Image Line Step.

pDst Destination-Image Pointer.

nDstStep Destination-Image Line Step.

oDstSizeROI Size of the destination region-of-interest.

Returns:

[Image Data Related Error Codes](#), [ROI Related Error Codes](#)

7.14.1.7 NppStatus nppiDup_32f_C1AC4R (const Npp32f * pSrc, int nSrcStep, Npp32f * pDst, int nDstStep, NppiSize oDstSizeROI)

1 channel 32-bit floating point source image duplicated in 3 channels of 4 channel destination image with alpha channel unaffected.

Parameters:

pSrc Source-Image Pointer.

nSrcStep Source-Image Line Step.

pDst Destination-Image Pointer.

nDstStep Destination-Image Line Step.

oDstSizeROI Size of the destination region-of-interest.

Returns:

[Image Data Related Error Codes](#), [ROI Related Error Codes](#)

7.14.1.8 NppStatus nppiDup_32f_C1C3R (const Npp32f * pSrc, int nSrcStep, Npp32f * pDst, int nDstStep, NppiSize oDstSizeROI)

1 channel 32-bit floating point source image duplicated in all 3 channels of destination image.

Parameters:

pSrc Source-Image Pointer.

nSrcStep Source-Image Line Step.

pDst Destination-Image Pointer.

nDstStep Destination-Image Line Step.

oDstSizeROI Size (width, height) of the destination region, i.e. the region that gets filled with data from the source image, source image ROI is assumed to be same as destination image ROI.

Returns:

[Image Data Related Error Codes](#), [ROI Related Error Codes](#)

7.14.1.9 NppStatus nppiDup_32f_C1C4R (const Npp32f * pSrc, int nSrcStep, Npp32f * pDst, int nDstStep, NppiSize oDstSizeROI)

1 channel 32-bit floating point source image duplicated in all 4 channels of destination image.

Parameters:

pSrc Source-Image Pointer.

nSrcStep Source-Image Line Step.

pDst Destination-Image Pointer.

nDstStep Destination-Image Line Step.

oDstSizeROI Size of the destination region-of-interest.

Returns:

[Image Data Related Error Codes](#), [ROI Related Error Codes](#)

7.14.1.10 NppStatus nppiDup_32s_C1AC4R (const Npp32s * pSrc, int nSrcStep, Npp32s * pDst, int nDstStep, NppiSize oDstSizeROI)

1 channel 32-bit signed integer source image duplicated in 3 channels of 4 channel destination image with alpha channel unaffected.

Parameters:

pSrc Source-Image Pointer.

nSrcStep Source-Image Line Step.

pDst Destination-Image Pointer.

nDstStep Destination-Image Line Step.

oDstSizeROI Size of the destination region-of-interest.

Returns:

[Image Data Related Error Codes](#), [ROI Related Error Codes](#)

7.14.1.11 NppStatus nppiDup_32s_C1C3R (const Npp32s * pSrc, int nSrcStep, Npp32s * pDst, int nDstStep, NppiSize oDstSizeROI)

1 channel 32-bit signed integer source image duplicated in all 3 channels of destination image.

Parameters:

pSrc Source-Image Pointer.

nSrcStep Source-Image Line Step.

pDst Destination-Image Pointer.

nDstStep Destination-Image Line Step.

oDstSizeROI Size (width, height) of the destination region, i.e. the region that gets filled with data from the source image, source image ROI is assumed to be same as destination image ROI.

Returns:

[Image Data Related Error Codes](#), [ROI Related Error Codes](#)

7.14.1.12 NppStatus nppiDup_32s_C1C4R (const Npp32s * pSrc, int nSrcStep, Npp32s * pDst, int nDstStep, NppiSize oDstSizeROI)

1 channel 32-bit signed integer source image duplicated in all 4 channels of destination image.

Parameters:

pSrc Source-Image Pointer.

nSrcStep Source-Image Line Step.

pDst Destination-Image Pointer.

nDstStep Destination-Image Line Step.

oDstSizeROI Size of the destination region-of-interest.

Returns:

[Image Data Related Error Codes](#), [ROI Related Error Codes](#)

7.14.1.13 NppStatus nppiDup_8u_C1AC4R (const Npp8u * pSrc, int nSrcStep, Npp8u * pDst, int nDstStep, NppiSize oDstSizeROI)

1 channel 8-bit unsigned integer source image duplicated in 3 channels of 4 channel destination image with alpha channel unaffected.

Parameters:

pSrc Source-Image Pointer.

nSrcStep Source-Image Line Step.

pDst Destination-Image Pointer.

nDstStep Destination-Image Line Step.

oDstSizeROI Size of the destination region-of-interest.

Returns:

[Image Data Related Error Codes](#), [ROI Related Error Codes](#)

7.14.1.14 `NppStatus nppiDup_8u_C1C3R (const Npp8u * pSrc, int nSrcStep, Npp8u * pDst, int nDstStep, NppiSize oDstSizeROI)`

1 channel 8-bit unsigned integer source image duplicated in all 3 channels of destination image.

Parameters:

pSrc Source-Image Pointer.

nSrcStep Source-Image Line Step.

pDst Destination-Image Pointer.

nDstStep Destination-Image Line Step.

oDstSizeROI Size (width, height) of the destination region, i.e. the region that gets filled with data from the source image, source image ROI is assumed to be same as destination image ROI.

Returns:

[Image Data Related Error Codes](#), [ROI Related Error Codes](#)

7.14.1.15 `NppStatus nppiDup_8u_C1C4R (const Npp8u * pSrc, int nSrcStep, Npp8u * pDst, int nDstStep, NppiSize oDstSizeROI)`

1 channel 8-bit unsigned integer source image duplicated in all 4 channels of destination image.

Parameters:

pSrc Source-Image Pointer.

nSrcStep Source-Image Line Step.

pDst Destination-Image Pointer.

nDstStep Destination-Image Line Step.

oDstSizeROI Size of the destination region-of-interest.

Returns:

[Image Data Related Error Codes](#), [ROI Related Error Codes](#)

7.15 Transpose

Transpose

Methods for transposing images of various types.

Like matrix transpose, image transpose is a mirror along the image's diagonal (upper-left to lower-right corner).

- `NppStatus nppiTranspose_8u_C1R` (const `Npp8u` *pSrc, int nSrcStep, `Npp8u` *pDst, int nDstStep, `NppiSize` oSrcROI)
1 channel 8-bit unsigned int image transpose.
- `NppStatus nppiTranspose_8u_C3R` (const `Npp8u` *pSrc, int nSrcStep, `Npp8u` *pDst, int nDstStep, `NppiSize` oSrcROI)
3 channel 8-bit unsigned int image transpose.
- `NppStatus nppiTranspose_8u_C4R` (const `Npp8u` *pSrc, int nSrcStep, `Npp8u` *pDst, int nDstStep, `NppiSize` oSrcROI)
4 channel 8-bit unsigned int image transpose.
- `NppStatus nppiTranspose_16u_C1R` (const `Npp16u` *pSrc, int nSrcStep, `Npp16u` *pDst, int nDstStep, `NppiSize` oSrcROI)
1 channel 16-bit unsigned int image transpose.
- `NppStatus nppiTranspose_16u_C3R` (const `Npp16u` *pSrc, int nSrcStep, `Npp16u` *pDst, int nDstStep, `NppiSize` oSrcROI)
3 channel 16-bit unsigned int image transpose.
- `NppStatus nppiTranspose_16u_C4R` (const `Npp16u` *pSrc, int nSrcStep, `Npp16u` *pDst, int nDstStep, `NppiSize` oSrcROI)
4 channel 16-bit unsigned int image transpose.
- `NppStatus nppiTranspose_16s_C1R` (const `Npp16s` *pSrc, int nSrcStep, `Npp16s` *pDst, int nDstStep, `NppiSize` oSrcROI)
1 channel 16-bit signed int image transpose.
- `NppStatus nppiTranspose_16s_C3R` (const `Npp16s` *pSrc, int nSrcStep, `Npp16s` *pDst, int nDstStep, `NppiSize` oSrcROI)
3 channel 16-bit signed int image transpose.
- `NppStatus nppiTranspose_16s_C4R` (const `Npp16s` *pSrc, int nSrcStep, `Npp16s` *pDst, int nDstStep, `NppiSize` oSrcROI)
4 channel 16-bit signed int image transpose.
- `NppStatus nppiTranspose_32s_C1R` (const `Npp32s` *pSrc, int nSrcStep, `Npp32s` *pDst, int nDstStep, `NppiSize` oSrcROI)
1 channel 32-bit signed int image transpose.
- `NppStatus nppiTranspose_32s_C3R` (const `Npp32s` *pSrc, int nSrcStep, `Npp32s` *pDst, int nDstStep, `NppiSize` oSrcROI)

3 channel 32-bit signed int image transpose.

- **NppStatus nppiTranspose_32s_C4R** (const **Npp32s** *pSrc, int nSrcStep, **Npp32s** *pDst, int nDstStep, **NppiSize** oSrcROI)

4 channel 32-bit signed int image transpose.

- **NppStatus nppiTranspose_32f_C1R** (const **Npp32f** *pSrc, int nSrcStep, **Npp32f** *pDst, int nDstStep, **NppiSize** oSrcROI)

1 channel 32-bit floating point image transpose.

- **NppStatus nppiTranspose_32f_C3R** (const **Npp32f** *pSrc, int nSrcStep, **Npp32f** *pDst, int nDstStep, **NppiSize** oSrcROI)

3 channel 32-bit floating point image transpose.

- **NppStatus nppiTranspose_32f_C4R** (const **Npp32f** *pSrc, int nSrcStep, **Npp32f** *pDst, int nDstStep, **NppiSize** oSrcROI)

4 channel 32-bit floating point image transpose.

7.15.1 Function Documentation

7.15.1.1 **NppStatus nppiTranspose_16s_C1R** (const **Npp16s** *pSrc, int nSrcStep, **Npp16s** *pDst, int nDstStep, **NppiSize** oSrcROI)

1 channel 16-bit signed int image transpose.

Parameters:

- pSrc* Source-Image Pointer.
- nSrcStep* Source-Image Line Step.
- pDst* Pointer to the destination ROI.
- nDstStep* Destination-Image Line Step.
- oSrcROI* Region-of-Interest (ROI).

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.15.1.2 **NppStatus nppiTranspose_16s_C3R** (const **Npp16s** *pSrc, int nSrcStep, **Npp16s** *pDst, int nDstStep, **NppiSize** oSrcROI)

3 channel 16-bit signed int image transpose.

Parameters:

- pSrc* Source-Image Pointer.
- nSrcStep* Source-Image Line Step.
- pDst* Pointer to the destination ROI.
- nDstStep* Destination-Image Line Step.

oSrcROI Region-of-Interest (ROI).

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.15.1.3 NppStatus nppiTranspose_16s_C4R (const Npp16s * pSrc, int nSrcStep, Npp16s * pDst, int nDstStep, NppiSize oSrcROI)

4 channel 16-bit signed int image transpose.

Parameters:

pSrc Source-Image Pointer.

nSrcStep Source-Image Line Step.

pDst Pointer to the destination ROI.

nDstStep Destination-Image Line Step.

oSrcROI Region-of-Interest (ROI).

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.15.1.4 NppStatus nppiTranspose_16u_C1R (const Npp16u * pSrc, int nSrcStep, Npp16u * pDst, int nDstStep, NppiSize oSrcROI)

1 channel 16-bit unsigned int image transpose.

Parameters:

pSrc Source-Image Pointer.

nSrcStep Source-Image Line Step.

pDst Pointer to the destination ROI.

nDstStep Destination-Image Line Step.

oSrcROI Region-of-Interest (ROI).

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.15.1.5 NppStatus nppiTranspose_16u_C3R (const Npp16u * pSrc, int nSrcStep, Npp16u * pDst, int nDstStep, NppiSize oSrcROI)

3 channel 16-bit unsigned int image transpose.

Parameters:

pSrc Source-Image Pointer.

nSrcStep Source-Image Line Step.

pDst Pointer to the destination ROI.
nDstStep Destination-Image Line Step.
oSrcROI Region-of-Interest (ROI).

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.15.1.6 NppStatus nppiTranspose_16u_C4R (const Npp16u * pSrc, int nSrcStep, Npp16u * pDst, int nDstStep, NppiSize oSrcROI)

4 channel 16-bit unsigned int image transpose.

Parameters:

pSrc Source-Image Pointer.
nSrcStep Source-Image Line Step.
pDst Pointer to the destination ROI.
nDstStep Destination-Image Line Step.
oSrcROI Region-of-Interest (ROI).

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.15.1.7 NppStatus nppiTranspose_32f_C1R (const Npp32f * pSrc, int nSrcStep, Npp32f * pDst, int nDstStep, NppiSize oSrcROI)

1 channel 32-bit floating point image transpose.

Parameters:

pSrc Source-Image Pointer.
nSrcStep Source-Image Line Step.
pDst Pointer to the destination ROI.
nDstStep Destination-Image Line Step.
oSrcROI Region-of-Interest (ROI).

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.15.1.8 NppStatus nppiTranspose_32f_C3R (const Npp32f * pSrc, int nSrcStep, Npp32f * pDst, int nDstStep, NppiSize oSrcROI)

3 channel 32-bit floating point image transpose.

Parameters:

pSrc Source-Image Pointer.

nSrcStep Source-Image Line Step.

pDst Pointer to the destination ROI.

nDstStep Destination-Image Line Step.

oSrcROI Region-of-Interest (ROI).

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.15.1.9 NppStatus nppiTranspose_32f_C4R (const Npp32f * pSrc, int nSrcStep, Npp32f * pDst, int nDstStep, NppiSize oSrcROI)

4 channel 32-bit floating point image transpose.

Parameters:

pSrc Source-Image Pointer.

nSrcStep Source-Image Line Step.

pDst Pointer to the destination ROI.

nDstStep Destination-Image Line Step.

oSrcROI Region-of-Interest (ROI).

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.15.1.10 NppStatus nppiTranspose_32s_C1R (const Npp32s * pSrc, int nSrcStep, Npp32s * pDst, int nDstStep, NppiSize oSrcROI)

1 channel 32-bit signed int image transpose.

Parameters:

pSrc Source-Image Pointer.

nSrcStep Source-Image Line Step.

pDst Pointer to the destination ROI.

nDstStep Destination-Image Line Step.

oSrcROI Region-of-Interest (ROI).

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.15.1.11 NppStatus nppiTranspose_32s_C3R (const Npp32s * pSrc, int nSrcStep, Npp32s * pDst, int nDstStep, NppiSize oSrcROI)

3 channel 32-bit signed int image transpose.

Parameters:

- pSrc* Source-Image Pointer.
- nSrcStep* Source-Image Line Step.
- pDst* Pointer to the destination ROI.
- nDstStep* Destination-Image Line Step.
- oSrcROI* Region-of-Interest (ROI).

Returns:

[Image Data Related Error Codes](#), [ROI Related Error Codes](#)

7.15.1.12 NppStatus nppiTranspose_32s_C4R (const Npp32s * pSrc, int nSrcStep, Npp32s * pDst, int nDstStep, NppiSize oSrcROI)

4 channel 32-bit signed int image transpose.

Parameters:

- pSrc* Source-Image Pointer.
- nSrcStep* Source-Image Line Step.
- pDst* Pointer to the destination ROI.
- nDstStep* Destination-Image Line Step.
- oSrcROI* Region-of-Interest (ROI).

Returns:

[Image Data Related Error Codes](#), [ROI Related Error Codes](#)

7.15.1.13 NppStatus nppiTranspose_8u_C1R (const Npp8u * pSrc, int nSrcStep, Npp8u * pDst, int nDstStep, NppiSize oSrcROI)

1 channel 8-bit unsigned int image transpose.

Parameters:

- pSrc* Source-Image Pointer.
- nSrcStep* Source-Image Line Step.
- pDst* Pointer to the destination ROI.
- nDstStep* Destination-Image Line Step.
- oSrcROI* Region-of-Interest (ROI).

Returns:

[Image Data Related Error Codes](#), [ROI Related Error Codes](#)

7.15.1.14 NppStatus nppiTranspose_8u_C3R (const Npp8u * pSrc, int nSrcStep, Npp8u * pDst, int nDstStep, NppiSize oSrcROI)

3 channel 8-bit unsigned int image transpose.

Parameters:

pSrc Source-Image Pointer.
nSrcStep Source-Image Line Step.
pDst Pointer to the destination ROI.
nDstStep Destination-Image Line Step.
oSrcROI Region-of-Interest (ROI).

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.15.1.15 NppStatus nppiTranspose_8u_C4R (const Npp8u * pSrc, int nSrcStep, Npp8u * pDst, int nDstStep, NppiSize oSrcROI)

4 channel 8-bit unsigned int image transpose.

Parameters:

pSrc Source-Image Pointer.
nSrcStep Source-Image Line Step.
pDst Pointer to the destination ROI.
nDstStep Destination-Image Line Step.
oSrcROI Region-of-Interest (ROI).

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.16 Swap Channels

SwapChannels

Functions for swapping and duplicating channels in multiple channel images.

The methods support arbitrary permutations of the original channels, including replication and setting one or more channels to a constant value.

- `NppStatus nppiSwapChannels_8u_C3R` (const `Npp8u` *pSrc, int nSrcStep, `Npp8u` *pDst, int nDstStep, `NppiSize` oSizeROI, const int aDstOrder[3])
3 channel 8-bit unsigned integer source image to 3 channel destination image.
- `NppStatus nppiSwapChannels_8u_C3IR` (`Npp8u` *pSrcDst, int nSrcDstStep, `NppiSize` oSizeROI, const int aDstOrder[3])
3 channel 8-bit unsigned integer in place image.
- `NppStatus nppiSwapChannels_8u_C4C3R` (const `Npp8u` *pSrc, int nSrcStep, `Npp8u` *pDst, int nDstStep, `NppiSize` oSizeROI, const int aDstOrder[3])
4 channel 8-bit unsigned integer source image to 3 channel destination image.
- `NppStatus nppiSwapChannels_8u_C4R` (const `Npp8u` *pSrc, int nSrcStep, `Npp8u` *pDst, int nDstStep, `NppiSize` oSizeROI, const int aDstOrder[4])
4 channel 8-bit unsigned integer source image to 4 channel destination image.
- `NppStatus nppiSwapChannels_8u_C4IR` (`Npp8u` *pSrcDst, int nSrcDstStep, `NppiSize` oSizeROI, const int aDstOrder[4])
4 channel 8-bit unsigned integer in place image.
- `NppStatus nppiSwapChannels_8u_C3C4R` (const `Npp8u` *pSrc, int nSrcStep, `Npp8u` *pDst, int nDstStep, `NppiSize` oSizeROI, const int aDstOrder[4], const `Npp8u` nValue)
3 channel 8-bit unsigned integer source image to 4 channel destination image.
- `NppStatus nppiSwapChannels_8u_AC4R` (const `Npp8u` *pSrc, int nSrcStep, `Npp8u` *pDst, int nDstStep, `NppiSize` oSizeROI, const int aDstOrder[3])
4 channel 8-bit unsigned integer source image to 4 channel destination image with destination alpha channel unaffected.
- `NppStatus nppiSwapChannels_16u_C3R` (const `Npp16u` *pSrc, int nSrcStep, `Npp16u` *pDst, int nDstStep, `NppiSize` oSizeROI, const int aDstOrder[3])
3 channel 16-bit unsigned integer source image to 3 channel destination image.
- `NppStatus nppiSwapChannels_16u_C3IR` (`Npp16u` *pSrcDst, int nSrcDstStep, `NppiSize` oSizeROI, const int aDstOrder[3])
3 channel 16-bit unsigned integer in place image.
- `NppStatus nppiSwapChannels_16u_C4C3R` (const `Npp16u` *pSrc, int nSrcStep, `Npp16u` *pDst, int nDstStep, `NppiSize` oSizeROI, const int aDstOrder[3])
4 channel 16-bit unsigned integer source image to 3 channel destination image.
- `NppStatus nppiSwapChannels_16u_C4R` (const `Npp16u` *pSrc, int nSrcStep, `Npp16u` *pDst, int nDstStep, `NppiSize` oSizeROI, const int aDstOrder[4])

4 channel 16-bit unsigned integer source image to 4 channel destination image.

- `NppStatus nppiSwapChannels_16u_C4IR` (`Npp16u *pSrcDst`, `int nSrcDstStep`, `NppiSize oSizeROI`, `const int aDstOrder[4]`)

4 channel 16-bit unsigned integer in place image.

- `NppStatus nppiSwapChannels_16u_C3C4R` (`const Npp16u *pSrc`, `int nSrcStep`, `Npp16u *pDst`, `int nDstStep`, `NppiSize oSizeROI`, `const int aDstOrder[4]`, `const Npp16u nValue`)

3 channel 16-bit unsigned integer source image to 4 channel destination image.

- `NppStatus nppiSwapChannels_16u_AC4R` (`const Npp16u *pSrc`, `int nSrcStep`, `Npp16u *pDst`, `int nDstStep`, `NppiSize oSizeROI`, `const int aDstOrder[3]`)

4 channel 16-bit unsigned integer source image to 4 channel destination image with destination alpha channel unaffected.

- `NppStatus nppiSwapChannels_16s_C3R` (`const Npp16s *pSrc`, `int nSrcStep`, `Npp16s *pDst`, `int nDstStep`, `NppiSize oSizeROI`, `const int aDstOrder[3]`)

3 channel 16-bit signed integer source image to 3 channel destination image.

- `NppStatus nppiSwapChannels_16s_C3IR` (`Npp16s *pSrcDst`, `int nSrcDstStep`, `NppiSize oSizeROI`, `const int aDstOrder[3]`)

3 channel 16-bit signed integer in place image.

- `NppStatus nppiSwapChannels_16s_C4C3R` (`const Npp16s *pSrc`, `int nSrcStep`, `Npp16s *pDst`, `int nDstStep`, `NppiSize oSizeROI`, `const int aDstOrder[3]`)

4 channel 16-bit signed integer source image to 3 channel destination image.

- `NppStatus nppiSwapChannels_16s_C4R` (`const Npp16s *pSrc`, `int nSrcStep`, `Npp16s *pDst`, `int nDstStep`, `NppiSize oSizeROI`, `const int aDstOrder[4]`)

4 channel 16-bit signed integer source image to 4 channel destination image.

- `NppStatus nppiSwapChannels_16s_C4IR` (`Npp16s *pSrcDst`, `int nSrcDstStep`, `NppiSize oSizeROI`, `const int aDstOrder[4]`)

4 channel 16-bit signed integer in place image.

- `NppStatus nppiSwapChannels_16s_C3C4R` (`const Npp16s *pSrc`, `int nSrcStep`, `Npp16s *pDst`, `int nDstStep`, `NppiSize oSizeROI`, `const int aDstOrder[4]`, `const Npp16s nValue`)

3 channel 16-bit signed integer source image to 4 channel destination image.

- `NppStatus nppiSwapChannels_16s_AC4R` (`const Npp16s *pSrc`, `int nSrcStep`, `Npp16s *pDst`, `int nDstStep`, `NppiSize oSizeROI`, `const int aDstOrder[3]`)

4 channel 16-bit signed integer source image to 4 channel destination image with destination alpha channel unaffected.

- `NppStatus nppiSwapChannels_32s_C3R` (`const Npp32s *pSrc`, `int nSrcStep`, `Npp32s *pDst`, `int nDstStep`, `NppiSize oSizeROI`, `const int aDstOrder[3]`)

3 channel 32-bit signed integer source image to 3 channel destination image.

- `NppStatus nppiSwapChannels_32s_C3IR` (`Npp32s *pSrcDst`, `int nSrcDstStep`, `NppiSize oSizeROI`, `const int aDstOrder[3]`)

3 channel 32-bit signed integer in place image.

- [NppStatus nppiSwapChannels_32s_C4C3R](#) (const [Npp32s](#) *pSrc, int nSrcStep, [Npp32s](#) *pDst, int nDstStep, [NppiSize](#) oSizeROI, const int aDstOrder[3])
4 channel 32-bit signed integer source image to 3 channel destination image.
- [NppStatus nppiSwapChannels_32s_C4R](#) (const [Npp32s](#) *pSrc, int nSrcStep, [Npp32s](#) *pDst, int nDstStep, [NppiSize](#) oSizeROI, const int aDstOrder[4])
4 channel 32-bit signed integer source image to 4 channel destination image.
- [NppStatus nppiSwapChannels_32s_C4IR](#) ([Npp32s](#) *pSrcDst, int nSrcDstStep, [NppiSize](#) oSizeROI, const int aDstOrder[4])
4 channel 32-bit signed integer in place image.
- [NppStatus nppiSwapChannels_32s_C3C4R](#) (const [Npp32s](#) *pSrc, int nSrcStep, [Npp32s](#) *pDst, int nDstStep, [NppiSize](#) oSizeROI, const int aDstOrder[4], const [Npp32s](#) nValue)
3 channel 32-bit signed integer source image to 4 channel destination image.
- [NppStatus nppiSwapChannels_32s_AC4R](#) (const [Npp32s](#) *pSrc, int nSrcStep, [Npp32s](#) *pDst, int nDstStep, [NppiSize](#) oSizeROI, const int aDstOrder[3])
4 channel 32-bit signed integer source image to 4 channel destination image with destination alpha channel unaffected.
- [NppStatus nppiSwapChannels_32f_C3R](#) (const [Npp32f](#) *pSrc, int nSrcStep, [Npp32f](#) *pDst, int nDstStep, [NppiSize](#) oSizeROI, const int aDstOrder[3])
3 channel 32-bit floating point source image to 3 channel destination image.
- [NppStatus nppiSwapChannels_32f_C3IR](#) ([Npp32f](#) *pSrcDst, int nSrcDstStep, [NppiSize](#) oSizeROI, const int aDstOrder[3])
3 channel 32-bit floating point in place image.
- [NppStatus nppiSwapChannels_32f_C4C3R](#) (const [Npp32f](#) *pSrc, int nSrcStep, [Npp32f](#) *pDst, int nDstStep, [NppiSize](#) oSizeROI, const int aDstOrder[3])
4 channel 32-bit floating point source image to 3 channel destination image.
- [NppStatus nppiSwapChannels_32f_C4R](#) (const [Npp32f](#) *pSrc, int nSrcStep, [Npp32f](#) *pDst, int nDstStep, [NppiSize](#) oSizeROI, const int aDstOrder[4])
4 channel 32-bit floating point source image to 4 channel destination image.
- [NppStatus nppiSwapChannels_32f_C4IR](#) ([Npp32f](#) *pSrcDst, int nSrcDstStep, [NppiSize](#) oSizeROI, const int aDstOrder[4])
4 channel 32-bit floating point in place image.
- [NppStatus nppiSwapChannels_32f_C3C4R](#) (const [Npp32f](#) *pSrc, int nSrcStep, [Npp32f](#) *pDst, int nDstStep, [NppiSize](#) oSizeROI, const int aDstOrder[4], const [Npp32f](#) nValue)
3 channel 32-bit floating point source image to 4 channel destination image.
- [NppStatus nppiSwapChannels_32f_AC4R](#) (const [Npp32f](#) *pSrc, int nSrcStep, [Npp32f](#) *pDst, int nDstStep, [NppiSize](#) oSizeROI, const int aDstOrder[3])
4 channel 32-bit floating point source image to 4 channel destination image with destination alpha channel unaffected.

7.16.1 Function Documentation

7.16.1.1 `NppStatus nppiSwapChannels_16s_AC4R (const Npp16s * pSrc, int nSrcStep, Npp16s * pDst, int nDstStep, NppiSize oSizeROI, const int aDstOrder[3])`

4 channel 16-bit signed integer source image to 4 channel destination image with destination alpha channel unaffected.

Parameters:

pSrc Source-Image Pointer.

nSrcStep Source-Image Line Step.

pDst Destination-Image Pointer.

nDstStep Destination-Image Line Step.

oSizeROI Region-of-Interest (ROI).

aDstOrder Host memory integer array describing how channel values are permuted. The n-th entry of the array contains the number of the channel that is stored in the n-th channel of the output image. E.g. Given an RGBA image, `aDstOrder = [2,1,0]` converts this to BGRA channel order. In the AC4R case, the alpha channel is always assumed to be channel 3.

Returns:

[Image Data Related Error Codes](#), [ROI Related Error Codes](#)

7.16.1.2 `NppStatus nppiSwapChannels_16s_C3C4R (const Npp16s * pSrc, int nSrcStep, Npp16s * pDst, int nDstStep, NppiSize oSizeROI, const int aDstOrder[4], const Npp16s nValue)`

3 channel 16-bit signed integer source image to 4 channel destination image.

Parameters:

pSrc Source-Image Pointer.

nSrcStep Source-Image Line Step.

pDst Destination-Image Pointer.

nDstStep Destination-Image Line Step.

oSizeROI Region-of-Interest (ROI).

aDstOrder Host memory integer array describing how channel values are permuted. The n-th entry of the array contains the number of the channel that is stored in the n-th channel of the output image. E.g. Given an RGB image, `aDstOrder = [3,2,1,0]` converts this to VBGR channel order.

nValue (V) Single channel constant value that can be replicated in one or more of the 4 destination channels. `nValue` is either written or not written to a particular channel depending on the `aDstOrder` entry for that destination channel. An `aDstOrder` value of 3 will output `nValue` to that channel, an `aDstOrder` value greater than 3 will leave that particular destination channel value unmodified.

Returns:

[Image Data Related Error Codes](#), [ROI Related Error Codes](#)

7.16.1.3 NppStatus nppiSwapChannels_16s_C3IR (Npp16s * pSrcDst, int nSrcDstStep, NppiSize oSizeROI, const int aDstOrder[3])

3 channel 16-bit signed integer in place image.

Parameters:

pSrcDst In-Place Image Pointer.

nSrcDstStep In-Place-Image Line Step.

oSizeROI Region-of-Interest (ROI).

aDstOrder Host memory integer array describing how channel values are permuted. The n-th entry of the array contains the number of the channel that is stored in the n-th channel of the output image. E.g. Given an RGB image, aDstOrder = [2,1,0] converts this to BGR channel order.

Returns:

[Image Data Related Error Codes](#), [ROI Related Error Codes](#)

7.16.1.4 NppStatus nppiSwapChannels_16s_C3R (const Npp16s * pSrc, int nSrcStep, Npp16s * pDst, int nDstStep, NppiSize oSizeROI, const int aDstOrder[3])

3 channel 16-bit signed integer source image to 3 channel destination image.

Parameters:

pSrc Source-Image Pointer.

nSrcStep Source-Image Line Step.

pDst Destination-Image Pointer.

nDstStep Destination-Image Line Step.

oSizeROI Region-of-Interest (ROI).

aDstOrder Host memory integer array describing how channel values are permuted. The n-th entry of the array contains the number of the channel that is stored in the n-th channel of the output image. E.g. Given an RGB image, aDstOrder = [2,1,0] converts this to BGR channel order.

Returns:

[Image Data Related Error Codes](#), [ROI Related Error Codes](#)

7.16.1.5 NppStatus nppiSwapChannels_16s_C4C3R (const Npp16s * pSrc, int nSrcStep, Npp16s * pDst, int nDstStep, NppiSize oSizeROI, const int aDstOrder[3])

4 channel 16-bit signed integer source image to 3 channel destination image.

Parameters:

pSrc Source-Image Pointer.

nSrcStep Source-Image Line Step.

pDst Destination-Image Pointer.

nDstStep Destination-Image Line Step.

oSizeROI [Region-of-Interest \(ROI\)](#).

aDstOrder Host memory integer array describing how channel values are permuted. The n-th entry of the array contains the number of the channel that is stored in the n-th channel of the output image. E.g. Given an RGBA image, *aDstOrder* = [2,1,0] converts this to a 3 channel BGR channel order.

Returns:

[Image Data Related Error Codes](#), [ROI Related Error Codes](#)

7.16.1.6 NppStatus nppiSwapChannels_16s_C4IR (Npp16s * pSrcDst, int nSrcDstStep, NppiSize oSizeROI, const int aDstOrder[4])

4 channel 16-bit signed integer in place image.

Parameters:

pSrcDst [In-Place Image Pointer](#).

nSrcDstStep [In-Place-Image Line Step](#).

oSizeROI [Region-of-Interest \(ROI\)](#).

aDstOrder Host memory integer array describing how channel values are permuted. The n-th entry of the array contains the number of the channel that is stored in the n-th channel of the output image. E.g. Given an ARGB image, *aDstOrder* = [3,2,1,0] converts this to BGRA channel order.

Returns:

[Image Data Related Error Codes](#), [ROI Related Error Codes](#)

7.16.1.7 NppStatus nppiSwapChannels_16s_C4R (const Npp16s * pSrc, int nSrcStep, Npp16s * pDst, int nDstStep, NppiSize oSizeROI, const int aDstOrder[4])

4 channel 16-bit signed integer source image to 4 channel destination image.

Parameters:

pSrc [Source-Image Pointer](#).

nSrcStep [Source-Image Line Step](#).

pDst [Destination-Image Pointer](#).

nDstStep [Destination-Image Line Step](#).

oSizeROI [Region-of-Interest \(ROI\)](#).

aDstOrder Host memory integer array describing how channel values are permuted. The n-th entry of the array contains the number of the channel that is stored in the n-th channel of the output image. E.g. Given an ARGB image, *aDstOrder* = [3,2,1,0] converts this to BGRA channel order.

Returns:

[Image Data Related Error Codes](#), [ROI Related Error Codes](#)

7.16.1.8 NppStatus nppiSwapChannels_16u_AC4R (const Npp16u * pSrc, int nSrcStep, Npp16u * pDst, int nDstStep, NppiSize oSizeROI, const int aDstOrder[3])

4 channel 16-bit unsigned integer source image to 4 channel destination image with destination alpha channel unaffected.

Parameters:

pSrc Source-Image Pointer.

nSrcStep Source-Image Line Step.

pDst Destination-Image Pointer.

nDstStep Destination-Image Line Step.

oSizeROI Region-of-Interest (ROI).

aDstOrder Host memory integer array describing how channel values are permuted. The n-th entry of the array contains the number of the channel that is stored in the n-th channel of the output image. E.g. Given an RGBA image, aDstOrder = [2,1,0] converts this to BGRA channel order. In the AC4R case, the alpha channel is always assumed to be channel 3.

Returns:

[Image Data Related Error Codes](#), [ROI Related Error Codes](#)

7.16.1.9 NppStatus nppiSwapChannels_16u_C3C4R (const Npp16u * pSrc, int nSrcStep, Npp16u * pDst, int nDstStep, NppiSize oSizeROI, const int aDstOrder[4], const Npp16u nValue)

3 channel 16-bit unsigned integer source image to 4 channel destination image.

Parameters:

pSrc Source-Image Pointer.

nSrcStep Source-Image Line Step.

pDst Destination-Image Pointer.

nDstStep Destination-Image Line Step.

oSizeROI Region-of-Interest (ROI).

aDstOrder Host memory integer array describing how channel values are permuted. The n-th entry of the array contains the number of the channel that is stored in the n-th channel of the output image. E.g. Given an RGB image, aDstOrder = [3,2,1,0] converts this to VBGR channel order.

nValue (V) Single channel constant value that can be replicated in one or more of the 4 destination channels. nValue is either written or not written to a particular channel depending on the aDstOrder entry for that destination channel. An aDstOrder value of 3 will output nValue to that channel, an aDstOrder value greater than 3 will leave that particular destination channel value unmodified.

Returns:

[Image Data Related Error Codes](#), [ROI Related Error Codes](#)

7.16.1.10 NppStatus nppiSwapChannels_16u_C3IR (Npp16u * pSrcDst, int nSrcDstStep, NppiSize oSizeROI, const int aDstOrder[3])

3 channel 16-bit unsigned integer in place image.

Parameters:

pSrcDst In-Place Image Pointer.

nSrcDstStep In-Place-Image Line Step.

oSizeROI Region-of-Interest (ROI).

aDstOrder Host memory integer array describing how channel values are permuted. The n-th entry of the array contains the number of the channel that is stored in the n-th channel of the output image. E.g. Given an RGB image, aDstOrder = [2,1,0] converts this to BGR channel order.

Returns:

[Image Data Related Error Codes](#), [ROI Related Error Codes](#)

7.16.1.11 NppStatus nppiSwapChannels_16u_C3R (const Npp16u * pSrc, int nSrcStep, Npp16u * pDst, int nDstStep, NppiSize oSizeROI, const int aDstOrder[3])

3 channel 16-bit unsigned integer source image to 3 channel destination image.

Parameters:

pSrc Source-Image Pointer.

nSrcStep Source-Image Line Step.

pDst Destination-Image Pointer.

nDstStep Destination-Image Line Step.

oSizeROI Region-of-Interest (ROI).

aDstOrder Host memory integer array describing how channel values are permuted. The n-th entry of the array contains the number of the channel that is stored in the n-th channel of the output image. E.g. Given an RGB image, aDstOrder = [2,1,0] converts this to BGR channel order.

Returns:

[Image Data Related Error Codes](#), [ROI Related Error Codes](#)

7.16.1.12 NppStatus nppiSwapChannels_16u_C4C3R (const Npp16u * pSrc, int nSrcStep, Npp16u * pDst, int nDstStep, NppiSize oSizeROI, const int aDstOrder[3])

4 channel 16-bit unsigned integer source image to 3 channel destination image.

Parameters:

pSrc Source-Image Pointer.

nSrcStep Source-Image Line Step.

pDst Destination-Image Pointer.

nDstStep Destination-Image Line Step.

oSizeROI [Region-of-Interest \(ROI\)](#).

aDstOrder Host memory integer array describing how channel values are permuted. The n-th entry of the array contains the number of the channel that is stored in the n-th channel of the output image. E.g. Given an RGBA image, *aDstOrder* = [2,1,0] converts this to a 3 channel BGR channel order.

Returns:

[Image Data Related Error Codes](#), [ROI Related Error Codes](#)

7.16.1.13 `NppStatus nppiSwapChannels_16u_C4IR (Npp16u * pSrcDst, int nSrcDstStep, NppiSize oSizeROI, const int aDstOrder[4])`

4 channel 16-bit unsigned integer in place image.

Parameters:

pSrcDst [In-Place Image Pointer](#).

nSrcDstStep [In-Place-Image Line Step](#).

oSizeROI [Region-of-Interest \(ROI\)](#).

aDstOrder Host memory integer array describing how channel values are permuted. The n-th entry of the array contains the number of the channel that is stored in the n-th channel of the output image. E.g. Given an ARGB image, *aDstOrder* = [3,2,1,0] converts this to BGRA channel order.

Returns:

[Image Data Related Error Codes](#), [ROI Related Error Codes](#)

7.16.1.14 `NppStatus nppiSwapChannels_16u_C4R (const Npp16u * pSrc, int nSrcStep, Npp16u * pDst, int nDstStep, NppiSize oSizeROI, const int aDstOrder[4])`

4 channel 16-bit unsigned integer source image to 4 channel destination image.

Parameters:

pSrc [Source-Image Pointer](#).

nSrcStep [Source-Image Line Step](#).

pDst [Destination-Image Pointer](#).

nDstStep [Destination-Image Line Step](#).

oSizeROI [Region-of-Interest \(ROI\)](#).

aDstOrder Host memory integer array describing how channel values are permuted. The n-th entry of the array contains the number of the channel that is stored in the n-th channel of the output image. E.g. Given an ARGB image, *aDstOrder* = [3,2,1,0] converts this to BGRA channel order.

Returns:

[Image Data Related Error Codes](#), [ROI Related Error Codes](#)

7.16.1.15 NppStatus nppiSwapChannels_32f_AC4R (const Npp32f * pSrc, int nSrcStep, Npp32f * pDst, int nDstStep, NppiSize oSizeROI, const int aDstOrder[3])

4 channel 32-bit floating point source image to 4 channel destination image with destination alpha channel unaffected.

Parameters:

pSrc Source-Image Pointer.

nSrcStep Source-Image Line Step.

pDst Destination-Image Pointer.

nDstStep Destination-Image Line Step.

oSizeROI Region-of-Interest (ROI).

aDstOrder Host memory integer array describing how channel values are permuted. The n-th entry of the array contains the number of the channel that is stored in the n-th channel of the output image. E.g. Given an RGBA image, *aDstOrder* = [2,1,0] converts this to BGRA channel order. In the AC4R case, the alpha channel is always assumed to be channel 3.

Returns:

[Image Data Related Error Codes](#), [ROI Related Error Codes](#)

7.16.1.16 NppStatus nppiSwapChannels_32f_C3C4R (const Npp32f * pSrc, int nSrcStep, Npp32f * pDst, int nDstStep, NppiSize oSizeROI, const int aDstOrder[4], const Npp32f nValue)

3 channel 32-bit floating point source image to 4 channel destination image.

Parameters:

pSrc Source-Image Pointer.

nSrcStep Source-Image Line Step.

pDst Destination-Image Pointer.

nDstStep Destination-Image Line Step.

oSizeROI Region-of-Interest (ROI).

aDstOrder Host memory integer array describing how channel values are permuted. The n-th entry of the array contains the number of the channel that is stored in the n-th channel of the output image. E.g. Given an RGB image, *aDstOrder* = [3,2,1,0] converts this to VBGR channel order.

nValue (V) Single channel constant value that can be replicated in one or more of the 4 destination channels. *nValue* is either written or not written to a particular channel depending on the *aDstOrder* entry for that destination channel. An *aDstOrder* value of 3 will output *nValue* to that channel, an *aDstOrder* value greater than 3 will leave that particular destination channel value unmodified.

Returns:

[Image Data Related Error Codes](#), [ROI Related Error Codes](#)

7.16.1.17 NppStatus nppiSwapChannels_32f_C3IR (Npp32f * pSrcDst, int nSrcDstStep, NppiSize oSizeROI, const int aDstOrder[3])

3 channel 32-bit floating point in place image.

Parameters:

pSrcDst In-Place Image Pointer.

nSrcDstStep In-Place-Image Line Step.

oSizeROI oSizeROI Region-of-Interest (ROI).

aDstOrder Host memory integer array describing how channel values are permuted. The n-th entry of the array contains the number of the channel that is stored in the n-th channel of the output image. E.g. Given an RGB image, aDstOrder = [2,1,0] converts this to BGR channel order.

Returns:

[Image Data Related Error Codes](#), [ROI Related Error Codes](#)

7.16.1.18 NppStatus nppiSwapChannels_32f_C3R (const Npp32f * pSrc, int nSrcStep, Npp32f * pDst, int nDstStep, NppiSize oSizeROI, const int aDstOrder[3])

3 channel 32-bit floating point source image to 3 channel destination image.

Parameters:

pSrc Source-Image Pointer.

nSrcStep Source-Image Line Step.

pDst Destination-Image Pointer.

nDstStep Destination-Image Line Step.

oSizeROI Region-of-Interest (ROI).

aDstOrder Host memory integer array describing how channel values are permuted. The n-th entry of the array contains the number of the channel that is stored in the n-th channel of the output image. E.g. Given an RGB image, aDstOrder = [2,1,0] converts this to BGR channel order.

Returns:

[Image Data Related Error Codes](#), [ROI Related Error Codes](#)

7.16.1.19 NppStatus nppiSwapChannels_32f_C4C3R (const Npp32f * pSrc, int nSrcStep, Npp32f * pDst, int nDstStep, NppiSize oSizeROI, const int aDstOrder[3])

4 channel 32-bit floating point source image to 3 channel destination image.

Parameters:

pSrc Source-Image Pointer.

nSrcStep Source-Image Line Step.

pDst Destination-Image Pointer.

nDstStep Destination-Image Line Step.

oSizeROI [Region-of-Interest \(ROI\)](#).

aDstOrder Host memory integer array describing how channel values are permuted. The n-th entry of the array contains the number of the channel that is stored in the n-th channel of the output image. E.g. Given an RGBA image, *aDstOrder* = [2,1,0] converts this to a 3 channel BGR channel order.

Returns:

[Image Data Related Error Codes](#), [ROI Related Error Codes](#)

7.16.1.20 `NppStatus nppiSwapChannels_32f_C4IR (Npp32f * pSrcDst, int nSrcDstStep, NppiSize oSizeROI, const int aDstOrder[4])`

4 channel 32-bit floating point in place image.

Parameters:

pSrcDst [In-Place Image Pointer](#).

nSrcDstStep [In-Place-Image Line Step](#).

oSizeROI [Region-of-Interest \(ROI\)](#).

aDstOrder Host memory integer array describing how channel values are permuted. The n-th entry of the array contains the number of the channel that is stored in the n-th channel of the output image. E.g. Given an ARGB image, *aDstOrder* = [3,2,1,0] converts this to BGRA channel order.

Returns:

[Image Data Related Error Codes](#), [ROI Related Error Codes](#)

7.16.1.21 `NppStatus nppiSwapChannels_32f_C4R (const Npp32f * pSrc, int nSrcStep, Npp32f * pDst, int nDstStep, NppiSize oSizeROI, const int aDstOrder[4])`

4 channel 32-bit floating point source image to 4 channel destination image.

Parameters:

pSrc [Source-Image Pointer](#).

nSrcStep [Source-Image Line Step](#).

pDst [Destination-Image Pointer](#).

nDstStep [Destination-Image Line Step](#).

oSizeROI [Region-of-Interest \(ROI\)](#).

aDstOrder Host memory integer array describing how channel values are permuted. The n-th entry of the array contains the number of the channel that is stored in the n-th channel of the output image. E.g. Given an ARGB image, *aDstOrder* = [3,2,1,0] converts this to BGRA channel order.

Returns:

[Image Data Related Error Codes](#), [ROI Related Error Codes](#)

7.16.1.22 `NppStatus nppiSwapChannels_32s_AC4R (const Npp32s * pSrc, int nSrcStep, Npp32s * pDst, int nDstStep, NppiSize oSizeROI, const int aDstOrder[3])`

4 channel 32-bit signed integer source image to 4 channel destination image with destination alpha channel unaffected.

Parameters:

pSrc Source-Image Pointer.

nSrcStep Source-Image Line Step.

pDst Destination-Image Pointer.

nDstStep Destination-Image Line Step.

oSizeROI Region-of-Interest (ROI).

aDstOrder Host memory integer array describing how channel values are permuted. The n-th entry of the array contains the number of the channel that is stored in the n-th channel of the output image. E.g. Given an RGBA image, `aDstOrder = [2,1,0]` converts this to BGRA channel order. In the AC4R case, the alpha channel is always assumed to be channel 3.

Returns:

[Image Data Related Error Codes](#), [ROI Related Error Codes](#)

7.16.1.23 `NppStatus nppiSwapChannels_32s_C3C4R (const Npp32s * pSrc, int nSrcStep, Npp32s * pDst, int nDstStep, NppiSize oSizeROI, const int aDstOrder[4], const Npp32s nValue)`

3 channel 32-bit signed integer source image to 4 channel destination image.

Parameters:

pSrc Source-Image Pointer.

nSrcStep Source-Image Line Step.

pDst Destination-Image Pointer.

nDstStep Destination-Image Line Step.

oSizeROI Region-of-Interest (ROI).

aDstOrder Host memory integer array describing how channel values are permuted. The n-th entry of the array contains the number of the channel that is stored in the n-th channel of the output image. E.g. Given an RGB image, `aDstOrder = [3,2,1,0]` converts this to VBGR channel order.

nValue (V) Single channel constant value that can be replicated in one or more of the 4 destination channels. `nValue` is either written or not written to a particular channel depending on the `aDstOrder` entry for that destination channel. An `aDstOrder` value of 3 will output `nValue` to that channel, an `aDstOrder` value greater than 3 will leave that particular destination channel value unmodified.

Returns:

[Image Data Related Error Codes](#), [ROI Related Error Codes](#)

7.16.1.24 NppStatus nppiSwapChannels_32s_C3IR (Npp32s * pSrcDst, int nSrcDstStep, NppiSize oSizeROI, const int aDstOrder[3])

3 channel 32-bit signed integer in place image.

Parameters:

pSrcDst In-Place Image Pointer.

nSrcDstStep In-Place-Image Line Step.

oSizeROI Region-of-Interest (ROI).

aDstOrder Host memory integer array describing how channel values are permuted. The n-th entry of the array contains the number of the channel that is stored in the n-th channel of the output image. E.g. Given an RGB image, aDstOrder = [2,1,0] converts this to BGR channel order.

Returns:

[Image Data Related Error Codes](#), [ROI Related Error Codes](#)

7.16.1.25 NppStatus nppiSwapChannels_32s_C3R (const Npp32s * pSrc, int nSrcStep, Npp32s * pDst, int nDstStep, NppiSize oSizeROI, const int aDstOrder[3])

3 channel 32-bit signed integer source image to 3 channel destination image.

Parameters:

pSrc Source-Image Pointer.

nSrcStep Source-Image Line Step.

pDst Destination-Image Pointer.

nDstStep Destination-Image Line Step.

oSizeROI Region-of-Interest (ROI).

aDstOrder Host memory integer array describing how channel values are permuted. The n-th entry of the array contains the number of the channel that is stored in the n-th channel of the output image. E.g. Given an RGB image, aDstOrder = [2,1,0] converts this to BGR channel order.

Returns:

[Image Data Related Error Codes](#), [ROI Related Error Codes](#)

7.16.1.26 NppStatus nppiSwapChannels_32s_C4C3R (const Npp32s * pSrc, int nSrcStep, Npp32s * pDst, int nDstStep, NppiSize oSizeROI, const int aDstOrder[3])

4 channel 32-bit signed integer source image to 3 channel destination image.

Parameters:

pSrc Source-Image Pointer.

nSrcStep Source-Image Line Step.

pDst Destination-Image Pointer.

nDstStep Destination-Image Line Step.

oSizeROI [Region-of-Interest \(ROI\)](#).

aDstOrder Host memory integer array describing how channel values are permuted. The n-th entry of the array contains the number of the channel that is stored in the n-th channel of the output image. E.g. Given an RGBA image, *aDstOrder* = [2,1,0] converts this to a 3 channel BGR channel order.

Returns:

[Image Data Related Error Codes](#), [ROI Related Error Codes](#)

7.16.1.27 `NppStatus nppiSwapChannels_32s_C4IR (Npp32s * pSrcDst, int nSrcDstStep, NppiSize oSizeROI, const int aDstOrder[4])`

4 channel 32-bit signed integer in place image.

Parameters:

pSrcDst [In-Place Image Pointer](#).

nSrcDstStep [In-Place-Image Line Step](#).

oSizeROI [Region-of-Interest \(ROI\)](#).

aDstOrder Host memory integer array describing how channel values are permuted. The n-th entry of the array contains the number of the channel that is stored in the n-th channel of the output image. E.g. Given an ARGB image, *aDstOrder* = [3,2,1,0] converts this to BGRA channel order.

Returns:

[Image Data Related Error Codes](#), [ROI Related Error Codes](#)

7.16.1.28 `NppStatus nppiSwapChannels_32s_C4R (const Npp32s * pSrc, int nSrcStep, Npp32s * pDst, int nDstStep, NppiSize oSizeROI, const int aDstOrder[4])`

4 channel 32-bit signed integer source image to 4 channel destination image.

Parameters:

pSrc [Source-Image Pointer](#).

nSrcStep [Source-Image Line Step](#).

pDst [Destination-Image Pointer](#).

nDstStep [Destination-Image Line Step](#).

oSizeROI [Region-of-Interest \(ROI\)](#).

aDstOrder Host memory integer array describing how channel values are permuted. The n-th entry of the array contains the number of the channel that is stored in the n-th channel of the output image. E.g. Given an ARGB image, *aDstOrder* = [3,2,1,0] converts this to BGRA channel order.

Returns:

[Image Data Related Error Codes](#), [ROI Related Error Codes](#)

7.16.1.29 NppStatus nppiSwapChannels_8u_AC4R (const Npp8u * pSrc, int nSrcStep, Npp8u * pDst, int nDstStep, NppiSize oSizeROI, const int aDstOrder[3])

4 channel 8-bit unsigned integer source image to 4 channel destination image with destination alpha channel unaffected.

Parameters:

pSrc Source-Image Pointer.

nSrcStep Source-Image Line Step.

pDst Destination-Image Pointer.

nDstStep Destination-Image Line Step.

oSizeROI Region-of-Interest (ROI).

aDstOrder Host memory integer array describing how channel values are permuted. The n-th entry of the array contains the number of the channel that is stored in the n-th channel of the output image. E.g. Given an RGB image, *aDstOrder* = [3,2,1,0] converts this to VBGR channel order. The n-th entry of the array contains the number of the channel that is stored in the n-th channel of the output image. E.g. Given an RGBA image, *aDstOrder* = [2,1,0] converts this to BGRA channel order. In the AC4R case, the alpha channel is always assumed to be channel 3.

Returns:

[Image Data Related Error Codes](#), [ROI Related Error Codes](#)

7.16.1.30 NppStatus nppiSwapChannels_8u_C3C4R (const Npp8u * pSrc, int nSrcStep, Npp8u * pDst, int nDstStep, NppiSize oSizeROI, const int aDstOrder[4], const Npp8u nValue)

3 channel 8-bit unsigned integer source image to 4 channel destination image.

Parameters:

pSrc Source-Image Pointer.

nSrcStep Source-Image Line Step.

pDst Destination-Image Pointer.

nDstStep Destination-Image Line Step.

oSizeROI Region-of-Interest (ROI).

aDstOrder Host memory integer array describing how channel values are permuted. The n-th entry of the array contains the number of the channel that is stored in the n-th channel of the output image. E.g. Given an RGB image, *aDstOrder* = [3,2,1,0] converts this to VBGR channel order.

nValue (V) Single channel constant value that can be replicated in one or more of the 4 destination channels. *nValue* is either written or not written to a particular channel depending on the *aDstOrder* entry for that destination channel. An *aDstOrder* value of 3 will output *nValue* to that channel, an *aDstOrder* value greater than 3 will leave that particular destination channel value unmodified.

Returns:

[Image Data Related Error Codes](#), [ROI Related Error Codes](#)

7.16.1.31 NppStatus nppiSwapChannels_8u_C3IR (Npp8u * pSrcDst, int nSrcDstStep, NppiSize oSizeROI, const int aDstOrder[3])

3 channel 8-bit unsigned integer in place image.

Parameters:

pSrcDst In-Place Image Pointer.

nSrcDstStep In-Place-Image Line Step.

oSizeROI Region-of-Interest (ROI).

aDstOrder Host memory integer array describing how channel values are permuted. The n-th entry of the array contains the number of the channel that is stored in the n-th channel of the output image. E.g. Given an RGB image, aDstOrder = [2,1,0] converts this to BGR channel order.

Returns:

[Image Data Related Error Codes](#), [ROI Related Error Codes](#)

7.16.1.32 NppStatus nppiSwapChannels_8u_C3R (const Npp8u * pSrc, int nSrcStep, Npp8u * pDst, int nDstStep, NppiSize oSizeROI, const int aDstOrder[3])

3 channel 8-bit unsigned integer source image to 3 channel destination image.

Parameters:

pSrc Source-Image Pointer.

nSrcStep Source-Image Line Step.

pDst Destination-Image Pointer.

nDstStep Destination-Image Line Step.

oSizeROI Region-of-Interest (ROI).

aDstOrder Host memory integer array describing how channel values are permuted. The n-th entry of the array contains the number of the channel that is stored in the n-th channel of the output image. E.g. Given an RGB image, aDstOrder = [2,1,0] converts this to BGR channel order.

Returns:

[Image Data Related Error Codes](#), [ROI Related Error Codes](#)

7.16.1.33 NppStatus nppiSwapChannels_8u_C4C3R (const Npp8u * pSrc, int nSrcStep, Npp8u * pDst, int nDstStep, NppiSize oSizeROI, const int aDstOrder[3])

4 channel 8-bit unsigned integer source image to 3 channel destination image.

Parameters:

pSrc Source-Image Pointer.

nSrcStep Source-Image Line Step.

pDst Destination-Image Pointer.

nDstStep Destination-Image Line Step.

oSizeROI [Region-of-Interest \(ROI\)](#).

aDstOrder Host memory integer array describing how channel values are permuted. The n-th entry of the array contains the number of the channel that is stored in the n-th channel of the output image. E.g. Given an RGBA image, *aDstOrder* = [2,1,0] converts this to a 3 channel BGR channel order.

Returns:

[Image Data Related Error Codes](#), [ROI Related Error Codes](#)

7.16.1.34 `NppStatus nppiSwapChannels_8u_C4IR (Npp8u * pSrcDst, int nSrcDstStep, NppiSize oSizeROI, const int aDstOrder[4])`

4 channel 8-bit unsigned integer in place image.

Parameters:

pSrcDst [In-Place Image Pointer](#).

nSrcDstStep [In-Place-Image Line Step](#).

oSizeROI [Region-of-Interest \(ROI\)](#).

aDstOrder Host memory integer array describing how channel values are permuted. The n-th entry of the array contains the number of the channel that is stored in the n-th channel of the output image. E.g. Given an ARGB image, *aDstOrder* = [3,2,1,0] converts this to BGRA channel order.

Returns:

[Image Data Related Error Codes](#), [ROI Related Error Codes](#)

7.16.1.35 `NppStatus nppiSwapChannels_8u_C4R (const Npp8u * pSrc, int nSrcStep, Npp8u * pDst, int nDstStep, NppiSize oSizeROI, const int aDstOrder[4])`

4 channel 8-bit unsigned integer source image to 4 channel destination image.

Parameters:

pSrc [Source-Image Pointer](#).

nSrcStep [Source-Image Line Step](#).

pDst [Destination-Image Pointer](#).

nDstStep [Destination-Image Line Step](#).

oSizeROI [Region-of-Interest \(ROI\)](#).

aDstOrder Host memory integer array describing how channel values are permuted. The n-th entry of the array contains the number of the channel that is stored in the n-th channel of the output image. E.g. Given an ARGB image, *aDstOrder* = [3,2,1,0] converts this to BGRA channel order.

Returns:

[Image Data Related Error Codes](#), [ROI Related Error Codes](#)

Chapter 8

Data Structure Documentation

8.1 NPP_ALIGN_16 Struct Reference

Complex Number This struct represents a long long complex number.

```
#include <nppdefs.h>
```

Data Fields

- [Npp64s re](#)
Real part.
- [Npp64s im](#)
Imaginary part.
- [Npp64f re](#)
Real part.
- [Npp64f im](#)
Imaginary part.

8.1.1 Detailed Description

Complex Number This struct represents a long long complex number.

Complex Number This struct represents a double floating-point complex number.

8.1.2 Field Documentation

8.1.2.1 Npp64f NPP_ALIGN_16::im

Imaginary part.

8.1.2.2 Npp64s NPP_ALIGN_16::im

Imaginary part.

8.1.2.3 Npp64f NPP_ALIGN_16::re

Real part.

8.1.2.4 Npp64s NPP_ALIGN_16::re

Real part.

The documentation for this struct was generated from the following file:

- C:/src/sw/rel/gpgpu/toolkit/r8.0/NPP/npp/include/nppdefs.h

8.2 NPP_ALIGN_8 Struct Reference

Complex Number This struct represents an unsigned int complex number.

```
#include <nppdefs.h>
```

Data Fields

- [Npp32u re](#)
Real part.
- [Npp32u im](#)
Imaginary part.
- [Npp32s re](#)
Real part.
- [Npp32s im](#)
Imaginary part.
- [Npp32f re](#)
Real part.
- [Npp32f im](#)
Imaginary part.

8.2.1 Detailed Description

Complex Number This struct represents an unsigned int complex number.

Complex Number This struct represents a single floating-point complex number.

Complex Number This struct represents a signed int complex number.

8.2.2 Field Documentation

8.2.2.1 Npp32f NPP_ALIGN_8::im

Imaginary part.

8.2.2.2 Npp32s NPP_ALIGN_8::im

Imaginary part.

8.2.2.3 Npp32u NPP_ALIGN_8::im

Imaginary part.

8.2.2.4 Npp32f NPP_ALIGN_8::re

Real part.

8.2.2.5 Npp32s NPP_ALIGN_8::re

Real part.

8.2.2.6 Npp32u NPP_ALIGN_8::re

Real part.

The documentation for this struct was generated from the following file:

- C:/src/sw/rel/gpgpu/toolkit/r8.0/NPP/npp/include/nppdefs.h

8.3 NppiHaarBuffer Struct Reference

```
#include <nppdefs.h>
```

Data Fields

- int `haarBufferSize`
size of the buffer
- `Npp32s * haarBuffer`
buffer

8.3.1 Field Documentation

8.3.1.1 `Npp32s* NppiHaarBuffer::haarBuffer`

buffer

8.3.1.2 `int NppiHaarBuffer::haarBufferSize`

size of the buffer

The documentation for this struct was generated from the following file:

- `C:/src/sw/rel/gpgpu/toolkit/r8.0/NPP/npp/include/nppdefs.h`

8.4 NppiHaarClassifier_32f Struct Reference

```
#include <nppdefs.h>
```

Data Fields

- int `numClassifiers`
number of classifiers
- `Npp32s * classifiers`
packed classifier data 40 bytes each
- `size_t classifierStep`
- `NppiSize classifierSize`
- `Npp32s * counterDevice`

8.4.1 Field Documentation

8.4.1.1 `Npp32s* NppiHaarClassifier_32f::classifiers`

packed classifier data 40 bytes each

8.4.1.2 `NppiSize NppiHaarClassifier_32f::classifierSize`

8.4.1.3 `size_t NppiHaarClassifier_32f::classifierStep`

8.4.1.4 `Npp32s* NppiHaarClassifier_32f::counterDevice`

8.4.1.5 `int NppiHaarClassifier_32f::numClassifiers`

number of classifiers

The documentation for this struct was generated from the following file:

- `C:/src/sw/rel/gpgpu/toolkit/r8.0/NPP/npp/include/nppdefs.h`

8.5 NppiPoint Struct Reference

2D Point

```
#include <nppdefs.h>
```

Data Fields

- `int x`
x-coordinate.
- `int y`
y-coordinate.

8.5.1 Detailed Description

2D Point

8.5.2 Field Documentation

8.5.2.1 `int NppiPoint::x`

x-coordinate.

8.5.2.2 `int NppiPoint::y`

y-coordinate.

The documentation for this struct was generated from the following file:

- `C:/src/sw/rel/gpgpu/toolkit/r8.0/NPP/npp/include/nppdefs.h`

8.6 NppiRect Struct Reference

2D Rectangle This struct contains position and size information of a rectangle in two space.

```
#include <nppdefs.h>
```

Data Fields

- `int x`
x-coordinate of upper left corner (lowest memory address).
- `int y`
y-coordinate of upper left corner (lowest memory address).
- `int width`
Rectangle width.
- `int height`
Rectangle height.

8.6.1 Detailed Description

2D Rectangle This struct contains position and size information of a rectangle in two space.

The rectangle's position is usually signified by the coordinate of its upper-left corner.

8.6.2 Field Documentation

8.6.2.1 `int NppiRect::height`

Rectangle height.

8.6.2.2 `int NppiRect::width`

Rectangle width.

8.6.2.3 `int NppiRect::x`

x-coordinate of upper left corner (lowest memory address).

8.6.2.4 `int NppiRect::y`

y-coordinate of upper left corner (lowest memory address).

The documentation for this struct was generated from the following file:

- `C:/src/sw/rel/gpgpu/toolkit/r8.0/NPP/npp/include/nppdefs.h`

8.7 NppiSize Struct Reference

2D Size This struct typically represents the size of a rectangular region in two space.

```
#include <nppdefs.h>
```

Data Fields

- `int width`
Rectangle width.
- `int height`
Rectangle height.

8.7.1 Detailed Description

2D Size This struct typically represents the size of a rectangular region in two space.

8.7.2 Field Documentation

8.7.2.1 `int NppiSize::height`

Rectangle height.

8.7.2.2 `int NppiSize::width`

Rectangle width.

The documentation for this struct was generated from the following file:

- `C:/src/sw/rel/gpgpu/toolkit/r8.0/NPP/npp/include/nppdefs.h`

8.8 NppLibraryVersion Struct Reference

```
#include <nppdefs.h>
```

Data Fields

- int `major`
Major version number.
- int `minor`
Minor version number.
- int `build`
Build number.

8.8.1 Field Documentation

8.8.1.1 int NppLibraryVersion::build

Build number.

This reflects the nightly build this release was made from.

8.8.1.2 int NppLibraryVersion::major

Major version number.

8.8.1.3 int NppLibraryVersion::minor

Minor version number.

The documentation for this struct was generated from the following file:

- `C:/src/sw/rel/gpgpu/toolkit/r8.0/NPP/npp/include/nppdefs.h`

Index

- `__align__`
 - `npp_basic_types`, 48, 49
- Basic NPP Data Types, 46
- build
 - `NppLibraryVersion`, 294
- classifiers
 - `NppiHaarClassifier_32f`, 290
- classifierSize
 - `NppiHaarClassifier_32f`, 290
- classifierStep
 - `NppiHaarClassifier_32f`, 290
- Convert, 144
- Copy, 97
- Copy Constant Border, 203
- Copy Replicate Border, 216
- Copy Sub-Pixel, 241
- Copy Wrap Border, 228
- core_npp
 - `nppGetGpuComputeCapability`, 28
 - `nppGetGpuDeviceProperties`, 28
 - `nppGetGpuName`, 28
 - `nppGetGpuNumSMs`, 28
 - `nppGetLibVersion`, 28
 - `nppGetMaxThreadsPerBlock`, 29
 - `nppGetMaxThreadsPerSM`, 29
 - `nppGetStream`, 29
 - `nppGetStreamMaxThreadsPerSM`, 29
 - `nppGetStreamNumSMs`, 29
 - `nppSetStream`, 29
- counterDevice
 - `NppiHaarClassifier_32f`, 290
- Data Exchange and Initialization, 62
- Duplicate Channel, 252
- haarBuffer
 - `NppiHaarBuffer`, 289
- haarBufferSize
 - `NppiHaarBuffer`, 289
- height
 - `NppiRect`, 292
 - `NppiSize`, 293
- im
 - `NPP_ALIGN_16`, 285
 - `NPP_ALIGN_8`, 287
- image_convert
 - `nppiConvert_16s16u_C1Rs`, 152
 - `nppiConvert_16s32f_AC4R`, 152
 - `nppiConvert_16s32f_C1R`, 153
 - `nppiConvert_16s32f_C3R`, 153
 - `nppiConvert_16s32f_C4R`, 153
 - `nppiConvert_16s32s_AC4R`, 154
 - `nppiConvert_16s32s_C1R`, 154
 - `nppiConvert_16s32s_C3R`, 154
 - `nppiConvert_16s32s_C4R`, 155
 - `nppiConvert_16s32u_C1Rs`, 155
 - `nppiConvert_16s8s_C1RSfs`, 155
 - `nppiConvert_16s8u_AC4R`, 156
 - `nppiConvert_16s8u_C1R`, 156
 - `nppiConvert_16s8u_C3R`, 156
 - `nppiConvert_16s8u_C4R`, 157
 - `nppiConvert_16u16s_C1RSfs`, 157
 - `nppiConvert_16u32f_AC4R`, 157
 - `nppiConvert_16u32f_C1R`, 158
 - `nppiConvert_16u32f_C3R`, 158
 - `nppiConvert_16u32f_C4R`, 158
 - `nppiConvert_16u32s_AC4R`, 159
 - `nppiConvert_16u32s_C1R`, 159
 - `nppiConvert_16u32s_C3R`, 159
 - `nppiConvert_16u32s_C4R`, 160
 - `nppiConvert_16u32u_C1R`, 160
 - `nppiConvert_16u8s_C1RSfs`, 160
 - `nppiConvert_16u8u_AC4R`, 161
 - `nppiConvert_16u8u_C1R`, 161
 - `nppiConvert_16u8u_C3R`, 161
 - `nppiConvert_16u8u_C4R`, 162
 - `nppiConvert_32f16s_AC4R`, 162
 - `nppiConvert_32f16s_C1R`, 162
 - `nppiConvert_32f16s_C1RSfs`, 163
 - `nppiConvert_32f16s_C3R`, 163
 - `nppiConvert_32f16s_C4R`, 164
 - `nppiConvert_32f16u_AC4R`, 164
 - `nppiConvert_32f16u_C1R`, 164
 - `nppiConvert_32f16u_C1RSfs`, 165
 - `nppiConvert_32f16u_C3R`, 165
 - `nppiConvert_32f16u_C4R`, 166
 - `nppiConvert_32f32s_C1RSfs`, 166
 - `nppiConvert_32f32u_C1RSfs`, 166

- nppiConvert_32f8s_AC4R, 167
- nppiConvert_32f8s_C1R, 167
- nppiConvert_32f8s_C1RSfs, 168
- nppiConvert_32f8s_C3R, 168
- nppiConvert_32f8s_C4R, 168
- nppiConvert_32f8u_AC4R, 169
- nppiConvert_32f8u_C1R, 169
- nppiConvert_32f8u_C1RSfs, 169
- nppiConvert_32f8u_C3R, 170
- nppiConvert_32f8u_C4R, 170
- nppiConvert_32s16s_C1RSfs, 171
- nppiConvert_32s16u_C1RSfs, 171
- nppiConvert_32s32f_C1R, 171
- nppiConvert_32s32u_C1Rs, 172
- nppiConvert_32s8s_AC4R, 172
- nppiConvert_32s8s_C1R, 172
- nppiConvert_32s8s_C3R, 173
- nppiConvert_32s8s_C4R, 173
- nppiConvert_32s8u_AC4R, 173
- nppiConvert_32s8u_C1R, 174
- nppiConvert_32s8u_C3R, 174
- nppiConvert_32s8u_C4R, 174
- nppiConvert_32u16s_C1RSfs, 175
- nppiConvert_32u16u_C1RSfs, 175
- nppiConvert_32u32f_C1R, 176
- nppiConvert_32u32s_C1RSfs, 176
- nppiConvert_32u8s_C1RSfs, 176
- nppiConvert_32u8u_C1RSfs, 177
- nppiConvert_8s16s_C1R, 177
- nppiConvert_8s16u_C1Rs, 178
- nppiConvert_8s32f_AC4R, 178
- nppiConvert_8s32f_C1R, 178
- nppiConvert_8s32f_C3R, 179
- nppiConvert_8s32f_C4R, 179
- nppiConvert_8s32s_AC4R, 179
- nppiConvert_8s32s_C1R, 180
- nppiConvert_8s32s_C3R, 180
- nppiConvert_8s32s_C4R, 180
- nppiConvert_8s32u_C1Rs, 181
- nppiConvert_8s8u_C1Rs, 181
- nppiConvert_8u16s_AC4R, 181
- nppiConvert_8u16s_C1R, 182
- nppiConvert_8u16s_C3R, 182
- nppiConvert_8u16s_C4R, 182
- nppiConvert_8u16u_AC4R, 183
- nppiConvert_8u16u_C1R, 183
- nppiConvert_8u16u_C3R, 183
- nppiConvert_8u16u_C4R, 184
- nppiConvert_8u32f_AC4R, 184
- nppiConvert_8u32f_C1R, 184
- nppiConvert_8u32f_C3R, 185
- nppiConvert_8u32f_C4R, 185
- nppiConvert_8u32s_AC4R, 185
- nppiConvert_8u32s_C1R, 186
- nppiConvert_8u32s_C3R, 186
- nppiConvert_8u32s_C4R, 186
- nppiConvert_8u8s_C1RSfs, 187
- image_copy
 - nppiCopy_16s_AC4MR, 106
 - nppiCopy_16s_AC4R, 107
 - nppiCopy_16s_C1C3R, 107
 - nppiCopy_16s_C1C4R, 107
 - nppiCopy_16s_C1MR, 108
 - nppiCopy_16s_C1R, 108
 - nppiCopy_16s_C3C1R, 108
 - nppiCopy_16s_C3CR, 109
 - nppiCopy_16s_C3MR, 109
 - nppiCopy_16s_C3P3R, 109
 - nppiCopy_16s_C3R, 110
 - nppiCopy_16s_C4C1R, 110
 - nppiCopy_16s_C4CR, 110
 - nppiCopy_16s_C4MR, 111
 - nppiCopy_16s_C4P4R, 111
 - nppiCopy_16s_C4R, 111
 - nppiCopy_16s_P3C3R, 112
 - nppiCopy_16s_P4C4R, 112
 - nppiCopy_16sc_AC4R, 112
 - nppiCopy_16sc_C1R, 113
 - nppiCopy_16sc_C2R, 113
 - nppiCopy_16sc_C3R, 113
 - nppiCopy_16sc_C4R, 114
 - nppiCopy_16u_AC4MR, 114
 - nppiCopy_16u_AC4R, 114
 - nppiCopy_16u_C1C3R, 115
 - nppiCopy_16u_C1C4R, 115
 - nppiCopy_16u_C1MR, 115
 - nppiCopy_16u_C1R, 116
 - nppiCopy_16u_C3C1R, 116
 - nppiCopy_16u_C3CR, 116
 - nppiCopy_16u_C3MR, 117
 - nppiCopy_16u_C3P3R, 117
 - nppiCopy_16u_C3R, 117
 - nppiCopy_16u_C4C1R, 118
 - nppiCopy_16u_C4CR, 118
 - nppiCopy_16u_C4MR, 118
 - nppiCopy_16u_C4P4R, 119
 - nppiCopy_16u_C4R, 119
 - nppiCopy_16u_P3C3R, 119
 - nppiCopy_16u_P4C4R, 120
 - nppiCopy_32f_AC4MR, 120
 - nppiCopy_32f_AC4R, 120
 - nppiCopy_32f_C1C3R, 121
 - nppiCopy_32f_C1C4R, 121
 - nppiCopy_32f_C1MR, 121
 - nppiCopy_32f_C1R, 122
 - nppiCopy_32f_C3C1R, 122
 - nppiCopy_32f_C3CR, 122
 - nppiCopy_32f_C3MR, 123

- nppiCopy_32f_C3P3R, 123
- nppiCopy_32f_C3R, 123
- nppiCopy_32f_C4C1R, 124
- nppiCopy_32f_C4CR, 124
- nppiCopy_32f_C4MR, 124
- nppiCopy_32f_C4P4R, 125
- nppiCopy_32f_C4R, 125
- nppiCopy_32f_P3C3R, 125
- nppiCopy_32f_P4C4R, 126
- nppiCopy_32fc_AC4R, 126
- nppiCopy_32fc_C1R, 126
- nppiCopy_32fc_C2R, 127
- nppiCopy_32fc_C3R, 127
- nppiCopy_32fc_C4R, 127
- nppiCopy_32s_AC4MR, 128
- nppiCopy_32s_AC4R, 128
- nppiCopy_32s_C1C3R, 128
- nppiCopy_32s_C1C4R, 129
- nppiCopy_32s_C1MR, 129
- nppiCopy_32s_C1R, 129
- nppiCopy_32s_C3C1R, 130
- nppiCopy_32s_C3CR, 130
- nppiCopy_32s_C3MR, 130
- nppiCopy_32s_C3P3R, 131
- nppiCopy_32s_C3R, 131
- nppiCopy_32s_C4C1R, 131
- nppiCopy_32s_C4CR, 132
- nppiCopy_32s_C4MR, 132
- nppiCopy_32s_C4P4R, 132
- nppiCopy_32s_C4R, 133
- nppiCopy_32s_P3C3R, 133
- nppiCopy_32s_P4C4R, 133
- nppiCopy_32sc_AC4R, 134
- nppiCopy_32sc_C1R, 134
- nppiCopy_32sc_C2R, 134
- nppiCopy_32sc_C3R, 135
- nppiCopy_32sc_C4R, 135
- nppiCopy_8s_AC4R, 135
- nppiCopy_8s_C1R, 136
- nppiCopy_8s_C2R, 136
- nppiCopy_8s_C3R, 136
- nppiCopy_8s_C4R, 137
- nppiCopy_8u_AC4MR, 137
- nppiCopy_8u_AC4R, 137
- nppiCopy_8u_C1C3R, 138
- nppiCopy_8u_C1C4R, 138
- nppiCopy_8u_C1MR, 138
- nppiCopy_8u_C1R, 139
- nppiCopy_8u_C3C1R, 139
- nppiCopy_8u_C3CR, 139
- nppiCopy_8u_C3MR, 140
- nppiCopy_8u_C3P3R, 140
- nppiCopy_8u_C3R, 140
- nppiCopy_8u_C4C1R, 141
- nppiCopy_8u_C4CR, 141
- nppiCopy_8u_C4MR, 141
- nppiCopy_8u_C4P4R, 142
- nppiCopy_8u_C4R, 142
- nppiCopy_8u_P3C3R, 142
- nppiCopy_8u_P4C4R, 143
- image_copy_constant_border
 - nppiCopyConstBorder_16s_AC4R, 205
 - nppiCopyConstBorder_16s_C1R, 205
 - nppiCopyConstBorder_16s_C3R, 206
 - nppiCopyConstBorder_16s_C4R, 206
 - nppiCopyConstBorder_16u_AC4R, 207
 - nppiCopyConstBorder_16u_C1R, 207
 - nppiCopyConstBorder_16u_C3R, 208
 - nppiCopyConstBorder_16u_C4R, 208
 - nppiCopyConstBorder_32f_AC4R, 209
 - nppiCopyConstBorder_32f_C1R, 209
 - nppiCopyConstBorder_32f_C3R, 210
 - nppiCopyConstBorder_32f_C4R, 210
 - nppiCopyConstBorder_32s_AC4R, 211
 - nppiCopyConstBorder_32s_C1R, 211
 - nppiCopyConstBorder_32s_C3R, 212
 - nppiCopyConstBorder_32s_C4R, 212
 - nppiCopyConstBorder_8u_AC4R, 213
 - nppiCopyConstBorder_8u_C1R, 213
 - nppiCopyConstBorder_8u_C3R, 214
 - nppiCopyConstBorder_8u_C4R, 214
- image_copy_replicate_border
 - nppiCopyReplicateBorder_16s_AC4R, 218
 - nppiCopyReplicateBorder_16s_C1R, 218
 - nppiCopyReplicateBorder_16s_C3R, 219
 - nppiCopyReplicateBorder_16s_C4R, 219
 - nppiCopyReplicateBorder_16u_AC4R, 220
 - nppiCopyReplicateBorder_16u_C1R, 220
 - nppiCopyReplicateBorder_16u_C3R, 221
 - nppiCopyReplicateBorder_16u_C4R, 221
 - nppiCopyReplicateBorder_32f_AC4R, 221
 - nppiCopyReplicateBorder_32f_C1R, 222
 - nppiCopyReplicateBorder_32f_C3R, 222
 - nppiCopyReplicateBorder_32f_C4R, 223
 - nppiCopyReplicateBorder_32s_AC4R, 223
 - nppiCopyReplicateBorder_32s_C1R, 224
 - nppiCopyReplicateBorder_32s_C3R, 224
 - nppiCopyReplicateBorder_32s_C4R, 225
 - nppiCopyReplicateBorder_8u_AC4R, 225
 - nppiCopyReplicateBorder_8u_C1R, 226
 - nppiCopyReplicateBorder_8u_C3R, 226
 - nppiCopyReplicateBorder_8u_C4R, 227
- image_copy_sub_pixel
 - nppiCopySubpix_16s_AC4R, 242
 - nppiCopySubpix_16s_C1R, 243
 - nppiCopySubpix_16s_C3R, 243
 - nppiCopySubpix_16s_C4R, 244
 - nppiCopySubpix_16u_AC4R, 244

- nppiCopySubpix_16u_C1R, 244
- nppiCopySubpix_16u_C3R, 245
- nppiCopySubpix_16u_C4R, 245
- nppiCopySubpix_32f_AC4R, 246
- nppiCopySubpix_32f_C1R, 246
- nppiCopySubpix_32f_C3R, 246
- nppiCopySubpix_32f_C4R, 247
- nppiCopySubpix_32s_AC4R, 247
- nppiCopySubpix_32s_C1R, 248
- nppiCopySubpix_32s_C3R, 248
- nppiCopySubpix_32s_C4R, 249
- nppiCopySubpix_8u_AC4R, 249
- nppiCopySubpix_8u_C1R, 249
- nppiCopySubpix_8u_C3R, 250
- nppiCopySubpix_8u_C4R, 250
- image_copy_wrap_border
 - nppiCopyWrapBorder_16s_AC4R, 230
 - nppiCopyWrapBorder_16s_C1R, 230
 - nppiCopyWrapBorder_16s_C3R, 231
 - nppiCopyWrapBorder_16s_C4R, 231
 - nppiCopyWrapBorder_16u_AC4R, 232
 - nppiCopyWrapBorder_16u_C1R, 232
 - nppiCopyWrapBorder_16u_C3R, 233
 - nppiCopyWrapBorder_16u_C4R, 233
 - nppiCopyWrapBorder_32f_AC4R, 234
 - nppiCopyWrapBorder_32f_C1R, 234
 - nppiCopyWrapBorder_32f_C3R, 235
 - nppiCopyWrapBorder_32f_C4R, 235
 - nppiCopyWrapBorder_32s_AC4R, 236
 - nppiCopyWrapBorder_32s_C1R, 236
 - nppiCopyWrapBorder_32s_C3R, 237
 - nppiCopyWrapBorder_32s_C4R, 237
 - nppiCopyWrapBorder_8u_AC4R, 238
 - nppiCopyWrapBorder_8u_C1R, 238
 - nppiCopyWrapBorder_8u_C3R, 239
 - nppiCopyWrapBorder_8u_C4R, 239
- image_duplicate_channel
 - nppiDup_16s_C1AC4R, 253
 - nppiDup_16s_C1C3R, 253
 - nppiDup_16s_C1C4R, 254
 - nppiDup_16u_C1AC4R, 254
 - nppiDup_16u_C1C3R, 254
 - nppiDup_16u_C1C4R, 255
 - nppiDup_32f_C1AC4R, 255
 - nppiDup_32f_C1C3R, 255
 - nppiDup_32f_C1C4R, 256
 - nppiDup_32s_C1AC4R, 256
 - nppiDup_32s_C1C3R, 256
 - nppiDup_32s_C1C4R, 257
 - nppiDup_8u_C1AC4R, 257
 - nppiDup_8u_C1C3R, 257
 - nppiDup_8u_C1C4R, 258
- image_memory_management
 - nppiFree, 52
 - nppiMalloc_16s_C1, 52
 - nppiMalloc_16s_C2, 52
 - nppiMalloc_16s_C4, 53
 - nppiMalloc_16sc_C1, 53
 - nppiMalloc_16sc_C2, 53
 - nppiMalloc_16sc_C3, 54
 - nppiMalloc_16sc_C4, 54
 - nppiMalloc_16u_C1, 54
 - nppiMalloc_16u_C2, 54
 - nppiMalloc_16u_C3, 55
 - nppiMalloc_16u_C4, 55
 - nppiMalloc_32f_C1, 55
 - nppiMalloc_32f_C2, 56
 - nppiMalloc_32f_C3, 56
 - nppiMalloc_32f_C4, 56
 - nppiMalloc_32fc_C1, 56
 - nppiMalloc_32fc_C2, 57
 - nppiMalloc_32fc_C3, 57
 - nppiMalloc_32fc_C4, 57
 - nppiMalloc_32s_C1, 58
 - nppiMalloc_32s_C3, 58
 - nppiMalloc_32s_C4, 58
 - nppiMalloc_32sc_C1, 58
 - nppiMalloc_32sc_C2, 59
 - nppiMalloc_32sc_C3, 59
 - nppiMalloc_32sc_C4, 59
 - nppiMalloc_8u_C1, 60
 - nppiMalloc_8u_C2, 60
 - nppiMalloc_8u_C3, 60
 - nppiMalloc_8u_C4, 60
- image_scale
 - nppiScale_16s8u_AC4R, 191
 - nppiScale_16s8u_C1R, 191
 - nppiScale_16s8u_C3R, 191
 - nppiScale_16s8u_C4R, 192
 - nppiScale_16u8u_AC4R, 192
 - nppiScale_16u8u_C1R, 192
 - nppiScale_16u8u_C3R, 193
 - nppiScale_16u8u_C4R, 193
 - nppiScale_32f8u_AC4R, 193
 - nppiScale_32f8u_C1R, 194
 - nppiScale_32f8u_C3R, 194
 - nppiScale_32f8u_C4R, 195
 - nppiScale_32s8u_AC4R, 195
 - nppiScale_32s8u_C1R, 195
 - nppiScale_32s8u_C3R, 196
 - nppiScale_32s8u_C4R, 196
 - nppiScale_8u16s_AC4R, 196
 - nppiScale_8u16s_C1R, 197
 - nppiScale_8u16s_C3R, 197
 - nppiScale_8u16s_C4R, 197
 - nppiScale_8u16u_AC4R, 198
 - nppiScale_8u16u_C1R, 198
 - nppiScale_8u16u_C3R, 198

- nppiScale_8u16u_C4R, [199](#)
- nppiScale_8u32f_AC4R, [199](#)
- nppiScale_8u32f_C1R, [199](#)
- nppiScale_8u32f_C3R, [200](#)
- nppiScale_8u32f_C4R, [200](#)
- nppiScale_8u32s_AC4R, [201](#)
- nppiScale_8u32s_C1R, [201](#)
- nppiScale_8u32s_C3R, [201](#)
- nppiScale_8u32s_C4R, [202](#)
- image_set
 - nppiSet_16s_AC4MR, [69](#)
 - nppiSet_16s_AC4R, [70](#)
 - nppiSet_16s_C1MR, [70](#)
 - nppiSet_16s_C1R, [70](#)
 - nppiSet_16s_C2R, [71](#)
 - nppiSet_16s_C3CR, [71](#)
 - nppiSet_16s_C3MR, [71](#)
 - nppiSet_16s_C3R, [72](#)
 - nppiSet_16s_C4CR, [72](#)
 - nppiSet_16s_C4MR, [72](#)
 - nppiSet_16s_C4R, [73](#)
 - nppiSet_16sc_AC4R, [73](#)
 - nppiSet_16sc_C1R, [73](#)
 - nppiSet_16sc_C2R, [74](#)
 - nppiSet_16sc_C3R, [74](#)
 - nppiSet_16sc_C4R, [74](#)
 - nppiSet_16u_AC4MR, [75](#)
 - nppiSet_16u_AC4R, [75](#)
 - nppiSet_16u_C1MR, [75](#)
 - nppiSet_16u_C1R, [76](#)
 - nppiSet_16u_C2R, [76](#)
 - nppiSet_16u_C3CR, [76](#)
 - nppiSet_16u_C3MR, [77](#)
 - nppiSet_16u_C3R, [77](#)
 - nppiSet_16u_C4CR, [77](#)
 - nppiSet_16u_C4MR, [78](#)
 - nppiSet_16u_C4R, [78](#)
 - nppiSet_32f_AC4MR, [78](#)
 - nppiSet_32f_AC4R, [79](#)
 - nppiSet_32f_C1MR, [79](#)
 - nppiSet_32f_C1R, [79](#)
 - nppiSet_32f_C2R, [80](#)
 - nppiSet_32f_C3CR, [80](#)
 - nppiSet_32f_C3MR, [80](#)
 - nppiSet_32f_C3R, [81](#)
 - nppiSet_32f_C4CR, [81](#)
 - nppiSet_32f_C4MR, [81](#)
 - nppiSet_32f_C4R, [82](#)
 - nppiSet_32fc_AC4R, [82](#)
 - nppiSet_32fc_C1R, [82](#)
 - nppiSet_32fc_C2R, [83](#)
 - nppiSet_32fc_C3R, [83](#)
 - nppiSet_32fc_C4R, [83](#)
 - nppiSet_32s_AC4MR, [84](#)
 - nppiSet_32s_AC4R, [84](#)
 - nppiSet_32s_C1MR, [84](#)
 - nppiSet_32s_C1R, [85](#)
 - nppiSet_32s_C2R, [85](#)
 - nppiSet_32s_C3CR, [85](#)
 - nppiSet_32s_C3MR, [86](#)
 - nppiSet_32s_C3R, [86](#)
 - nppiSet_32s_C4CR, [86](#)
 - nppiSet_32s_C4MR, [87](#)
 - nppiSet_32s_C4R, [87](#)
 - nppiSet_32sc_AC4R, [87](#)
 - nppiSet_32sc_C1R, [88](#)
 - nppiSet_32sc_C2R, [88](#)
 - nppiSet_32sc_C3R, [88](#)
 - nppiSet_32sc_C4R, [89](#)
 - nppiSet_32u_AC4R, [89](#)
 - nppiSet_32u_C1R, [89](#)
 - nppiSet_32u_C2R, [90](#)
 - nppiSet_32u_C3R, [90](#)
 - nppiSet_32u_C4R, [90](#)
 - nppiSet_8s_AC4R, [91](#)
 - nppiSet_8s_C1R, [91](#)
 - nppiSet_8s_C2R, [91](#)
 - nppiSet_8s_C3R, [92](#)
 - nppiSet_8s_C4R, [92](#)
 - nppiSet_8u_AC4MR, [92](#)
 - nppiSet_8u_AC4R, [93](#)
 - nppiSet_8u_C1MR, [93](#)
 - nppiSet_8u_C1R, [93](#)
 - nppiSet_8u_C2R, [94](#)
 - nppiSet_8u_C3CR, [94](#)
 - nppiSet_8u_C3MR, [94](#)
 - nppiSet_8u_C3R, [95](#)
 - nppiSet_8u_C4CR, [95](#)
 - nppiSet_8u_C4MR, [95](#)
 - nppiSet_8u_C4R, [96](#)
- image_swap_channels
 - nppiSwapChannels_16s_AC4R, [269](#)
 - nppiSwapChannels_16s_C3C4R, [269](#)
 - nppiSwapChannels_16s_C3IR, [269](#)
 - nppiSwapChannels_16s_C3R, [270](#)
 - nppiSwapChannels_16s_C4C3R, [270](#)
 - nppiSwapChannels_16s_C4IR, [271](#)
 - nppiSwapChannels_16s_C4R, [271](#)
 - nppiSwapChannels_16u_AC4R, [271](#)
 - nppiSwapChannels_16u_C3C4R, [272](#)
 - nppiSwapChannels_16u_C3IR, [272](#)
 - nppiSwapChannels_16u_C3R, [273](#)
 - nppiSwapChannels_16u_C4C3R, [273](#)
 - nppiSwapChannels_16u_C4IR, [274](#)
 - nppiSwapChannels_16u_C4R, [274](#)
 - nppiSwapChannels_32f_AC4R, [274](#)
 - nppiSwapChannels_32f_C3C4R, [275](#)
 - nppiSwapChannels_32f_C3IR, [275](#)

- nppiSwapChannels_32f_C3R, 276
- nppiSwapChannels_32f_C4C3R, 276
- nppiSwapChannels_32f_C4IR, 277
- nppiSwapChannels_32f_C4R, 277
- nppiSwapChannels_32s_AC4R, 277
- nppiSwapChannels_32s_C3C4R, 278
- nppiSwapChannels_32s_C3IR, 278
- nppiSwapChannels_32s_C3R, 279
- nppiSwapChannels_32s_C4C3R, 279
- nppiSwapChannels_32s_C4IR, 280
- nppiSwapChannels_32s_C4R, 280
- nppiSwapChannels_8u_AC4R, 280
- nppiSwapChannels_8u_C3C4R, 281
- nppiSwapChannels_8u_C3IR, 281
- nppiSwapChannels_8u_C3R, 282
- nppiSwapChannels_8u_C4C3R, 282
- nppiSwapChannels_8u_C4IR, 283
- nppiSwapChannels_8u_C4R, 283
- image_transpose
 - nppiTranspose_16s_C1R, 260
 - nppiTranspose_16s_C3R, 260
 - nppiTranspose_16s_C4R, 261
 - nppiTranspose_16u_C1R, 261
 - nppiTranspose_16u_C3R, 261
 - nppiTranspose_16u_C4R, 262
 - nppiTranspose_32f_C1R, 262
 - nppiTranspose_32f_C3R, 262
 - nppiTranspose_32f_C4R, 263
 - nppiTranspose_32s_C1R, 263
 - nppiTranspose_32s_C3R, 263
 - nppiTranspose_32s_C4R, 264
 - nppiTranspose_8u_C1R, 264
 - nppiTranspose_8u_C3R, 264
 - nppiTranspose_8u_C4R, 265
- major
 - NppLibraryVersion, 294
- Memory Management, 50
- minor
 - NppLibraryVersion, 294
- NPP Core, 27
- NPP Type Definitions and Constants, 31
- Npp16s
 - npp_basic_types, 47
- Npp16sc
 - npp_basic_types, 49
- Npp16u
 - npp_basic_types, 47
- Npp16uc
 - npp_basic_types, 49
- Npp32f
 - npp_basic_types, 47
- Npp32fc
 - npp_basic_types, 47
- Npp32s
 - npp_basic_types, 47
- Npp32sc
 - npp_basic_types, 47
- Npp32u
 - npp_basic_types, 48
- Npp32uc
 - npp_basic_types, 48
- Npp64f
 - npp_basic_types, 48
- Npp64fc
 - npp_basic_types, 48
- Npp64s
 - npp_basic_types, 48
- Npp64sc
 - npp_basic_types, 48
- Npp64u
 - npp_basic_types, 48
- Npp8s
 - npp_basic_types, 48
- Npp8u
 - npp_basic_types, 48
- Npp8uc
 - npp_basic_types, 49
- NPP_AFFINE_QUAD_INCORRECT_WARNING
 - typedefs_npp, 44
- NPP_ALG_HINT_ACCURATE
 - typedefs_npp, 39
- NPP_ALG_HINT_FAST
 - typedefs_npp, 39
- NPP_ALG_HINT_NONE
 - typedefs_npp, 39
- NPP_ALIGNMENT_ERROR
 - typedefs_npp, 43
- NPP_ANCHOR_ERROR
 - typedefs_npp, 43
- NPP_BAD_ARGUMENT_ERROR
 - typedefs_npp, 44
- NPP_BORDER_CONSTANT
 - typedefs_npp, 40
- NPP_BORDER_MIRROR
 - typedefs_npp, 40
- NPP_BORDER_NONE
 - typedefs_npp, 40
- NPP_BORDER_REPLICATE
 - typedefs_npp, 40
- NPP_BORDER_UNDEFINED
 - typedefs_npp, 40
- NPP_BORDER_WRAP
 - typedefs_npp, 40
- NPP_BOTH_AXIS
 - typedefs_npp, 40
- NPP_CHANNEL_ERROR

- typedefs_npp, 43
- NPP_CHANNEL_ORDER_ERROR
 - typedefs_npp, 43
- NPP_CMP_EQ
 - typedefs_npp, 39
- NPP_CMP_GREATER
 - typedefs_npp, 39
- NPP_CMP_GREATER_EQ
 - typedefs_npp, 39
- NPP_CMP_LESS
 - typedefs_npp, 38
- NPP_CMP_LESS_EQ
 - typedefs_npp, 38
- NPP_COEFFICIENT_ERROR
 - typedefs_npp, 43
- NPP_COI_ERROR
 - typedefs_npp, 43
- NPP_CONTEXT_MATCH_ERROR
 - typedefs_npp, 44
- NPP_CORRUPTED_DATA_ERROR
 - typedefs_npp, 43
- NPP_CUDA_1_0
 - typedefs_npp, 39
- NPP_CUDA_1_1
 - typedefs_npp, 39
- NPP_CUDA_1_2
 - typedefs_npp, 39
- NPP_CUDA_1_3
 - typedefs_npp, 39
- NPP_CUDA_2_0
 - typedefs_npp, 39
- NPP_CUDA_2_1
 - typedefs_npp, 39
- NPP_CUDA_3_0
 - typedefs_npp, 39
- NPP_CUDA_3_2
 - typedefs_npp, 39
- NPP_CUDA_3_5
 - typedefs_npp, 39
- NPP_CUDA_3_7
 - typedefs_npp, 39
- NPP_CUDA_5_0
 - typedefs_npp, 39
- NPP_CUDA_5_2
 - typedefs_npp, 39
- NPP_CUDA_5_3
 - typedefs_npp, 39
- NPP_CUDA_6_0
 - typedefs_npp, 39
- NPP_CUDA_KERNEL_EXECUTION_ERROR
 - typedefs_npp, 43
- NPP_CUDA_NOT_CAPABLE
 - typedefs_npp, 39
- NPP_CUDA_UNKNOWN_VERSION
 - typedefs_npp, 39
- NPP_DATA_TYPE_ERROR
 - typedefs_npp, 44
- NPP_DIVIDE_BY_ZERO_ERROR
 - typedefs_npp, 44
- NPP_DIVIDE_BY_ZERO_WARNING
 - typedefs_npp, 44
- NPP_DIVISOR_ERROR
 - typedefs_npp, 43
- NPP_DOUBLE_SIZE_WARNING
 - typedefs_npp, 44
- NPP_ERROR
 - typedefs_npp, 44
- NPP_ERROR_RESERVED
 - typedefs_npp, 44
- NPP_FFT_FLAG_ERROR
 - typedefs_npp, 44
- NPP_FFT_ORDER_ERROR
 - typedefs_npp, 44
- NPP_FILTER_SCHARR
 - typedefs_npp, 40
- NPP_FILTER_SOBEL
 - typedefs_npp, 40
- NPP_HAAR_CLASSIFIER_PIXEL_MATCH_ERROR
 - typedefs_npp, 43
- NPP_HISTOGRAM_NUMBER_OF_LEVELS_ERROR
 - typedefs_npp, 43
- NPP_HORIZONTAL_AXIS
 - typedefs_npp, 40
- NPP_INTERPOLATION_ERROR
 - typedefs_npp, 44
- NPP_INVALID_DEVICE_POINTER_ERROR
 - typedefs_npp, 43
- NPP_INVALID_HOST_POINTER_ERROR
 - typedefs_npp, 43
- NPP_LUT_NUMBER_OF_LEVELS_ERROR
 - typedefs_npp, 43
- NPP_LUT_PALETTE_BITSIZE_ERROR
 - typedefs_npp, 43
- NPP_MASK_SIZE_11_X_11
 - typedefs_npp, 41
- NPP_MASK_SIZE_13_X_13
 - typedefs_npp, 41
- NPP_MASK_SIZE_15_X_15
 - typedefs_npp, 41
- NPP_MASK_SIZE_1_X_3
 - typedefs_npp, 41
- NPP_MASK_SIZE_1_X_5
 - typedefs_npp, 41
- NPP_MASK_SIZE_3_X_1
 - typedefs_npp, 41
- NPP_MASK_SIZE_3_X_3
 - typedefs_npp, 41

- typedefs_npp, 41
- NPP_MASK_SIZE_5_X_1
 - typedefs_npp, 41
- NPP_MASK_SIZE_5_X_5
 - typedefs_npp, 41
- NPP_MASK_SIZE_7_X_7
 - typedefs_npp, 41
- NPP_MASK_SIZE_9_X_9
 - typedefs_npp, 41
- NPP_MASK_SIZE_ERROR
 - typedefs_npp, 43
- NPP_MEMCPY_ERROR
 - typedefs_npp, 43
- NPP_MEMFREE_ERROR
 - typedefs_npp, 43
- NPP_MEMORY_ALLOCATION_ERR
 - typedefs_npp, 44
- NPP_MEMSET_ERROR
 - typedefs_npp, 43
- NPP_MIRROR_FLIP_ERROR
 - typedefs_npp, 44
- NPP_MISALIGNED_DST_ROI_WARNING
 - typedefs_npp, 44
- NPP_MOMENT_00_ZERO_ERROR
 - typedefs_npp, 44
- NPP_NO_ERROR
 - typedefs_npp, 44
- NPP_NO_MEMORY_ERROR
 - typedefs_npp, 44
- NPP_NO_OPERATION_WARNING
 - typedefs_npp, 44
- NPP_NOT_EVEN_STEP_ERROR
 - typedefs_npp, 43
- NPP_NOT_IMPLEMENTED_ERROR
 - typedefs_npp, 44
- NPP_NOT_SUFFICIENT_COMPUTE_-
 - CAPABILITY
 - typedefs_npp, 43
- NPP_NOT_SUPPORTED_MODE_ERROR
 - typedefs_npp, 43
- NPP_NULL_POINTER_ERROR
 - typedefs_npp, 44
- NPP_NUMBER_OF_CHANNELS_ERROR
 - typedefs_npp, 43
- NPP_OUT_OFF_RANGE_ERROR
 - typedefs_npp, 44
- NPP_OVERFLOW_ERROR
 - typedefs_npp, 43
- NPP_QUADRANGLE_ERROR
 - typedefs_npp, 43
- NPP_QUALITY_INDEX_ERROR
 - typedefs_npp, 43
- NPP_RANGE_ERROR
 - typedefs_npp, 44
- NPP_RECTANGLE_ERROR
 - typedefs_npp, 43
- NPP_RESIZE_FACTOR_ERROR
 - typedefs_npp, 44
- NPP_RESIZE_NO_OPERATION_ERROR
 - typedefs_npp, 43
- NPP_RND_FINANCIAL
 - typedefs_npp, 42
- NPP_RND_NEAR
 - typedefs_npp, 42
- NPP_RND_ZERO
 - typedefs_npp, 42
- NPP_ROUND_MODE_NOT_SUPPORTED_-
 - ERROR
 - typedefs_npp, 43
- NPP_ROUND_NEAREST_TIES_AWAY_-
 - FROM_ZERO
 - typedefs_npp, 42
- NPP_ROUND_NEAREST_TIES_TO_EVEN
 - typedefs_npp, 42
- NPP_ROUND_TOWARD_ZERO
 - typedefs_npp, 42
- NPP_SCALE_RANGE_ERROR
 - typedefs_npp, 44
- NPP_SIZE_ERROR
 - typedefs_npp, 44
- NPP_STEP_ERROR
 - typedefs_npp, 44
- NPP_STRIDE_ERROR
 - typedefs_npp, 43
- NPP_SUCCESS
 - typedefs_npp, 44
- NPP_TEXTURE_BIND_ERROR
 - typedefs_npp, 43
- NPP_THRESHOLD_ERROR
 - typedefs_npp, 44
- NPP_THRESHOLD_NEGATIVE_LEVEL_-
 - ERROR
 - typedefs_npp, 44
- NPP_VERTICAL_AXIS
 - typedefs_npp, 40
- NPP_WRONG_INTERSECTION_QUAD_-
 - WARNING
 - typedefs_npp, 44
- NPP_WRONG_INTERSECTION_ROI_ERROR
 - typedefs_npp, 43
- NPP_WRONG_INTERSECTION_ROI_-
 - WARNING
 - typedefs_npp, 44
- NPP_ZC_MODE_NOT_SUPPORTED_ERROR
 - typedefs_npp, 43
- NPP_ZERO_MASK_VALUE_ERROR
 - typedefs_npp, 43
- NPP_ALIGN_16, 285

- im, 285
- re, 286
- NPP_ALIGN_8, 287
 - im, 287
 - re, 287, 288
- npp_basic_types
 - __align__, 48, 49
 - Npp16s, 47
 - Npp16sc, 49
 - Npp16u, 47
 - Npp16uc, 49
 - Npp32f, 47
 - Npp32fc, 47
 - Npp32s, 47
 - Npp32sc, 47
 - Npp32u, 48
 - Npp32uc, 48
 - Npp64f, 48
 - Npp64fc, 48
 - Npp64s, 48
 - Npp64sc, 48
 - Npp64u, 48
 - Npp8s, 48
 - Npp8u, 48
 - Npp8uc, 49
- NPP_MAX_16S
 - typedefs_npp, 37
- NPP_MAX_16U
 - typedefs_npp, 37
- NPP_MAX_32S
 - typedefs_npp, 37
- NPP_MAX_32U
 - typedefs_npp, 37
- NPP_MAX_64S
 - typedefs_npp, 37
- NPP_MAX_64U
 - typedefs_npp, 37
- NPP_MAX_8S
 - typedefs_npp, 37
- NPP_MAX_8U
 - typedefs_npp, 37
- NPP_MAXABS_32F
 - typedefs_npp, 37
- NPP_MAXABS_64F
 - typedefs_npp, 37
- NPP_MIN_16S
 - typedefs_npp, 37
- NPP_MIN_16U
 - typedefs_npp, 38
- NPP_MIN_32S
 - typedefs_npp, 38
- NPP_MIN_32U
 - typedefs_npp, 38
- NPP_MIN_64S
 - typedefs_npp, 38
- NPP_MIN_64U
 - typedefs_npp, 38
- NPP_MIN_8S
 - typedefs_npp, 38
- NPP_MIN_8U
 - typedefs_npp, 38
- NPP_MINABS_32F
 - typedefs_npp, 38
- NPP_MINABS_64F
 - typedefs_npp, 38
- NppCmpOp
 - typedefs_npp, 38
- nppGetGpuComputeCapability
 - core_npp, 28
- nppGetGpuDeviceProperties
 - core_npp, 28
- nppGetGpuName
 - core_npp, 28
- nppGetGpuNumSMs
 - core_npp, 28
- nppGetLibVersion
 - core_npp, 28
- nppGetMaxThreadsPerBlock
 - core_npp, 29
- nppGetMaxThreadsPerSM
 - core_npp, 29
- nppGetStream
 - core_npp, 29
- nppGetStreamMaxThreadsPerSM
 - core_npp, 29
- nppGetStreamNumSMs
 - core_npp, 29
- NppGpuComputeCapability
 - typedefs_npp, 39
- NppHintAlgorithm
 - typedefs_npp, 39
- NPPI_BAYER_BGGR
 - typedefs_npp, 40
- NPPI_BAYER_GBRG
 - typedefs_npp, 40
- NPPI_BAYER_GRBG
 - typedefs_npp, 40
- NPPI_BAYER_RGGB
 - typedefs_npp, 40
- NPPI_INTER_CUBIC
 - typedefs_npp, 41
- NPPI_INTER_CUBIC2P_B05C03
 - typedefs_npp, 41
- NPPI_INTER_CUBIC2P_BSPLINE
 - typedefs_npp, 41
- NPPI_INTER_CUBIC2P_CATMULLROM
 - typedefs_npp, 41
- NPPI_INTER_LANCZOS
 - typedefs_npp, 41

- typedefs_npp, 41
- NPPI_INTER_LANCZOS3_ADVANCED
 - typedefs_npp, 41
- NPPI_INTER_LINEAR
 - typedefs_npp, 41
- NPPI_INTER_NN
 - typedefs_npp, 41
- NPPI_INTER_SUPER
 - typedefs_npp, 41
- NPPI_INTER_UNDEFINED
 - typedefs_npp, 41
- NPPI_OP_ALPHA_ATOP
 - typedefs_npp, 39
- NPPI_OP_ALPHA_ATOP_PREMUL
 - typedefs_npp, 40
- NPPI_OP_ALPHA_IN
 - typedefs_npp, 39
- NPPI_OP_ALPHA_IN_PREMUL
 - typedefs_npp, 40
- NPPI_OP_ALPHA_OUT
 - typedefs_npp, 39
- NPPI_OP_ALPHA_OUT_PREMUL
 - typedefs_npp, 40
- NPPI_OP_ALPHA_OVER
 - typedefs_npp, 39
- NPPI_OP_ALPHA_OVER_PREMUL
 - typedefs_npp, 40
- NPPI_OP_ALPHA_PLUS
 - typedefs_npp, 39
- NPPI_OP_ALPHA_PLUS_PREMUL
 - typedefs_npp, 40
- NPPI_OP_ALPHA_PREMUL
 - typedefs_npp, 40
- NPPI_OP_ALPHA_XOR
 - typedefs_npp, 39
- NPPI_OP_ALPHA_XOR_PREMUL
 - typedefs_npp, 40
- NPPI_SMOOTH_EDGE
 - typedefs_npp, 41
- nppiACTable
 - typedefs_npp, 41
- NppiAlphaOp
 - typedefs_npp, 39
- NppiAxis
 - typedefs_npp, 40
- NppiBayerGridPosition
 - typedefs_npp, 40
- NppiBorderType
 - typedefs_npp, 40
- nppiConvert_16s16u_C1Rs
 - image_convert, 152
- nppiConvert_16s32f_AC4R
 - image_convert, 152
- nppiConvert_16s32f_C1R
 - image_convert, 153
- nppiConvert_16s32f_C3R
 - image_convert, 153
- nppiConvert_16s32f_C4R
 - image_convert, 153
- nppiConvert_16s32s_AC4R
 - image_convert, 154
- nppiConvert_16s32s_C1R
 - image_convert, 154
- nppiConvert_16s32s_C3R
 - image_convert, 154
- nppiConvert_16s32s_C4R
 - image_convert, 155
- nppiConvert_16s32u_C1Rs
 - image_convert, 155
- nppiConvert_16s8s_C1RSfs
 - image_convert, 155
- nppiConvert_16s8u_AC4R
 - image_convert, 156
- nppiConvert_16s8u_C1R
 - image_convert, 156
- nppiConvert_16s8u_C3R
 - image_convert, 156
- nppiConvert_16s8u_C4R
 - image_convert, 157
- nppiConvert_16u16s_C1RSfs
 - image_convert, 157
- nppiConvert_16u32f_AC4R
 - image_convert, 157
- nppiConvert_16u32f_C1R
 - image_convert, 158
- nppiConvert_16u32f_C3R
 - image_convert, 158
- nppiConvert_16u32f_C4R
 - image_convert, 158
- nppiConvert_16u32s_AC4R
 - image_convert, 159
- nppiConvert_16u32s_C1R
 - image_convert, 159
- nppiConvert_16u32s_C3R
 - image_convert, 159
- nppiConvert_16u32s_C4R
 - image_convert, 160
- nppiConvert_16u32u_C1R
 - image_convert, 160
- nppiConvert_16u8s_C1RSfs
 - image_convert, 160
- nppiConvert_16u8u_AC4R
 - image_convert, 161
- nppiConvert_16u8u_C1R
 - image_convert, 161
- nppiConvert_16u8u_C3R
 - image_convert, 161
- nppiConvert_16u8u_C4R
 - image_convert, 161

- image_convert, 162
- nppiConvert_32f16s_AC4R
 - image_convert, 162
- nppiConvert_32f16s_C1R
 - image_convert, 162
- nppiConvert_32f16s_C1RSfs
 - image_convert, 163
- nppiConvert_32f16s_C3R
 - image_convert, 163
- nppiConvert_32f16s_C4R
 - image_convert, 164
- nppiConvert_32f16u_AC4R
 - image_convert, 164
- nppiConvert_32f16u_C1R
 - image_convert, 164
- nppiConvert_32f16u_C1RSfs
 - image_convert, 165
- nppiConvert_32f16u_C3R
 - image_convert, 165
- nppiConvert_32f16u_C4R
 - image_convert, 166
- nppiConvert_32f32s_C1RSfs
 - image_convert, 166
- nppiConvert_32f32u_C1RSfs
 - image_convert, 166
- nppiConvert_32f8s_AC4R
 - image_convert, 167
- nppiConvert_32f8s_C1R
 - image_convert, 167
- nppiConvert_32f8s_C1RSfs
 - image_convert, 168
- nppiConvert_32f8s_C3R
 - image_convert, 168
- nppiConvert_32f8s_C4R
 - image_convert, 168
- nppiConvert_32f8u_AC4R
 - image_convert, 169
- nppiConvert_32f8u_C1R
 - image_convert, 169
- nppiConvert_32f8u_C1RSfs
 - image_convert, 169
- nppiConvert_32f8u_C3R
 - image_convert, 170
- nppiConvert_32f8u_C4R
 - image_convert, 170
- nppiConvert_32s16s_C1RSfs
 - image_convert, 171
- nppiConvert_32s16u_C1RSfs
 - image_convert, 171
- nppiConvert_32s32f_C1R
 - image_convert, 171
- nppiConvert_32s32u_C1Rs
 - image_convert, 172
- nppiConvert_32s8s_AC4R
 - image_convert, 172
- nppiConvert_32s8s_C1R
 - image_convert, 172
- nppiConvert_32s8s_C3R
 - image_convert, 173
- nppiConvert_32s8s_C4R
 - image_convert, 173
- nppiConvert_32s8u_AC4R
 - image_convert, 173
- nppiConvert_32s8u_C1R
 - image_convert, 174
- nppiConvert_32s8u_C3R
 - image_convert, 174
- nppiConvert_32s8u_C4R
 - image_convert, 174
- nppiConvert_32u16s_C1RSfs
 - image_convert, 175
- nppiConvert_32u16u_C1RSfs
 - image_convert, 175
- nppiConvert_32u32f_C1R
 - image_convert, 176
- nppiConvert_32u32s_C1RSfs
 - image_convert, 176
- nppiConvert_32u8s_C1RSfs
 - image_convert, 176
- nppiConvert_32u8u_C1RSfs
 - image_convert, 177
- nppiConvert_8s16s_C1R
 - image_convert, 177
- nppiConvert_8s16u_C1Rs
 - image_convert, 178
- nppiConvert_8s32f_AC4R
 - image_convert, 178
- nppiConvert_8s32f_C1R
 - image_convert, 178
- nppiConvert_8s32f_C3R
 - image_convert, 179
- nppiConvert_8s32f_C4R
 - image_convert, 179
- nppiConvert_8s32s_AC4R
 - image_convert, 179
- nppiConvert_8s32s_C1R
 - image_convert, 180
- nppiConvert_8s32s_C3R
 - image_convert, 180
- nppiConvert_8s32s_C4R
 - image_convert, 180
- nppiConvert_8s32u_C1Rs
 - image_convert, 181
- nppiConvert_8s8u_C1Rs
 - image_convert, 181
- nppiConvert_8u16s_AC4R
 - image_convert, 181
- nppiConvert_8u16s_C1R

- image_convert, 182
- nppiConvert_8u16s_C3R
 - image_convert, 182
- nppiConvert_8u16s_C4R
 - image_convert, 182
- nppiConvert_8u16u_AC4R
 - image_convert, 183
- nppiConvert_8u16u_C1R
 - image_convert, 183
- nppiConvert_8u16u_C3R
 - image_convert, 183
- nppiConvert_8u16u_C4R
 - image_convert, 184
- nppiConvert_8u32f_AC4R
 - image_convert, 184
- nppiConvert_8u32f_C1R
 - image_convert, 184
- nppiConvert_8u32f_C3R
 - image_convert, 185
- nppiConvert_8u32f_C4R
 - image_convert, 185
- nppiConvert_8u32s_AC4R
 - image_convert, 185
- nppiConvert_8u32s_C1R
 - image_convert, 186
- nppiConvert_8u32s_C3R
 - image_convert, 186
- nppiConvert_8u32s_C4R
 - image_convert, 186
- nppiConvert_8u8s_C1RSfs
 - image_convert, 187
- nppiCopy_16s_AC4MR
 - image_copy, 106
- nppiCopy_16s_AC4R
 - image_copy, 107
- nppiCopy_16s_C1C3R
 - image_copy, 107
- nppiCopy_16s_C1C4R
 - image_copy, 107
- nppiCopy_16s_C1MR
 - image_copy, 108
- nppiCopy_16s_C1R
 - image_copy, 108
- nppiCopy_16s_C3C1R
 - image_copy, 108
- nppiCopy_16s_C3CR
 - image_copy, 109
- nppiCopy_16s_C3MR
 - image_copy, 109
- nppiCopy_16s_C3P3R
 - image_copy, 109
- nppiCopy_16s_C3R
 - image_copy, 110
- nppiCopy_16s_C4C1R
 - image_copy, 110
- nppiCopy_16s_C4CR
 - image_copy, 110
- nppiCopy_16s_C4MR
 - image_copy, 111
- nppiCopy_16s_C4P4R
 - image_copy, 111
- nppiCopy_16s_C4R
 - image_copy, 111
- nppiCopy_16s_P3C3R
 - image_copy, 112
- nppiCopy_16s_P4C4R
 - image_copy, 112
- nppiCopy_16sc_AC4R
 - image_copy, 112
- nppiCopy_16sc_C1R
 - image_copy, 113
- nppiCopy_16sc_C2R
 - image_copy, 113
- nppiCopy_16sc_C3R
 - image_copy, 113
- nppiCopy_16sc_C4R
 - image_copy, 114
- nppiCopy_16u_AC4MR
 - image_copy, 114
- nppiCopy_16u_AC4R
 - image_copy, 114
- nppiCopy_16u_C1C3R
 - image_copy, 115
- nppiCopy_16u_C1C4R
 - image_copy, 115
- nppiCopy_16u_C1MR
 - image_copy, 115
- nppiCopy_16u_C1R
 - image_copy, 116
- nppiCopy_16u_C3C1R
 - image_copy, 116
- nppiCopy_16u_C3CR
 - image_copy, 116
- nppiCopy_16u_C3MR
 - image_copy, 117
- nppiCopy_16u_C3P3R
 - image_copy, 117
- nppiCopy_16u_C3R
 - image_copy, 117
- nppiCopy_16u_C4C1R
 - image_copy, 118
- nppiCopy_16u_C4CR
 - image_copy, 118
- nppiCopy_16u_C4MR
 - image_copy, 118
- nppiCopy_16u_C4P4R
 - image_copy, 119
- nppiCopy_16u_C4R

image_copy, 119
nppiCopy_16u_P3C3R
image_copy, 119
nppiCopy_16u_P4C4R
image_copy, 120
nppiCopy_32f_AC4MR
image_copy, 120
nppiCopy_32f_AC4R
image_copy, 120
nppiCopy_32f_C1C3R
image_copy, 121
nppiCopy_32f_C1C4R
image_copy, 121
nppiCopy_32f_C1MR
image_copy, 121
nppiCopy_32f_C1R
image_copy, 122
nppiCopy_32f_C3C1R
image_copy, 122
nppiCopy_32f_C3CR
image_copy, 122
nppiCopy_32f_C3MR
image_copy, 123
nppiCopy_32f_C3P3R
image_copy, 123
nppiCopy_32f_C3R
image_copy, 123
nppiCopy_32f_C4C1R
image_copy, 124
nppiCopy_32f_C4CR
image_copy, 124
nppiCopy_32f_C4MR
image_copy, 124
nppiCopy_32f_C4P4R
image_copy, 125
nppiCopy_32f_C4R
image_copy, 125
nppiCopy_32f_P3C3R
image_copy, 125
nppiCopy_32f_P4C4R
image_copy, 126
nppiCopy_32fc_AC4R
image_copy, 126
nppiCopy_32fc_C1R
image_copy, 126
nppiCopy_32fc_C2R
image_copy, 127
nppiCopy_32fc_C3R
image_copy, 127
nppiCopy_32fc_C4R
image_copy, 127
nppiCopy_32s_AC4MR
image_copy, 128
nppiCopy_32s_AC4R
image_copy, 128
nppiCopy_32s_C1C3R
image_copy, 128
nppiCopy_32s_C1C4R
image_copy, 129
nppiCopy_32s_C1MR
image_copy, 129
nppiCopy_32s_C1R
image_copy, 129
nppiCopy_32s_C3C1R
image_copy, 130
nppiCopy_32s_C3CR
image_copy, 130
nppiCopy_32s_C3MR
image_copy, 130
nppiCopy_32s_C3P3R
image_copy, 131
nppiCopy_32s_C3R
image_copy, 131
nppiCopy_32s_C4C1R
image_copy, 131
nppiCopy_32s_C4CR
image_copy, 132
nppiCopy_32s_C4MR
image_copy, 132
nppiCopy_32s_C4P4R
image_copy, 132
nppiCopy_32s_C4R
image_copy, 133
nppiCopy_32s_P3C3R
image_copy, 133
nppiCopy_32s_P4C4R
image_copy, 133
nppiCopy_32sc_AC4R
image_copy, 134
nppiCopy_32sc_C1R
image_copy, 134
nppiCopy_32sc_C2R
image_copy, 134
nppiCopy_32sc_C3R
image_copy, 135
nppiCopy_32sc_C4R
image_copy, 135
nppiCopy_8s_AC4R
image_copy, 135
nppiCopy_8s_C1R
image_copy, 136
nppiCopy_8s_C2R
image_copy, 136
nppiCopy_8s_C3R
image_copy, 136
nppiCopy_8s_C4R
image_copy, 137
nppiCopy_8u_AC4MR

- image_copy, 137
- nppiCopy_8u_AC4R
 - image_copy, 137
- nppiCopy_8u_C1C3R
 - image_copy, 138
- nppiCopy_8u_C1C4R
 - image_copy, 138
- nppiCopy_8u_C1MR
 - image_copy, 138
- nppiCopy_8u_C1R
 - image_copy, 139
- nppiCopy_8u_C3C1R
 - image_copy, 139
- nppiCopy_8u_C3CR
 - image_copy, 139
- nppiCopy_8u_C3MR
 - image_copy, 140
- nppiCopy_8u_C3P3R
 - image_copy, 140
- nppiCopy_8u_C3R
 - image_copy, 140
- nppiCopy_8u_C4C1R
 - image_copy, 141
- nppiCopy_8u_C4CR
 - image_copy, 141
- nppiCopy_8u_C4MR
 - image_copy, 141
- nppiCopy_8u_C4P4R
 - image_copy, 142
- nppiCopy_8u_C4R
 - image_copy, 142
- nppiCopy_8u_P3C3R
 - image_copy, 142
- nppiCopy_8u_P4C4R
 - image_copy, 143
- nppiCopyConstBorder_16s_AC4R
 - image_copy_constant_border, 205
- nppiCopyConstBorder_16s_C1R
 - image_copy_constant_border, 205
- nppiCopyConstBorder_16s_C3R
 - image_copy_constant_border, 206
- nppiCopyConstBorder_16s_C4R
 - image_copy_constant_border, 206
- nppiCopyConstBorder_16u_AC4R
 - image_copy_constant_border, 207
- nppiCopyConstBorder_16u_C1R
 - image_copy_constant_border, 207
- nppiCopyConstBorder_16u_C3R
 - image_copy_constant_border, 208
- nppiCopyConstBorder_16u_C4R
 - image_copy_constant_border, 208
- nppiCopyConstBorder_32f_AC4R
 - image_copy_constant_border, 209
- nppiCopyConstBorder_32f_C1R
 - image_copy_constant_border, 209
- nppiCopyConstBorder_32f_C3R
 - image_copy_constant_border, 210
- nppiCopyConstBorder_32f_C4R
 - image_copy_constant_border, 210
- nppiCopyConstBorder_32s_AC4R
 - image_copy_constant_border, 211
- nppiCopyConstBorder_32s_C1R
 - image_copy_constant_border, 211
- nppiCopyConstBorder_32s_C3R
 - image_copy_constant_border, 212
- nppiCopyConstBorder_32s_C4R
 - image_copy_constant_border, 212
- nppiCopyConstBorder_8u_AC4R
 - image_copy_constant_border, 213
- nppiCopyConstBorder_8u_C1R
 - image_copy_constant_border, 213
- nppiCopyConstBorder_8u_C3R
 - image_copy_constant_border, 214
- nppiCopyConstBorder_8u_C4R
 - image_copy_constant_border, 214
- nppiCopyReplicateBorder_16s_AC4R
 - image_copy_replicate_border, 218
- nppiCopyReplicateBorder_16s_C1R
 - image_copy_replicate_border, 218
- nppiCopyReplicateBorder_16s_C3R
 - image_copy_replicate_border, 219
- nppiCopyReplicateBorder_16s_C4R
 - image_copy_replicate_border, 219
- nppiCopyReplicateBorder_16u_AC4R
 - image_copy_replicate_border, 220
- nppiCopyReplicateBorder_16u_C1R
 - image_copy_replicate_border, 220
- nppiCopyReplicateBorder_16u_C3R
 - image_copy_replicate_border, 221
- nppiCopyReplicateBorder_16u_C4R
 - image_copy_replicate_border, 221
- nppiCopyReplicateBorder_32f_AC4R
 - image_copy_replicate_border, 221
- nppiCopyReplicateBorder_32f_C1R
 - image_copy_replicate_border, 222
- nppiCopyReplicateBorder_32f_C3R
 - image_copy_replicate_border, 222
- nppiCopyReplicateBorder_32f_C4R
 - image_copy_replicate_border, 223
- nppiCopyReplicateBorder_32s_AC4R
 - image_copy_replicate_border, 223
- nppiCopyReplicateBorder_32s_C1R
 - image_copy_replicate_border, 224
- nppiCopyReplicateBorder_32s_C3R
 - image_copy_replicate_border, 224
- nppiCopyReplicateBorder_32s_C4R
 - image_copy_replicate_border, 225
- nppiCopyReplicateBorder_8u_AC4R

- image_copy_replicate_border, 225
- nppiCopyReplicateBorder_8u_C1R
 - image_copy_replicate_border, 226
- nppiCopyReplicateBorder_8u_C3R
 - image_copy_replicate_border, 226
- nppiCopyReplicateBorder_8u_C4R
 - image_copy_replicate_border, 227
- nppiCopySubpix_16s_AC4R
 - image_copy_sub_pixel, 242
- nppiCopySubpix_16s_C1R
 - image_copy_sub_pixel, 243
- nppiCopySubpix_16s_C3R
 - image_copy_sub_pixel, 243
- nppiCopySubpix_16s_C4R
 - image_copy_sub_pixel, 244
- nppiCopySubpix_16u_AC4R
 - image_copy_sub_pixel, 244
- nppiCopySubpix_16u_C1R
 - image_copy_sub_pixel, 244
- nppiCopySubpix_16u_C3R
 - image_copy_sub_pixel, 245
- nppiCopySubpix_16u_C4R
 - image_copy_sub_pixel, 245
- nppiCopySubpix_32f_AC4R
 - image_copy_sub_pixel, 246
- nppiCopySubpix_32f_C1R
 - image_copy_sub_pixel, 246
- nppiCopySubpix_32f_C3R
 - image_copy_sub_pixel, 246
- nppiCopySubpix_32f_C4R
 - image_copy_sub_pixel, 247
- nppiCopySubpix_32s_AC4R
 - image_copy_sub_pixel, 247
- nppiCopySubpix_32s_C1R
 - image_copy_sub_pixel, 248
- nppiCopySubpix_32s_C3R
 - image_copy_sub_pixel, 248
- nppiCopySubpix_32s_C4R
 - image_copy_sub_pixel, 249
- nppiCopySubpix_8u_AC4R
 - image_copy_sub_pixel, 249
- nppiCopySubpix_8u_C1R
 - image_copy_sub_pixel, 249
- nppiCopySubpix_8u_C3R
 - image_copy_sub_pixel, 250
- nppiCopySubpix_8u_C4R
 - image_copy_sub_pixel, 250
- nppiCopyWrapBorder_16s_AC4R
 - image_copy_wrap_border, 230
- nppiCopyWrapBorder_16s_C1R
 - image_copy_wrap_border, 230
- nppiCopyWrapBorder_16s_C3R
 - image_copy_wrap_border, 231
- nppiCopyWrapBorder_16s_C4R
 - image_copy_wrap_border, 231
- nppiCopyWrapBorder_16u_AC4R
 - image_copy_wrap_border, 232
- nppiCopyWrapBorder_16u_C1R
 - image_copy_wrap_border, 232
- nppiCopyWrapBorder_16u_C3R
 - image_copy_wrap_border, 233
- nppiCopyWrapBorder_16u_C4R
 - image_copy_wrap_border, 233
- nppiCopyWrapBorder_32f_AC4R
 - image_copy_wrap_border, 234
- nppiCopyWrapBorder_32f_C1R
 - image_copy_wrap_border, 234
- nppiCopyWrapBorder_32f_C3R
 - image_copy_wrap_border, 235
- nppiCopyWrapBorder_32f_C4R
 - image_copy_wrap_border, 235
- nppiCopyWrapBorder_32s_AC4R
 - image_copy_wrap_border, 236
- nppiCopyWrapBorder_32s_C1R
 - image_copy_wrap_border, 236
- nppiCopyWrapBorder_32s_C3R
 - image_copy_wrap_border, 237
- nppiCopyWrapBorder_32s_C4R
 - image_copy_wrap_border, 237
- nppiCopyWrapBorder_8u_AC4R
 - image_copy_wrap_border, 238
- nppiCopyWrapBorder_8u_C1R
 - image_copy_wrap_border, 238
- nppiCopyWrapBorder_8u_C3R
 - image_copy_wrap_border, 239
- nppiCopyWrapBorder_8u_C4R
 - image_copy_wrap_border, 239
- nppiDCTable
 - typedefs_npp, 41
- NppiDifferentialKernel
 - typedefs_npp, 40
- nppiDup_16s_C1AC4R
 - image_duplicate_channel, 253
- nppiDup_16s_C1C3R
 - image_duplicate_channel, 253
- nppiDup_16s_C1C4R
 - image_duplicate_channel, 254
- nppiDup_16u_C1AC4R
 - image_duplicate_channel, 254
- nppiDup_16u_C1C3R
 - image_duplicate_channel, 254
- nppiDup_16u_C1C4R
 - image_duplicate_channel, 255
- nppiDup_32f_C1AC4R
 - image_duplicate_channel, 255
- nppiDup_32f_C1C3R
 - image_duplicate_channel, 255
- nppiDup_32f_C1C4R
 - image_duplicate_channel, 255

- image_duplicate_channel, 256
- nppiDup_32s_C1AC4R
 - image_duplicate_channel, 256
- nppiDup_32s_C1C3R
 - image_duplicate_channel, 256
- nppiDup_32s_C1C4R
 - image_duplicate_channel, 257
- nppiDup_8u_C1AC4R
 - image_duplicate_channel, 257
- nppiDup_8u_C1C3R
 - image_duplicate_channel, 257
- nppiDup_8u_C1C4R
 - image_duplicate_channel, 258
- nppiFree
 - image_memory_management, 52
- NppiHaarBuffer, 289
 - haarBuffer, 289
 - haarBufferSize, 289
- NppiHaarClassifier_32f, 290
 - classifiers, 290
 - classifierSize, 290
 - classifierStep, 290
 - counterDevice, 290
 - numClassifiers, 290
- NppiHuffmanTableType
 - typedefs_npp, 40
- NppiInterpolationMode
 - typedefs_npp, 41
- nppiMalloc_16s_C1
 - image_memory_management, 52
- nppiMalloc_16s_C2
 - image_memory_management, 52
- nppiMalloc_16s_C4
 - image_memory_management, 53
- nppiMalloc_16sc_C1
 - image_memory_management, 53
- nppiMalloc_16sc_C2
 - image_memory_management, 53
- nppiMalloc_16sc_C3
 - image_memory_management, 54
- nppiMalloc_16sc_C4
 - image_memory_management, 54
- nppiMalloc_16u_C1
 - image_memory_management, 54
- nppiMalloc_16u_C2
 - image_memory_management, 54
- nppiMalloc_16u_C3
 - image_memory_management, 55
- nppiMalloc_16u_C4
 - image_memory_management, 55
- nppiMalloc_32f_C1
 - image_memory_management, 55
- nppiMalloc_32f_C2
 - image_memory_management, 56
- nppiMalloc_32f_C3
 - image_memory_management, 56
- nppiMalloc_32f_C4
 - image_memory_management, 56
- nppiMalloc_32fc_C1
 - image_memory_management, 56
- nppiMalloc_32fc_C2
 - image_memory_management, 57
- nppiMalloc_32fc_C3
 - image_memory_management, 57
- nppiMalloc_32fc_C4
 - image_memory_management, 57
- nppiMalloc_32s_C1
 - image_memory_management, 58
- nppiMalloc_32s_C3
 - image_memory_management, 58
- nppiMalloc_32s_C4
 - image_memory_management, 58
- nppiMalloc_32sc_C1
 - image_memory_management, 58
- nppiMalloc_32sc_C2
 - image_memory_management, 59
- nppiMalloc_32sc_C3
 - image_memory_management, 59
- nppiMalloc_32sc_C4
 - image_memory_management, 59
- nppiMalloc_8u_C1
 - image_memory_management, 60
- nppiMalloc_8u_C2
 - image_memory_management, 60
- nppiMalloc_8u_C3
 - image_memory_management, 60
- nppiMalloc_8u_C4
 - image_memory_management, 60
- NppiMaskSize
 - typedefs_npp, 41
- NppiNorm
 - typedefs_npp, 41
- nppiNormInf
 - typedefs_npp, 42
- nppiNormL1
 - typedefs_npp, 42
- nppiNormL2
 - typedefs_npp, 42
- NppiPoint, 291
 - x, 291
 - y, 291
- NppiRect, 292
 - height, 292
 - width, 292
 - x, 292
 - y, 292
- nppiScale_16s8u_AC4R
 - image_scale, 191

- nppiScale_16s8u_C1R
 - image_scale, [191](#)
- nppiScale_16s8u_C3R
 - image_scale, [191](#)
- nppiScale_16s8u_C4R
 - image_scale, [192](#)
- nppiScale_16u8u_AC4R
 - image_scale, [192](#)
- nppiScale_16u8u_C1R
 - image_scale, [192](#)
- nppiScale_16u8u_C3R
 - image_scale, [193](#)
- nppiScale_16u8u_C4R
 - image_scale, [193](#)
- nppiScale_32f8u_AC4R
 - image_scale, [193](#)
- nppiScale_32f8u_C1R
 - image_scale, [194](#)
- nppiScale_32f8u_C3R
 - image_scale, [194](#)
- nppiScale_32f8u_C4R
 - image_scale, [195](#)
- nppiScale_32s8u_AC4R
 - image_scale, [195](#)
- nppiScale_32s8u_C1R
 - image_scale, [195](#)
- nppiScale_32s8u_C3R
 - image_scale, [196](#)
- nppiScale_32s8u_C4R
 - image_scale, [196](#)
- nppiScale_8u16s_AC4R
 - image_scale, [196](#)
- nppiScale_8u16s_C1R
 - image_scale, [197](#)
- nppiScale_8u16s_C3R
 - image_scale, [197](#)
- nppiScale_8u16s_C4R
 - image_scale, [197](#)
- nppiScale_8u16u_AC4R
 - image_scale, [198](#)
- nppiScale_8u16u_C1R
 - image_scale, [198](#)
- nppiScale_8u16u_C3R
 - image_scale, [198](#)
- nppiScale_8u16u_C4R
 - image_scale, [199](#)
- nppiScale_8u32f_AC4R
 - image_scale, [199](#)
- nppiScale_8u32f_C1R
 - image_scale, [199](#)
- nppiScale_8u32f_C3R
 - image_scale, [200](#)
- nppiScale_8u32f_C4R
 - image_scale, [200](#)
- nppiScale_8u32s_AC4R
 - image_scale, [201](#)
- nppiScale_8u32s_C1R
 - image_scale, [201](#)
- nppiScale_8u32s_C3R
 - image_scale, [201](#)
- nppiScale_8u32s_C4R
 - image_scale, [202](#)
- nppiSet_16s_AC4MR
 - image_set, [69](#)
- nppiSet_16s_AC4R
 - image_set, [70](#)
- nppiSet_16s_C1MR
 - image_set, [70](#)
- nppiSet_16s_C1R
 - image_set, [70](#)
- nppiSet_16s_C2R
 - image_set, [71](#)
- nppiSet_16s_C3CR
 - image_set, [71](#)
- nppiSet_16s_C3MR
 - image_set, [71](#)
- nppiSet_16s_C3R
 - image_set, [72](#)
- nppiSet_16s_C4CR
 - image_set, [72](#)
- nppiSet_16s_C4MR
 - image_set, [72](#)
- nppiSet_16s_C4R
 - image_set, [73](#)
- nppiSet_16sc_AC4R
 - image_set, [73](#)
- nppiSet_16sc_C1R
 - image_set, [73](#)
- nppiSet_16sc_C2R
 - image_set, [74](#)
- nppiSet_16sc_C3R
 - image_set, [74](#)
- nppiSet_16sc_C4R
 - image_set, [74](#)
- nppiSet_16u_AC4MR
 - image_set, [75](#)
- nppiSet_16u_AC4R
 - image_set, [75](#)
- nppiSet_16u_C1MR
 - image_set, [75](#)
- nppiSet_16u_C1R
 - image_set, [76](#)
- nppiSet_16u_C2R
 - image_set, [76](#)
- nppiSet_16u_C3CR
 - image_set, [76](#)
- nppiSet_16u_C3MR
 - image_set, [77](#)

- nppiSet_16u_C3R
image_set, 77
- nppiSet_16u_C4CR
image_set, 77
- nppiSet_16u_C4MR
image_set, 78
- nppiSet_16u_C4R
image_set, 78
- nppiSet_32f_AC4MR
image_set, 78
- nppiSet_32f_AC4R
image_set, 79
- nppiSet_32f_C1MR
image_set, 79
- nppiSet_32f_C1R
image_set, 79
- nppiSet_32f_C2R
image_set, 80
- nppiSet_32f_C3CR
image_set, 80
- nppiSet_32f_C3MR
image_set, 80
- nppiSet_32f_C3R
image_set, 81
- nppiSet_32f_C4CR
image_set, 81
- nppiSet_32f_C4MR
image_set, 81
- nppiSet_32f_C4R
image_set, 82
- nppiSet_32fc_AC4R
image_set, 82
- nppiSet_32fc_C1R
image_set, 82
- nppiSet_32fc_C2R
image_set, 83
- nppiSet_32fc_C3R
image_set, 83
- nppiSet_32fc_C4R
image_set, 83
- nppiSet_32s_AC4MR
image_set, 84
- nppiSet_32s_AC4R
image_set, 84
- nppiSet_32s_C1MR
image_set, 84
- nppiSet_32s_C1R
image_set, 85
- nppiSet_32s_C2R
image_set, 85
- nppiSet_32s_C3CR
image_set, 85
- nppiSet_32s_C3MR
image_set, 86
- nppiSet_32s_C3R
image_set, 86
- nppiSet_32s_C4CR
image_set, 86
- nppiSet_32s_C4MR
image_set, 87
- nppiSet_32s_C4R
image_set, 87
- nppiSet_32sc_AC4R
image_set, 87
- nppiSet_32sc_C1R
image_set, 88
- nppiSet_32sc_C2R
image_set, 88
- nppiSet_32sc_C3R
image_set, 88
- nppiSet_32sc_C4R
image_set, 89
- nppiSet_32u_AC4R
image_set, 89
- nppiSet_32u_C1R
image_set, 89
- nppiSet_32u_C2R
image_set, 90
- nppiSet_32u_C3R
image_set, 90
- nppiSet_32u_C4R
image_set, 90
- nppiSet_8s_AC4R
image_set, 91
- nppiSet_8s_C1R
image_set, 91
- nppiSet_8s_C2R
image_set, 91
- nppiSet_8s_C3R
image_set, 92
- nppiSet_8s_C4R
image_set, 92
- nppiSet_8u_AC4MR
image_set, 92
- nppiSet_8u_AC4R
image_set, 93
- nppiSet_8u_C1MR
image_set, 93
- nppiSet_8u_C1R
image_set, 93
- nppiSet_8u_C2R
image_set, 94
- nppiSet_8u_C3CR
image_set, 94
- nppiSet_8u_C3MR
image_set, 94
- nppiSet_8u_C3R
image_set, 95

- nppiSet_8u_C4CR
 - image_swap_channels, 95
- nppiSet_8u_C4MR
 - image_swap_channels, 95
- nppiSet_8u_C4R
 - image_swap_channels, 96
- NppiSize, 293
 - height, 293
 - width, 293
- nppiSwapChannels_16s_AC4R
 - image_swap_channels, 269
- nppiSwapChannels_16s_C3C4R
 - image_swap_channels, 269
- nppiSwapChannels_16s_C3IR
 - image_swap_channels, 269
- nppiSwapChannels_16s_C3R
 - image_swap_channels, 270
- nppiSwapChannels_16s_C4C3R
 - image_swap_channels, 270
- nppiSwapChannels_16s_C4IR
 - image_swap_channels, 271
- nppiSwapChannels_16s_C4R
 - image_swap_channels, 271
- nppiSwapChannels_16u_AC4R
 - image_swap_channels, 271
- nppiSwapChannels_16u_C3C4R
 - image_swap_channels, 272
- nppiSwapChannels_16u_C3IR
 - image_swap_channels, 272
- nppiSwapChannels_16u_C3R
 - image_swap_channels, 273
- nppiSwapChannels_16u_C4C3R
 - image_swap_channels, 273
- nppiSwapChannels_16u_C4IR
 - image_swap_channels, 274
- nppiSwapChannels_16u_C4R
 - image_swap_channels, 274
- nppiSwapChannels_32f_AC4R
 - image_swap_channels, 274
- nppiSwapChannels_32f_C3C4R
 - image_swap_channels, 275
- nppiSwapChannels_32f_C3IR
 - image_swap_channels, 275
- nppiSwapChannels_32f_C3R
 - image_swap_channels, 276
- nppiSwapChannels_32f_C4C3R
 - image_swap_channels, 276
- nppiSwapChannels_32f_C4IR
 - image_swap_channels, 277
- nppiSwapChannels_32f_C4R
 - image_swap_channels, 277
- nppiSwapChannels_32s_AC4R
 - image_swap_channels, 277
- nppiSwapChannels_32s_C3C4R
 - image_swap_channels, 278
- nppiSwapChannels_32s_C3IR
 - image_swap_channels, 278
- nppiSwapChannels_32s_C3R
 - image_swap_channels, 279
- nppiSwapChannels_32s_C4C3R
 - image_swap_channels, 279
- nppiSwapChannels_32s_C4IR
 - image_swap_channels, 280
- nppiSwapChannels_32s_C4R
 - image_swap_channels, 280
- nppiSwapChannels_8u_AC4R
 - image_swap_channels, 280
- nppiSwapChannels_8u_C3C4R
 - image_swap_channels, 281
- nppiSwapChannels_8u_C3IR
 - image_swap_channels, 281
- nppiSwapChannels_8u_C3R
 - image_swap_channels, 282
- nppiSwapChannels_8u_C4C3R
 - image_swap_channels, 282
- nppiSwapChannels_8u_C4IR
 - image_swap_channels, 283
- nppiSwapChannels_8u_C4R
 - image_swap_channels, 283
- nppiTranspose_16s_C1R
 - image_transpose, 260
- nppiTranspose_16s_C3R
 - image_transpose, 260
- nppiTranspose_16s_C4R
 - image_transpose, 261
- nppiTranspose_16u_C1R
 - image_transpose, 261
- nppiTranspose_16u_C3R
 - image_transpose, 261
- nppiTranspose_16u_C4R
 - image_transpose, 262
- nppiTranspose_32f_C1R
 - image_transpose, 262
- nppiTranspose_32f_C3R
 - image_transpose, 262
- nppiTranspose_32f_C4R
 - image_transpose, 263
- nppiTranspose_32s_C1R
 - image_transpose, 263
- nppiTranspose_32s_C3R
 - image_transpose, 263
- nppiTranspose_32s_C4R
 - image_transpose, 264
- nppiTranspose_8u_C1R
 - image_transpose, 264
- nppiTranspose_8u_C3R
 - image_transpose, 264
- nppiTranspose_8u_C4R

- image_transpose, 265
- NppLibraryVersion, 294
 - build, 294
 - major, 294
 - minor, 294
- NppRoundMode
 - typedefs_npp, 42
- nppSetStream
 - core_npp, 29
- NppStatus
 - typedefs_npp, 42
- NppsZCType
 - typedefs_npp, 44
- nppZCC
 - typedefs_npp, 45
- nppZCR
 - typedefs_npp, 45
- nppZCXor
 - typedefs_npp, 45
- numClassifiers
 - NppiHaarClassifier_32f, 290
- re
 - NPP_ALIGN_16, 286
 - NPP_ALIGN_8, 287, 288
- Scale, 188
- Set, 63
- Swap Channels, 266
- Transpose, 259
- typedefs_npp
 - NPP_AFFINE_QUAD_INCORRECT_-
WARNING, 44
 - NPP_ALG_HINT_ACCURATE, 39
 - NPP_ALG_HINT_FAST, 39
 - NPP_ALG_HINT_NONE, 39
 - NPP_ALIGNMENT_ERROR, 43
 - NPP_ANCHOR_ERROR, 43
 - NPP_BAD_ARGUMENT_ERROR, 44
 - NPP_BORDER_CONSTANT, 40
 - NPP_BORDER_MIRROR, 40
 - NPP_BORDER_NONE, 40
 - NPP_BORDER_REPLICATE, 40
 - NPP_BORDER_UNDEFINED, 40
 - NPP_BORDER_WRAP, 40
 - NPP_BOTH_AXIS, 40
 - NPP_CHANNEL_ERROR, 43
 - NPP_CHANNEL_ORDER_ERROR, 43
 - NPP_CMP_EQ, 39
 - NPP_CMP_GREATER, 39
 - NPP_CMP_GREATER_EQ, 39
 - NPP_CMP_LESS, 38
 - NPP_CMP_LESS_EQ, 38
 - NPP_COEFFICIENT_ERROR, 43
 - NPP_COI_ERROR, 43
 - NPP_CONTEXT_MATCH_ERROR, 44
 - NPP_CORRUPTED_DATA_ERROR, 43
 - NPP_CUDA_1_0, 39
 - NPP_CUDA_1_1, 39
 - NPP_CUDA_1_2, 39
 - NPP_CUDA_1_3, 39
 - NPP_CUDA_2_0, 39
 - NPP_CUDA_2_1, 39
 - NPP_CUDA_3_0, 39
 - NPP_CUDA_3_2, 39
 - NPP_CUDA_3_5, 39
 - NPP_CUDA_3_7, 39
 - NPP_CUDA_5_0, 39
 - NPP_CUDA_5_2, 39
 - NPP_CUDA_5_3, 39
 - NPP_CUDA_6_0, 39
 - NPP_CUDA_KERNEL_EXECUTION_-
ERROR, 43
 - NPP_CUDA_NOT_CAPABLE, 39
 - NPP_CUDA_UNKNOWN_VERSION, 39
 - NPP_DATA_TYPE_ERROR, 44
 - NPP_DIVIDE_BY_ZERO_ERROR, 44
 - NPP_DIVIDE_BY_ZERO_WARNING, 44
 - NPP_DIVISOR_ERROR, 43
 - NPP_DOUBLE_SIZE_WARNING, 44
 - NPP_ERROR, 44
 - NPP_ERROR_RESERVED, 44
 - NPP_FFT_FLAG_ERROR, 44
 - NPP_FFT_ORDER_ERROR, 44
 - NPP_FILTER_SCHARR, 40
 - NPP_FILTER_SOBEL, 40
 - NPP_HAAR_CLASSIFIER_PIXEL_-
MATCH_ERROR, 43
 - NPP_HISTOGRAM_NUMBER_OF_-
LEVELS_ERROR, 43
 - NPP_HORIZONTAL_AXIS, 40
 - NPP_INTERPOLATION_ERROR, 44
 - NPP_INVALID_DEVICE_POINTER_-
ERROR, 43
 - NPP_INVALID_HOST_POINTER_ERROR,
43
 - NPP_LUT_NUMBER_OF_LEVELS_-
ERROR, 43
 - NPP_LUT_PALETTE_BITSIZE_ERROR, 43
 - NPP_MASK_SIZE_11_X_11, 41
 - NPP_MASK_SIZE_13_X_13, 41
 - NPP_MASK_SIZE_15_X_15, 41
 - NPP_MASK_SIZE_1_X_3, 41
 - NPP_MASK_SIZE_1_X_5, 41
 - NPP_MASK_SIZE_3_X_1, 41
 - NPP_MASK_SIZE_3_X_3, 41
 - NPP_MASK_SIZE_5_X_1, 41

- NPP_MASK_SIZE_5_X_5, 41
- NPP_MASK_SIZE_7_X_7, 41
- NPP_MASK_SIZE_9_X_9, 41
- NPP_MASK_SIZE_ERROR, 43
- NPP_MEMCPY_ERROR, 43
- NPP_MEMFREE_ERROR, 43
- NPP_MEMORY_ALLOCATION_ERR, 44
- NPP_MEMSET_ERROR, 43
- NPP_MIRROR_FLIP_ERROR, 44
- NPP_MISALIGNED_DST_ROI_WARNING, 44
- NPP_MOMENT_00_ZERO_ERROR, 44
- NPP_NO_ERROR, 44
- NPP_NO_MEMORY_ERROR, 44
- NPP_NO_OPERATION_WARNING, 44
- NPP_NOT_EVEN_STEP_ERROR, 43
- NPP_NOT_IMPLEMENTED_ERROR, 44
- NPP_NOT_SUFFICIENT_COMPUTE_CAPABILITY, 43
- NPP_NOT_SUPPORTED_MODE_ERROR, 43
- NPP_NULL_POINTER_ERROR, 44
- NPP_NUMBER_OF_CHANNELS_ERROR, 43
- NPP_OUT_OFF_RANGE_ERROR, 44
- NPP_OVERFLOW_ERROR, 43
- NPP_QUADRANGLE_ERROR, 43
- NPP_QUALITY_INDEX_ERROR, 43
- NPP_RANGE_ERROR, 44
- NPP_RECTANGLE_ERROR, 43
- NPP_RESIZE_FACTOR_ERROR, 44
- NPP_RESIZE_NO_OPERATION_ERROR, 43
- NPP_RND_FINANCIAL, 42
- NPP_RND_NEAR, 42
- NPP_RND_ZERO, 42
- NPP_ROUND_MODE_NOT_SUPPORTED_ERROR, 43
- NPP_ROUND_NEAREST_TIES_AWAY_FROM_ZERO, 42
- NPP_ROUND_NEAREST_TIES_TO_EVEN, 42
- NPP_ROUND_TOWARD_ZERO, 42
- NPP_SCALE_RANGE_ERROR, 44
- NPP_SIZE_ERROR, 44
- NPP_STEP_ERROR, 44
- NPP_STRIDE_ERROR, 43
- NPP_SUCCESS, 44
- NPP_TEXTURE_BIND_ERROR, 43
- NPP_THRESHOLD_ERROR, 44
- NPP_THRESHOLD_NEGATIVE_LEVEL_ERROR, 44
- NPP_VERTICAL_AXIS, 40
- NPP_WRONG_INTERSECTION_QUAD_WARNING, 44
- NPP_WRONG_INTERSECTION_ROI_ERROR, 43
- NPP_WRONG_INTERSECTION_ROI_WARNING, 44
- NPP_ZC_MODE_NOT_SUPPORTED_ERROR, 43
- NPP_ZERO_MASK_VALUE_ERROR, 43
- NPPI_BAYER_BGGR, 40
- NPPI_BAYER_GBRG, 40
- NPPI_BAYER_GRBG, 40
- NPPI_BAYER_RGGB, 40
- NPPI_INTER_CUBIC, 41
- NPPI_INTER_CUBIC2P_B05C03, 41
- NPPI_INTER_CUBIC2P_BSPLINE, 41
- NPPI_INTER_CUBIC2P_CATMULLROM, 41
- NPPI_INTER_LANCZOS, 41
- NPPI_INTER_LANCZOS3_ADVANCED, 41
- NPPI_INTER_LINEAR, 41
- NPPI_INTER_NN, 41
- NPPI_INTER_SUPER, 41
- NPPI_INTER_UNDEFINED, 41
- NPPI_OP_ALPHA_ATOP, 39
- NPPI_OP_ALPHA_ATOP_PREMUL, 40
- NPPI_OP_ALPHA_IN, 39
- NPPI_OP_ALPHA_IN_PREMUL, 40
- NPPI_OP_ALPHA_OUT, 39
- NPPI_OP_ALPHA_OUT_PREMUL, 40
- NPPI_OP_ALPHA_OVER, 39
- NPPI_OP_ALPHA_OVER_PREMUL, 40
- NPPI_OP_ALPHA_PLUS, 39
- NPPI_OP_ALPHA_PLUS_PREMUL, 40
- NPPI_OP_ALPHA_PREMUL, 40
- NPPI_OP_ALPHA_XOR, 39
- NPPI_OP_ALPHA_XOR_PREMUL, 40
- NPPI_SMOOTH_EDGE, 41
- nppiACTable, 41
- nppiDCTable, 41
- nppiNormInf, 42
- nppiNormL1, 42
- nppiNormL2, 42
- nppZCC, 45
- nppZCR, 45
- nppZCxor, 45
- typedefs_npp
 - NPP_MAX_16S, 37
 - NPP_MAX_16U, 37
 - NPP_MAX_32S, 37
 - NPP_MAX_32U, 37
 - NPP_MAX_64S, 37
 - NPP_MAX_64U, 37
 - NPP_MAX_8S, 37

NPP_MAX_8U, 37
NPP_MAXABS_32F, 37
NPP_MAXABS_64F, 37
NPP_MIN_16S, 37
NPP_MIN_16U, 38
NPP_MIN_32S, 38
NPP_MIN_32U, 38
NPP_MIN_64S, 38
NPP_MIN_64U, 38
NPP_MIN_8S, 38
NPP_MIN_8U, 38
NPP_MINABS_32F, 38
NPP_MINABS_64F, 38
NppCmpOp, 38
NppGpuComputeCapability, 39
NppHintAlgorithm, 39
NppiAlphaOp, 39
NppiAxis, 40
NppiBayerGridPosition, 40
NppiBorderType, 40
NppiDifferentialKernel, 40
NppiHuffmanTableType, 40
NppiInterpolationMode, 41
NppiMaskSize, 41
NppiNorm, 41
NppRoundMode, 42
NppStatus, 42
NppsZCType, 44

width

NppiRect, 292
NppiSize, 293

x

NppiPoint, 291
NppiRect, 292

y

NppiPoint, 291
NppiRect, 292