## Contents

# SurfaceLevel 2.0

## Overview

SurfaceLevel 2.0 is a texture converter and basic image editor. It loads many image file formats, converts to and from almost every video-game API texture (OpenGL, Vulkan, Metal, and Direct3D 12), offers a fast and high-quality resampler, and saves to many file formats. It is useful for performing nearly all video-game–related texture operations, such as compression, pre-multiplying alpha, generating normal maps, resampling/generating crisp mipmaps, etc., and for general-purpose image resampling, colorspace conversion, format conversion, etc. SurfaceLevel 2.0 is meant to gather a large quantity of features together and to provide more options for those features than you will find in any other tool. For example, when resampling, it is possible to specify different samplers for your image width, height, and depth, and different samplers for the RGB colors and the alpha channel—the sharpest resamplers often create ringing, which is particularly bad for the alpha channel; here you can using a ringing filter for the colors but a non-ringing filter for alpha (while specifying the same or different filters for both the color and alpha widths, heights, and depths).

SurfaceLevel 2.0 seeks to be useful in the level of detail for each feature, the number of features, and the performance of each feature. Supports volume (3-D) textures, texture arrays, cube maps, and mipmaps.

## Control Flow

SurfaceLevel 2.0 doesn't try to be overly smart, so as long as you understand the basic internal workflow it should be easy to predict what it will do in any edge cases. Internally it performs the following operations in order:

1. Loads the image file.
2. Performs a conversion to the desired target format.
    a. Converts to RGBA64F.
    b. Converts to linear, applying any applicable colorspace conversions and gamma corrections necessary.
    c. Applies requested transforms, such as swapping, swizzling, flipping, etc.
    d. Performs resampling and generates mipmaps.
    e. Applies pre-multiplied alpha.
    f. Converts to the requested texture format, applying gamma-correction as necessary, etc.
3. Saves to the desired file.

a. If the current format is not directly supported by the file, it is converted to the nearest format that is supported by the file.

The ways it tries to be smart:

1. By default, it will try to ensure only 1 gamma transform occurs from source to linear and from linear to destination. If images do not contain gamma data, they are assumed to be sRGB (`-g`, `-gamma`, `-srgb`, `-rgbe`, and `-linear` to change this assumption). If they contain gamma data, whether implicit by the format (`VK_FORMAT_B8G8R8_SRGB`, for example) or from an embedded ICC profile, that is used instead. However, if you manually supply a source gamma value with `-g`, `-gamma`, `-srgb`, `-rgbe`, or `-linear`, your supplied gamma will be stacked with any contained/embedded gamma data. This can allow you to correct images that may have been saved with incorrect gamma. To specify your own gamma curve to be used *in-place* of any embedded or selected colorspace profiles, use `-g`, `-gamma`, `-srgb`, `-rgbe`, or `-linear` to define your own gamma curve and `-ignore_input_colorspace_gamma` to ignore the gamma curve in any colorspace profiles being used.
2. To simplify the process, it is not necessary to specify the export format. If your conversion format is not supported, it will find the closest match that is supported. It will try to ensure a lossless conversion, but the option to specify the export format is always available if needed. This only applies to general image formats. For specialized GPU-leaning formats, such as DDS, KTX, PVR, etc., export will fail if the format specified in `-format` is not supported by the file.
   For example, `-png_format` is never strictly necessary because any format supplied by `-format` can be automatically converted to a format supported by the PNG file specification; `-png_format` is entirely optional. No automatic conversion is made when saving to DDS, for example, so the format supplied by `-format` must be supported by the DDS file specification.

# Commands

## File

| Command | Parameter | Description |
|---|---|---|
| **-file** | <file path> | A path to an image file to load and convert. The **-file**, **-yuv_file**, and **-outfile** commands can be used multiple times to load and save multiple files. |

| | | |
|---|---|---|
| **-yuv_file** | <file path> <width> <height> | Path to a YUV file to load.  If the extension does not indicate the YUV encoding, then **-yuv_input_format** must be called to specify the YUV encoding.<br>Recognized file extensions:<br>yuv444p16<br>yuv444p12le<br>yuv444p10le<br>yuv444p<br>yuv444y16<br>yuv444y12le<br>yuv444y10le<br>yuv444y<br>yuv422p16<br>yuv422p12le<br>yuv422p10le<br>yuv422p<br>yuv422y16<br>yuv422y12le<br>yuv422y10le<br>yuv422y<br>yuv420p16<br>yuv420p12le<br>yuv420p10le<br>yuv420p<br>yuv420y16<br>yuv420y12le<br>yuv420y10le<br>yuv420y<br>yuva12le<br>yuva10le<br>uyvy16<br>uyvy12le<br>uyvy10le<br>uyvy<br>y210<br>yuy2<br>yv12<br>yv12<br>p016<br>p010<br>y012<br>y010<br>nv12 |

| | | |
|---|---|---|
| | | nv21<br>y416<br>y410<br>ayuv<br>Extensions other than these will require the format to be explicitly set. |
| **-outfile**<br>**-out_file** | <file path> | The path to which to save the file supplied with the last **-file** command.<br>The destination file format is determined by the file extension.<br>Currently supported formats:<br>PNG<br>BMP<br>TGA<br>JPG<br>J2K<br>JP2<br>EXR<br>DDS<br>KTX<br>PVR |

## Gamma/Colorspaces

| Command | Parameter | Description |
|---|---|---|
| **-gamma**<br>**-g** | <gamma> | Sets the input gamma power.<br>Defaults to **-2.2** (precise sRGB).<br>See Notes. |
| **-targetgamma** | <gamma> | Sets the output gamma power.<br>Defaults to **-2.2** (precise sRGB). |
| **-rgbe**<br>**-linear** | | Sets the source and output gamma to 0.0. |
| **-srgb** | | Sets the source gamma to -2.2 (precise sRGB). |
| **-target_srgb** | | Sets the output gamma to -2.2 (precise sRGB). |
| **-input_colorspace** | sRGB<br>sRGB_precise | Sets the source colorspace profile to an accurate no-gap sRGB. |

| | | |
|---|---|---|
| | sRGB_std<br>sRGB_standard | Sets the source colorspace profile to the standard sRGB. |
| | smpte_170<br>smpte_170m<br>170m | Sets the source colorspace profile to an accurate no-gap SMPTE 170M-1999. |
| | smpte_170_std<br>smpte_170m_std<br>170m_std<br>smpte_170_standard<br>smpte_170m_standard<br>170m_standard | Sets the source colorspace profile to the standard SMPTE 170M-1999. |
| | rec709<br>rec.709<br>bt709<br>bt.709<br>itu_bt709<br>itu_bt.709 | Sets the source colorspace profile to an accurate no-gap ITU-R Recommendation BT.709-5. |
| | rec709_std<br>rec.709_std<br>bt709_std<br>bt.709_std<br>itu_bt709_std<br>itu_bt.709_std<br>rec709_standard<br>rec.709_standard<br>bt709_standard<br>bt.709_standard<br>itu_bt709_standard<br>itu_bt.709_standard | Sets the source colorspace profile to the standard ITU-R Recommendation BT.709-5. |
| | adobe<br>adobergb<br>adobe_rgb | Sets the source colorspace profile to Adobe RGB (1998) Color Image Encoding Version 2005-05. |
| | bt2020<br>bt.2020<br>itu_bt2020<br>itu_bt.2020 | Sets the source colorspace profile to an accurate no-gap ITU-R Recommendation BT.2020. |
| | bt2020_std<br>bt.2020_std | Sets the source colorspace profile to |

| | | |
|---|---|---|
| | itu_bt2020_std<br>itu_bt.2020_std<br>bt2020_standard<br>bt.2020_standard<br>itu_bt2020_standard<br>itu_bt.2020_standard | the standard ITU-R Recommendation BT.2020. |
| | dcip3<br>dci-p3<br>dci_p3 | Sets the source colorspace profile to SMPTE RP 431-2:2011. |
| | smpte_240<br>smpte_240m<br>240m | Sets the source colorspace profile to an accurate no-gap SMPTE 240M-1999. |
| | smpte_240_std<br>smpte_240m_std<br>240m_std<br>smpte_240_standard<br>smpte_240m_standard<br>240m_standard | Sets the source colorspace profile to the standard SMPTE 240M-1999. |
| | ntsc_1953<br>ntsc1953 | Sets the source colorspace profile to an accurate no-gap NTSC 1953. |
| | ntsc_1953_std<br>ntsc1953_std<br>ntsc_1953_standard<br>ntsc1953_standard | Sets the source colorspace profile to the standard NTSC 1953. |
| | tech_3213<br>tech3213 | Sets the source colorspace profile to an accurate no-gap EBU Tech. 3213. |
| | tech_3213_std<br>tech3213_std<br>tech_3213_standard<br>tech3213_standard | Sets the source colorspace profile to the standard EBU Tech. 3213. |
| | displayp3<br>display-p3<br>display_p3 | Sets the source colorspace profile to an accurate no-gap Display P3 Color Encoding (v 1.0). |
| | displayp3_std<br>display-p3_std<br>display_p3_std | Sets the source colorspace profile to |

| | | |
|---|---|---|
| | displayp3_standard<br>display-p3_standard<br>display_p3_standard | the standard Display P3 Color Encoding (v 1.0). |
| | rec601<br>rec.601<br>bt601<br>bt.601<br>itu_bt601<br>itu_bt.601 | Sets the source colorspace profile to an accurate no-gap ITU-R Recommendation BT.601 (525). |
| | rec601_std<br>rec.601_std<br>bt601_std<br>bt.601_std<br>itu_bt601_std<br>itu_bt.601_std<br>rec601_standard<br>rec.601_standard<br>bt601_standard<br>bt.601_standard<br>itu_bt601_standard<br>itu_bt.601_standard | Sets the source colorspace profile to the standard ITU-R Recommendation BT.601 (525). |
| | rec601_pal<br>rec.601_pal<br>bt601_pal<br>bt.601_pal<br>itu_bt601_pal<br>itu_bt.601_pal | Sets the source colorspace profile to an accurate no-gap ITU-R Recommendation BT.601 (625). |
| | rec601_pal_std<br>rec.601_pal_std<br>bt601_pal_std<br>bt.601_pal_std<br>itu_bt601_pal_std<br>itu_bt.601_pal_std<br>rec601_pal_standard<br>rec.601_pal_standard<br>bt601_pal_standard<br>bt.601_pal_standard<br>itu_bt601_pal_standard<br>itu_bt.601_pal_standard | Sets the source colorspace profile to the standard ITU-R Recommendation BT.601 (625). |
| | generic_film<br>film | Sets the source colorspace profile to generic film. |

| | | |
|---|---|---|
| | bt470_ntsc<br>bt470_m_ntsc | Sets the source colorspace profile to an accurate no-gap Rec. ITU-R BT.470-6 (M/NTSC). |
| | bt470_pal<br>bt470_m_pal | Sets the source colorspace profile to an accurate no-gap Rec. ITU-R BT.470-6 (M/PAL). |
| | bt470_b<br>bt470_b1<br>bt470_d<br>bt470_d1<br>bt470_g<br>bt470_h<br>bt470_k<br>bt470_k1<br>bt470_l<br>bt470_n_pal<br>bt470_secam<br>bt470_l_secam | Sets the source colorspace profile to an accurate no-gap Rec. ITU-R BT.470-6 (B, B1, D, D1, G, H, K, N/PAL, K1, L/SECAM). |
| | ntsc_1987<br>smpte_c | Sets the source colorspace profile to SMPTE C with a pow(2.2) curve. |
| | ntsc_1987_std<br>smpte_c_std | Sets the source colorspace profile to the standard SMPTE C. |
| | romm_rgb<br>rommrgb | Sets the source colorspace profile to Reference Output Medium Metric RGB (ROMM RGB). |
| | rimm_rgb<br>rimmrgb | Sets the source colorspace profile to Reference Input Medium Metric RGB (RIMM RGB). |
| | erimm_rgb<br>erimmrgb | Sets the source colorspace profile to Extended Reference Input Medium Metric RGB (ERIMM RGB). |

| | | |
|---|---|---|
| | plasa<br>plasa_ansi | Sets the source colorspace profile to PLASA ANSI E1.54. |
| | protune<br>gopro | Sets the source colorspace profile to Protune Native (GoPro). |
| | s-gamut<br>sgamut<br>s_gamut | Sets the source colorspace profile to S-Gamut. |
| | s-gamut3<br>sgamut3<br>s_gamut3 | Sets the source colorspace profile to S-Gamut3. |
| | s-gamut3cine<br>sgamut3cine<br>s_gamut3cine<br>s-gamut3_cine<br>sgamut3_cine<br>s_gamut3_cine | Sets the source colorspace profile to S-Gamut3.Cine. |
| **-target_colorspace** | Same as for **-input_colorspace**. | Sets the output colorspace profile. |
| **-input_colorspace_file** | <file path> | Sets the input colorspace profile. Loads .ICC and .ICM files. |
| **-target_colorspace_file** | <file path> | Sets the output colorspace profile. Loads .ICC and .ICM files. |
| **-dont_embed_icc** | | No colorspace profile will be embedded into files with colorspace-profile support. |
| **-embed_icc** | | Any specified output colorspace profiles will be embedded into files with colorspace-profile support. This is the default. |
| **-ignore_input_colorspace_gamma** | | The gamma in any supplied or embedded input colorspace profile will be ignored. |

| | | |
|---|---|---|
| **-rendering_intent**<br>**-render_intent** | perceptual | All colors are scaled to fit into the target colorspace. Useful for converting from wide colorspaces to more narrow ones. |
| | relative_colorimetric | Colors in gamut are unchanged, but colors outside of gamut are clipped to the nearest in-gamut color.<br>This is the default. |
| | saturation | Like **perceptual**, but tends to make colors more saturated. |
| | absolute_colorimetric | Not intended for color conversion, but rather typically for digital inkjet proofing. |

## Resampling

| **Command** | **Parameter** | **Description** |
|---|---|---|
| **-filter** | box<br>point | Applies the selected filter to all non-mipmap filters. |
| | tent<br>linear | The default mipmap alpha filter. |
| | quadraticsharp<br>quadratic_sharp | The default non-mipmap filter. |
| | quadratic | |
| | quadraticapprox<br>quadraticapproximate<br>quadratic_approximate | |
| | quadraticmix<br>quadratic_mix | |
| | kaiser | |
| | lanczos2 | |
| | lanczos3 | |
| | lanczos4 | |
| | lanczos6 | |
| | lanczos8 | |
| | lanczos12 | |
| | lanczos64 | |

| | | |
|---|---|---|
| | mitchell | One of the best choices for upscaling. |
| | catmul<br>catmulrom<br>catmul_rom<br>catmul-rom | |
| | bspline<br>b-spline<br>b_spline | |
| | cardinal<br>card<br>cardinaluniform<br>cardinal_uniform | The default mipmap color filter. |
| | hermite | |
| | hamming | |
| | hanning | |
| | blackman | |
| | gaussiansharp<br>gaussian_sharp | |
| | gaussian | |
| | bell | |
| **-filterw** | Same as for **-filter**. | Sets the non-mipmap color and alpha *width* filter. |
| **-filterh** | Same as for **-filter**. | Sets the non-mipmap color and alpha *height* filter. |
| **- filterd** | Same as for **-filter**. | Sets the non-mipmap color and alpha *depth* filter. |
| **-filterw_color** | Same as for **-filter**. | Sets the non-mipmap color *width* filter. |
| **-filterh_color** | Same as for **-filter**. | Sets the non-mipmap color *height* filter. |
| **-filterd_color** | Same as for **-filter**. | Sets the non-mipmap color *depth* filter. |
| **-filterw_alpha** | Same as for **-filter**. | Sets the non-mipmap alpha *width* filter. |
| **-filterh_alpha** | Same as for **-filter**. | Sets the non-mipmap alpha *height* filter. |
| **-filterd_alpha** | Same as for **-filter**. | Sets the non-mipmap alpha *depth* filter. |
| **-prescale** | <new width> <new height> | Resamples the image to the given width/height using the selected non-mipmap filters. |

| | | |
|---|---|---|
| **-prescale3**<br>**-resample_size** | <new width> <new height> <new depth> | Resamples the image to the given width/height/depth using the selected non-mipmap filters. |
| **-resample_to** | nearest | Resamples to the nearest power of 2 in each dimension. |
| | lo | Resamples to the next power-of-2 down. |
| | hi | Resamples to the next power-of-2 up. |
| **-rel_scale** | <width multiplier> <height multiplier> | Resamples by the given width and height multipliers. |
| **-rel_scale3** | <width multiplier> <height multiplier> <depth multiplier> | Resamples by the given width, height, and depth multipliers. |
| **-clamp2**<br>**-clamp** | <width> <height> | Clamps the image to the given width and height. |
| **-clamp3** | <width> <height> <depth> | Clamps the image to the given width, height, and depth. |

## Texture Addressing

| **Command** | **Parameter** | **Description** |
|---|---|---|
| **-textureaddressing**<br>**-ta** | clamp | U, V, and W coordinates are clamped to the edge of the texture.<br>Equal to `D3D12_TEXTURE_ADDRESS_MODE_CLAMP`. |
| | repeat<br>wrap | U, V, and W coordinates repeat beyond the 0..1 range.<br>Equal to `D3D12_TEXTURE_ADDRESS_MODE_WRAP`. |
| | mirror<br>reflect | U, V, and W are mirrored beyond the 0..1 range.<br>Equal to `D3D12_TEXTURE_ADDRESS_MODE_MIRROR`. |
| | mirroronce<br>mirror_once | U, V, and W are mirrored 1 time beyond the 0..1 range, after which clamping is used.<br>Equal to `D3D12_TEXTURE_ADDRESS_MODE_MIRROR_ONCE`. |

|  |  |  |
|---|---|---|
|  | border<br>bordercolor<br>border_color | The border color is used when U, V, and W go outside of 0..1.<br>Equal to `D3D12_TEXTURE_ADDRESS_MODE_BORDER`. |
|  | no_border<br>nul_border | Nothing is considered to exist beyond the U, V, and W texture edges.<br>This is the default. |
| **-textureaddressingw**<br>**-taw** | Same as **-ta**. | Applies only to the U coordinate. |
| **-textureaddressingh**<br>**-tah** | Same as **-ta**. | Applies only to the V coordinate. |
| **-textureaddressingd**<br>**-tad** | Same as **-ta**. | Applies only to the W coordinate. |
| **-textureaddressingw_opaque**<br>**-taw_color** | Same as **-ta**. | Applies only to the U coordinate and to color channels. |
| **-textureaddressingh_opaque**<br>**-tah_color** | Same as **-ta**. | Applies only to the V coordinate and to color channels. |
| **-textureaddressingd_opaque**<br>**-tad_color** | Same as **-ta**. | Applies only to the W coordinate and to color channels. |
| **-textureaddressingw_alpha**<br>**-taw_alpha** | Same as **-ta**. | Applies only to the U coordinate and to the alpha channel. |
| **-textureaddressingh_alpha**<br>**-tah_alpha** | Same as **-ta**. | Applies only to the V coordinate and to the alpha channel. |
| **-textureaddressingd_alpha**<br>**-tad_alpha** | Same as **-ta**. | Applies only to the W coordinate and to the alpha channel. |
| **-border_color** | <r> <g> <b> <a> | Sets the border color for the U, V, and W coordinates.<br>Defaults to **0.0 0.0 0.0 1.0**. |

## YUV Options

| **Command** | **Parameter** | **Description** |
|---|---|---|
| **-yuv_input_format** | <any Vulkan, DXGI, or Metal | Sets the format (encoding) of the YUV file being loaded. |

| | | |
|---|---|---|
| | YUV format> | |
| | nv12 | DXGI_FORMAT_NV12/VK_FORMAT_G8_B8R8_2PLANE_420_UNORM |
| | nv21 | DXGI_FORMAT_NV21 |
| | yv12 | DXGI_FORMAT_YV12 |
| | yuy2 | DXGI_FORMAT_YUY2/VK_FORMAT_G8B8G8R8_422_UNORM/DXGI_FORMAT_G8R8_G8B8_UNORM/MTLPixelFormatGBGR422 |
| | uyvy | DXGI_FORMAT_R8G8_B8G8_UNORM/VK_FORMAT_B8G8R8G8_422_UNORM/MTLPixelFormatBGRG422 |
| | p010 | DXGI_FORMAT_P010/VK_FORMAT_G10X6_B10X6R10X6_2PLANE_420_UNORM_3PACK16 |
| | p016 | DXGI_FORMAT_P016/VK_FORMAT_G16_B16R16_2PLANE_420_UNORM |
| | p210 | DXGI_FORMAT_P210/VK_FORMAT_G10X6_B10X6R10X6_2PLANE_422_UNORM_3PACK16 |
| | p216 | DXGI_FORMAT_P216/VK_FORMAT_G16_B16R16_2PLANE_422_UNORM |
| | y210 | DXGI_FORMAT_Y210/VK_FORMAT_G10X6B10X6G10X6R10X6_422_UNORM_4PACK16 |
| | y216 | DXGI_FORMAT_Y216/VK_FORMAT_G16B16G16R16_422_UNORM |
| | y410 | DXGI_FORMAT_Y410 |
| | y416 | DXGI_FORMAT_Y416 |
| | ayuv | DXGI_FORMAT_AYUV |
| **-yuv_format** | Same as **-yuv_input_format**. | Sets the encoding when saving to a YUV format. |
| **-yuv_input_use_approx** | | Uses a common approximate YUV -> RGB conversion when loading a YUV file.<br>By default the full YUV -> RGB algorithm is used. |
| **-yuv_use_approx** | | Uses a common approximate RGB -> YUV conversion when writing to a YUV file.<br>By default the full RGB -> YUV algorithm is used. |
| | REC_709<br>REC709 | Sets the Kr and Kb factors according to the ITU-R Recommendation BT.709-5 standard. |

| | | |
|---|---|---|
| **-yuv_input_kr_kb** | | 0.212639005871510 and 0.072192315360734.<br>This is the default. |
| | REC_2020<br>REC2020 | Sets the Kr and Kb factors according to the ITU-R Recommendation BT.2020 standard.<br>0.2627 and 0.0593. |
| | SMPTC | Sets the Kr and Kb factors according to the SMPTE C standard.<br>0.2124 and 0.0866. |
| | REC_601<br>REC601 | Sets the Kr and Kb factors according to the ITU-R Recommendation BT.601 standard.<br>0.2988390 and 0.1143500. |
| | CIE_1931<br>CIE1931 | Sets the Kr and Kb factors according to the CIE 1931 standard.<br>0.3086 and 0.0820. |
| | NTSC_1953<br>NTSC1953 | Sets the Kr and Kb factors according to the NTSC 1953 standard.<br>0.3 and 0.11. |
| | EBU_TECH_3213<br>EBUTECH3213 | Sets the Kr and Kb factors according to the EBU Tech. 3213 standard.<br>0.2988390 and 0.1143500. |
| **-yuv_kr_kb** | Same as **-yuv_input_kr_kb**. | Sets the output Kr and Kb factors when saving to a YUV file. |
| **-yuv_input_set_kr_kb** | <kr><br><kb> | Manually specifies the Kr and Kb factors for loading a YUV file. |
| **-yuv_set_kr_kb** | <kr><br><kb> | Manually specifies the Kr and Kb factors for saving to a YUV file. |
| **-yuv_input_set_z**<br>**-yuv_input_set_black** | <black level> | Sets the black level (0..1) for loading a YUV file.<br>Defaults to **0.0**. |
| **-yuv_set_z**<br>**-yuv_set_black** | <black level> | Sets the black level (0..1) for saving to a YUV file. |
| **-yuv_input_set_s** | <scale> | Sets the scaler (0..1) for loading a YUV file.<br>Defaults to **1.0**. |

| | | |
|---|---|---|
| -yuv_input_set_scale | | |
| -yuv_set_s<br>-yuv_set_scale | <scale> | Sets the scaler (0..1) for saving to a YUV file.<br>Defaults to **1.0**. |
| -yuv_input_pc | | Sets the black level to 0.0, scale to 1.0, Kr and Kb to the ITU-R Recommendation BT.709-5 standard, and enables the full non-approximate conversion routine for loading YUV files. |
| -yuv_pc | | Sets the black level to 0.0, scale to 1.0, Kr and Kb to the ITU-R Recommendation BT.709-5 standard, and enables the full non-approximate conversion routine for saving to a YUV file. |
| -yuv_input_studio | | Sets the black level to (16.0 / 255.0), scale to (219.0 / 255.0), Kr and Kb to the ITU-R Recommendation BT.709-5 standard, and enables the full non-approximate conversion routine for loading YUV files. |
| -yuv_studio | | Sets the black level to (16.0 / 255.0), scale to (219.0 / 255.0), Kr and Kb to the ITU-R Recommendation BT.709-5 standard, and enables the full non-approximate conversion routine for saving to a YUV file. |

## Indices & Palettes

| Command | Parameter | Description |
|---|---|---|
| -gen_pal<br>-gen_palette | | Generates a new palette for indexed images. |
| -gen_pal_iterations | <iterations> | Sets the maximum number of iterations when generating a palette.  The higher the better.<br>Defaults to the number of colors in the palette.  In practice, the number of colors in a palette serves as a hard maximum, but it will exit early if iterations stop causing refinements. |
| -pal_dither | floyd<br>floyd-steinburg | Selects the type of dithering to use for palettes.<br>Implements Floyd-Steinburg Dithering.<br>This is the default. |
| | jjn | Implements Jarvis, Judice, and Ninke Dithering. |

| | | |
|---|---|---|
| | stucki | Implements Stucki Dithering, presented 5 years after JJN, offering a similar dither with a slight performance improvement. |
| | burkes | Implements Burkes Dithering, presented 7years after Stucki, offering yet-another minor hit to quality in exchange for performance. |
| | sierra | Implements Sierra Dithering. |
| | sierra2row | Implements Two-Row Sierra Dithering. |
| | sierralite<br>sierra_lite | Implements Sierra Lite Dithering. |
| | atkinson<br>atk | Implements Atkinson Dithering, used by the original Macintosh computer. |
| | bayer4<br>bayer4x4 | Implements a 4×4 Bayer Dither. |
| | bayer8<br>bayer8x8 | Implements an 8×8 Bayer Dither. |
| **-dither_error_weight**<br>**-dither_error_weights** | <r> <g> <b> <a> | Sets the per-channel weights for dithering.<br>Defaults to **0.925 0.925 0.925 1.0**. |
| **-dither_error_weight_full**<br>**-dither_error_weight_100** | | Sets the dithering weights to 1.0 1.0 1.0 1.0. |
| **-dither_error_weight_75** | | Sets the dithering weights to 0.75 0.75 0.75 1.0. |
| **-dither_error_weight_half**<br>**-dither_error_weight_50** | | Sets the dithering weights to 0.5 0.5 0.5 1.0. |
| **-dither_error_weight_25** | | Sets the dithering weights to 0.25 0.25 0.25 1.0. |
| **-dither_error_weight_perceptual**<br>**-dither_error_weight_perc** | REC_709<br>REC709 | Sets the dithering weights to 0.212639005871510 0.715168678767756 0.072192315360734. |
| | REC_2020<br>REC2020 | Sets the dithering weights to 0.2627 0.678 0.0593. |
| | SMPTC | Sets the dithering weights to 0.2124 0.7011 0.0866. |
| | REC_601<br>REC601 | Sets the dithering weights to 0.2988390 0.5868110 0.1143500. |

| | | |
|---|---|---|
| | EBU_TECH_3213<br>EBUTECH3213 | |
| | CIE_1931<br>CIE1931 | Sets the dithering weights to 0.3086 0.6094 0.0820. |
| | NTSC_1953<br>NTSC1953 | Sets the dithering weights to 0.3 0.59 0.11. |
| **-dither_error_weight_scale** | <scale> | Scales the dithering weights by the given amount. Can be used to strengthen or soften the effects of perceptual weights. Weights are multiplied each time this command is encountered. |

## Mipmaps

| Command | Parameter | Description |
|---|---|---|
| **-nomips**<br>**-nomipmaps**<br>**-no_mips**<br>**-no_mipmaps** | | No mipmaps are generated and existing mipmaps are discarded. |
| **-nmips** | <total mipmaps> | Sets the total number of mipmaps desired. Set to 0 to generate (or keep) a full mipmap chain.<br>Defaults to **0**. |
| **-keepmips**<br>**-keepmipmaps**<br>**-keep_mips**<br>**-keep_mipmaps** | | By default, new mipmaps will be generated as specified by **-nmips** (which defaults to **0**, so a full chain is generated).<br>This command allows existing mipmaps to be retained instead of overwritten. Existing mipmaps beyond the **-nmips** specification will be truncated, and if **-nmips** extends beyond the number of existing mipmaps then new mipmaps will be generated to fill the gap. |
| **-mip_filter** | Same as for **-filter**. | Applies the selected filter to all mipmap filters. |
| **-mip_filterw** | Same as for **-filter**. | Sets the mipmap color and alpha *width* filter. |
| **-mip_filterh** | Same as for **-filter**. | Sets the mipmap color and alpha *height* filter. |
| **-mip_filterd** | Same as for **-filter**. | Sets the mipmap color and alpha *depth* filter. |
| **-mip_filterw_color** | Same as for **-filter**. | Sets the mipmap color *width* filter. |
| **-mip_filterh_color** | Same as for **-filter**. | Sets the mipmap color *height* filter. |
| **-mip_filterd_color** | Same as for **-filter**. | Sets the mipmap color *depth* filter. |

| | | |
|---|---|---|
| **-mip_filterw_alpha** | Same as for **-filter**. | Sets the mipmap alpha *width* filter. |
| **-mip_filterh_alpha** | Same as for **-filter**. | Sets the mipmap alpha *height* filter. |
| **-mip_filterd_alpha** | Same as for **-filter**. | Sets the mipmap alpha *depth* filter. |

## Normal Maps

| Command | Parameter | Description |
|---|---|---|
| **-nm_channel** | r<br>red | The normal map will be generated using the R channel. |
| | g<br>green | The normal map will be generated using the G channel. |
| | b<br>blue | The normal map will be generated using the B channel. |
| | a<br>alpha | The normal map will be generated using the A channel. |
| | max | The normal map will be generated using the max value between the RGBA channels.<br>This is the default. |
| | rgb | The normal map will be generated using the average value between the RGB channels. |
| | colorspace | The normal map will be generated using the weighted average value between the RGB channels.<br>Use **-luma** to select from predefined weights or **–weight** to manually specify weights. |
| **-norm**<br>**-normalize** | | Indicates that the normal maps should be normalized. |
| **-opengl**<br>**-unity**<br>**-blender**<br>**-maya** | | Specifies that the normal map should be compatible with OpenGL. |
| **-directx**<br>**-ue4**<br>**-unreal**<br>**-unrealengine**<br>**-ue**<br>**-dsmax** | | Specifies that the normal map should be compatible with DirectX. |
| **-n3x3** | | Normal maps will be generated with a 3×3 kernel. |
| **-n5x5** | | Normal maps will be generated with a 5×5 kernel. |
| **-n7x7** | | Normal maps will be generated with a 7×7 kernel. |

|  |  |  |
|---|---|---|
| **-n9x9** |  | Normal maps will be generated with a 9×9 kernel. |
| **-scale**<br>**-nm_z** |  | Specifies the normal map’s Z influence. |

## Transforms

| Command | Parameter | Description |
|---|---|---|
| **-format** | <any Vulkan, DXGI, or Metal format> | Converts the loaded image to the given texture format. |
| **-ogl_format** | <internal format> <type> <base internal format> | Converts the loaded image to the given OpenGL texture format. |
| **-ignore_alpha** |  | Any alpha channel is set to all 1’s. |
| **-alpha_threshold** | <cutoff> | Sets the alpha cutoff ([0..255]) for conversions to formats with binary alpha. Defaults to **128**. |
| **-premultiply_alpha**<br>**-premult_alpha** |  | Specifies that alpha should be pre-multiplied.<br>If an image is already pre-multiplied it is not pre-multiplied again. |
| **-swizzle** | <swizzle> | Specifies a swizzle to apply. Valid swizzle characters: rgbaxyzw01. Must be 4 characters long and is not case-sensitive. |
| **-swap** |  | Swaps the R and B channels. |

## Quality Settings

| Command | Parameter | Description |
|---|---|---|
| **-quality_highest**<br>**-very_slow** |  | The highest quality setting for compressing textures in BC*, EAC, ETC*, PVR, and ASTC formats. |
| **-quality_production**<br>**-slow** |  | The 2nd-highest quality setting for compressing textures in BC*, EAC, ETC*, PVR, and ASTC formats. |
| **-quality_normal**<br>**-basic** |  | The normal quality setting for compressing textures in BC*, EAC, ETC*, PVR, and ASTC formats. |
| **-fast** |  | A fast but somewhat low-quality setting for compressing textures in BC*, EAC, ETC*, PVR, and ASTC formats. |

| | | |
|---|---|---|
| **-quick**<br>**-veryfast** | | A faster but lower-quality setting for compressing textures in BC*, EAC, ETC*, PVR, and ASTC formats. |
| **-ultrafast** | | The fastest but lowest-quality setting for compressing textures in BC*, EAC, ETC*, PVR, and ASTC formats. |

## Misc.

| Command | Parameter | Description |
|---|---|---|
| **-weight**<br>**-weights** | <red weight> <green weight> <blue weight> | Sets the luminance weight factors manually. |
| **-luma** | REC_709<br>REC709 | Sets the luminance weight factors according to the ITU-R Recommendation BT.709-5 standard.<br>0.212639005871510, 0.715168678767756, and 0.072192315360734.<br>This is the default. |
| | REC_2020<br>REC2020 | Sets the luminance weight factors according to the ITU-R Recommendation BT.2020 standard.<br>0.2627, 0.678, and 0.0593. |
| | SMPTC | Sets the luminance weight factors according to the SMPTE C standard.<br>0.2124, 0.7011, and 0.0866. |
| | REC_601<br>REC601 | Sets the luminance weight factors according to the ITU-R Recommendation BT.601 standard.<br>0.2988390, 0.5868110, and 0.1143500. |
| | CIE_1931<br>CIE1931 | Sets the luminance weight factors according to the CIE 1931 standard.<br>0.3086, 0.6094, and 0.0820. |
| | NTSC_1953<br>NTSC1953 | Sets the luminance weight factors according to the NTSC 1953 standard.<br>0.3, 0.59, and 0.11. |
| | EBU_TECH_3213<br>EBUTECH3213 | Sets the luminance weight factors according to the EBU Tech. 3213 standard.<br>0.2988390, 0.5868110, and 0.1143500. |
| **-printformats**<br>**-print_formats** | | Prints all supported formats that can be supplied to **-format**. |

## PNG Options

| Command | Parameter | Description |
|---|---|---|
| **-png_default** | | Default PNG compression (6) will be used. |
| **-png_bestspeed** | | Fast PNG compression (1) will be used. |
| **-png_bestcompression** | | Best PNG compression (9) will be used. |
| **-png_level** | | Specifies the PNG compression level.<br>[0..9]. |
| **-png_nocompression** | | No PNG compression will be used. |
| **-png_interlaced** | | Interlacing will be used. The default is no interlacing. |
| **-png_format** | R8G8B8<br>RGB24<br>RGB | Specifies the PNG format to which to save.<br>If the format is not specified, the closest format to what was specified by **-format** (or the original file's format if **-format** is not specified) will be used. |
| | R8G8B8_SRGB<br>RGB24_SRGB<br>RGB_SRGB | |
| | R8G8B8A8<br>RGBA32<br>RGBA | |
| | R8G8B8A8_SRGB<br>RGBA32_SRGB<br>RGBA_SRGB | |
| | R16G16B16<br>RGB16 | |
| | R16G16B16A16<br>RGBA16 | |
| | L8<br>LUMINANCE8<br>GREYSCALE8<br>GRAYSCALE8 | |

| | | |
|---|---|---|
| | L16<br>LUMINANCE16<br>GREYSCALE16<br>GRAYSCALE16 | |
| | I1<br>INDEXED1 | |
| | I2<br>INDEXED2 | |
| | I4<br>INDEXED4 | |
| | I8<br>INDEXED8 | |

## BMP Options

| Command | Parameter | Description |
|---|---|---|
| **-bmp_rle** | | RLE encoding will be used to save the BMP file.<br>This is not set by default. |
| **-bmp_noalpha** | | Alpha will not be saved or will be set to 1 in the BMP file.<br>The default is to store alpha when available. |
| **-bmp_nobitmask**<br>**-bmp_nomask** | | By default, the BMP file will contain masks for the R, G, B, and A channels.<br>This setting causes the BMP file to be saved without the masks.<br>See Notes. |
| **-bmp_format** | R8G8B8<br>RGB24<br>RGB | Specifies the BMP format to which to save.<br>See Notes.<br>If the format is not specified, the closest format to what was specified by **-format** (or the original file's format if **-format** is not specified) will be used. |
| | R8G8B8_SRGB<br>RGB24_SRGB<br>RGB_SRGB | |

| | | |
|---|---|---|
| | R8G8B8A8<br>RGBA32<br>RGBA | |
| | R8G8B8A8_SRGB<br>RGBA32_SRGB<br>RGBA_SRGB | |
| | B8G8R8A8 | |
| | B8G8R8A8_SRGB | |
| | A8B8G8R8 | |
| | A8B8G8R8_SRGB | |
| | R4G4B4A4 | |
| | B4G4R4A4 | |
| | A4R4G4B4 | |
| | A4B4G4R4 | |
| | R5G6B5 | |
| | B5G6R5 | |
| | R5G5B5A1 | |
| | A1B5G5R5 | |
| | A1R5G5B5 | |
| | A4B4G4R4 | |

## TGA Options

| Command | Parameter | Description |
|---|---|---|
| **-tga_rle** | | The TGA file will be saved with RLE encoding. |
| **-tga_format** | R8G8B8<br>RGB24<br>RGB | Specifies the TGA format to which to save.<br>If the format is not specified, the closest format to what was specified by **-format** (or the original file's format if **-format** is not specified) will be used. |
| | R8G8B8_SRGB<br>RGB24_SRGB<br>RGB_SRGB | |
| | R8G8B8A8<br>RGBA32<br>RGBA | |
| | R8G8B8A8_SRGB<br>RGBA32_SRGB<br>RGBA_SRGB | |

| | | |
|---|---|---|
| | A1R5G5B5<br>A1RGB5 | |
| | L8<br>LUMINANCE8<br>GREYSCALE8<br>GRAYSCALE8 | |

# Notes

## Gamma

- Positive values use a raw pow(1/ γ)-based gamma curve.  Negative values are divided into 2 halves: If γ is <= -1.0, an XtoLinear transform is applied; if -1.0 < γ < 0.0, a LinearToX transform is applied, where X is the curve specified by one of the standards (sRGB, ITU-R Recommendation BT.709-5, etc.)  Generally, gamma will be above 1.0 or below -1.0.
- The default standard curve is sRGB Precise, so a default value of -2.2 results in a precise sRGB -> Linear transform.
- The relationship is reversed for target gamma.  A positive value results in a pow( γ) transform being applied, while if γ is <= -1.0, a LinearToX transform is applied; if -1.0 < γ < 0.0, an XtoLinear transform is applied.
- This means that when considering gamma, you specify what the input is and what the target should be.  By knowing what the input gamma is, the reverse transform can be applied to put the image back into linear space, and by knowing what the target gamma should be, a proper transform from linear to the target gamma can be made.

## Texture Addressing

- Texture addressing is used during resampling and normal-map creation.
- For standard image resampling, **nul_border** is appropriate, as it will only sample from in-image texels, meaning no influence from outside 0..1 at all, and edge texels won’t have an abnormally large influence as they would with **clamp**.  For in-game textures, select the addressing mode that matches how it will be addressed in the game.

## YUV

- By default, YUV <-> RGB algorithms that take additional parameters (Kr, Kb, Z, and B) are used. These provide reliable conversions, but many implementations use an approximation for these conversions, which can result in slight adjustments to the colors. Switching to the approximate conversion would be appropriate if you are dealing with a YUV file that has been adjusted to account for this color shift.
- The approximate YUV algorithm does not use Kr, Kb, Z, or B.
- YUV files that contain multiple frames are loaded as 3-D volume textures, with each slice of the depth component being a frame. You can resample the depth component to change the number of frames in the animation. This effectively smoothly speeds up or slows down the animation. Because this is a resampling through time rather than over color frequencies, a linear filter is most appropriate unless a specific visual effect is desired.

## BMP

- Bit masks aren’t used when saving a file as RLE.
- Some packed formats, such as A4R4G4B4, will only retain their component orders if bit masks are used. Without bit masks, the saved BMP file may have swizzled the components (for example to R4G4B4A4).