NAME
PTStitcherNG – merge images to panoramas

SYNOPSIS
PTStitcherNG *[options] projectfile*

DESCRIPTION
PTStitcherNG reads and transforms any number and type of input images, and combines them into one seamless panoramic image with larger field-of-view. Given enough input images, full spherical 360x180 degree views can be synthesized suitable for virtual-reality viewers or printing. Transformation parameters and names of the input images are read from a plain-text projectfile. The transformations consist of correcting distortions due to camera lenses, perspective tranlation and remapping to any of the usual panorama projections. Merging employs an eight level multiresolution algorithm which hides seams even if the source images fit badly. PTStitcherNG natively reads PPM, TIFF and JPEG images, and almost any raw or other format through plug-ins. The tight coupling of remapper and merger together with the utilization of processor resources (SIMD, multi-core cpus, CUDA) make PTStitcherNG fast and suitable for real-time processing.

Options:

–o *file*   Write panoramic image to the specified *file* (mandatory). If *file* ends with .tif or .jpg these formats are used irrespective of the setting in the scriptfile. Otherwise an extension is added depending on the selected format. If set to "-", the panorama is written to stdout. This forces image format to *ppm* overwriting any script settings, and enables option "-d" (deferred merging, to prevent seek operations in the the output stream). A pipe may thus be setup to convert the panorama to any format.

–i *file*   Same as option –o but switch to inverse mode: The panoramic image *file* must exist, and is used as input. The partial images specified in the projectfile (usually the source images) are extracted from the panorama and written to disk. Caution: if this mode is used with existing panorama projects, all source files will be overwritten.

–f *file*   Projectfile is specified here or as first non-option argument.

–j *num*   PTStitcherNG creates *num* threads for parallel processing. By default, the actual number of cores on the host as returned by *sysconf* is used.

–z *angle*

–n *angle*
          Blend across zenith and/or nadir (only applicable for 360x180 degree equirectangular panoramas). Intermediate polar images (field-of-view: *2 x angle*) are created and blended, and inserted into the panorama afterwards. This avoids artefacts due to the ambiguity of the polar region in equirectangular format, especially if images overlap there. Default *angle* is 0 degrees (no blending across zenith/nadir).

–r *file*   Full path to a plain-text file listing optional image readers which are used by PTStitcherNG to handle unusual fileformats. Any application which writes PPM-format to standard output is suitable as plug-in. Recommended is the utility *dcraw* for reading raw formats of almost any digital camera, and the *netpbm* suite of tools. By default, a file named *PTImageReader.txt* in PTStitcherNG's directory is used, please open it for more info.

–p   Set output pixel size to 16 bits per sample (48 bit RGB). Default is 8 bit (24 bit RGB). This option only works for TIFF and PPM output.

–c   Run as console application without displaying progress report and message boxes (only Windows).

–s *num*   Specify 0 if the input images should be merged exactly in the order they are listed in the projectfile. By default (1) PTStitcherNG optimizes the sequence of images for maximum speed and

minimum memory requirements.

−u   Use the host ram for temporary data instead of ram on the graphics card. This option should be used on CUDA-platforms if the graphics card has very little ram. It has no effect on non-CUDA platforms.

−x *num* PTStitcherNG performs a forward transformation of each image to determine its destination position in the final panorama. This is used to speed up the subsequent reverse transformation which creates the actual panoramic image. Specifying *0* transforms all pixels, while specifying *1* transforms just the outer frame of the input image. This latter method may cause errors if there are wrapping input images. *2* no forward transformation; use the entire panorama area as destination rectangle. By default, PTStitcherNG chooses *0* if the field of view of the panorama is 360 degrees, *1* else, except for non invertible transformations (multiple rectilinear projection, format *f11*) which use *2*.

−l *num* Set maximum size of pyramid for blending. PTStitcherNG creates *num* resized versions of each remapped image, which are blended and fused into the final panorama. Maximum *num-* value is 10, default is 8. The width of the blending region between any two source images extends up to 2^{num} pixels if the overlap permits.

−m *num*
         Set minimum pyramid level for blending. This is similar to applying a feather tool with width 2^{num} pixels to the blending masks. This option is used to hide mismatches in blended images (default 3).

−b *num* specify seam placement: *0* : choose the center of the overlap region (default). *1* : optimize depending on content. Experimental option, not available on the Cuda platform. *2* : no seam, images are blended with equal weight. This is a diagnostic format used for seam editing.

−a   switch to colour mode. Each source image is replaced by a unique color. This is a diagnostic format used for seam editing.

−e *mask*
         Load image named *mask* which is used as source for seam placement. This image must use the same colors as generated by the *−a* option. Seam editing is performed by first generating a mask image, then editing this image in any graphics program, and finally using it as input mask for stitching. Colors are used as index to images. Therefore, lossless image formats (tiff) should be used.

−d   defer merging. Effective only for flat tiff and ppm output. Transformed images are merged after all other images are transformed. This mode should be used on platforms which cannot seek well into empty files. Usually, tranformations should be faster without this option, test for yourself. Using this option consumes much memory.

−q   Be quiet, don't display diagnostics and timing information.

−h   Print help message and exit.

FILES

The plain-text file *PTImageReader.txt* is used to register input filters, e.g. for unusual file formats or preprocessing. See the option *r* above and the supplied version of this file for examples. A *projectfile* is required as input. A subset of traditional PTStitcher options and PTGui *pts-* commands are supported, see the file *PTStitcherNG.html* for details.