MPRemap v.0.2a Remap From and To Motion Pictures

Helmut Dersch derfh-furtwangen.de

December 9, 2007

Summary

With MPRemap motion pictures can be remapped from and to almost any geometric projection, or created from static panoramic images. Images made with fisheye lenses can be "rectified" to wide-angle rectilinear or cylindrical formats. Arbitrary partial views can be extracted from wide angle images. Lens distortion corrections and perspective changes can be accomplished. Any transformation supported by the underlying Panorama-Tools-library may be applied.

Transformations use either global, constant parameters, or per-frame specifications. The latter method allows the user to generate virtual moving camera positons (pan, tilt and zoom) from static source material. An optional motion blur feature leads to realistic images for fast movements. See the examples-section of this manual for tutorials. A few examples can be viewed at my website www.fh-furtwangen.de/~dersch.

MPRemap is distributed as binary for the Windows operating system. Binaries for Linux and other Unix-like operating-systems can be created using the supplied sources.

MPRemap is distributed under the terms of the GNU General Public License.

1 Installation

Version 0.2a of MPRemap consists of a Java-program which serves as a frontend and user-interface to the underlying helper applications mpremap.exe and

ffmpeg.exe. mpremap.exe is used to convert individual frames using the Panotools-library. ffmpeg.exe breaks up almost any video stream into individual frames and reassembles the converted frames into video streams. To use MPRemap on the Windows-platform follow these installation instructions:

- Get and install the ffmpeg application. A Google-search for ffmpeg and your OS-name should bring up many download-links.
- Install a recent Java runtime (Sun JRE version $\geq 1.4.2$).
- Unzip MPRemap.zip and put the content somewhere on your computer.
 Running from non-writeable media (CD/DVD) is ok as long as the current working directory is writeable.
- Depending on your installation, start MPRemap by either double-clicking the file MPRemap.jar or issuing the command java -jar MPRemap.jar.

On all other platforms you first have to build the mpremap-application. See the section Sources for details. All other steps are equal.

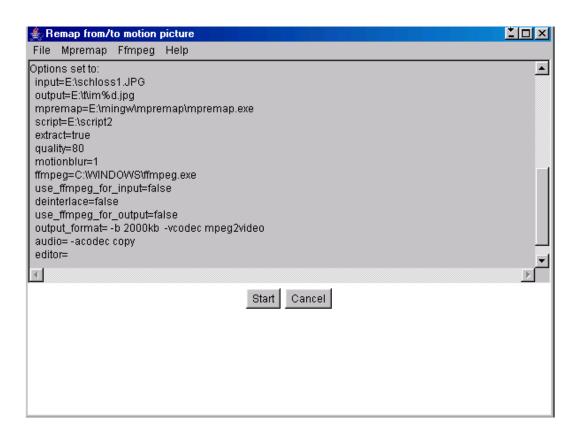
2 Usage

MPRemaps main window exhibits a menu bar and a text area displaying status and help messages. Upon startup a copyright message and values for all settable options are displayed (see figure 2).

Many options have to be set just once like the path to the helper applications ffmpeg.exe (Ffmpeg \rightarrow Path to ffmpeg) and mpremap.exe (MPRemap \rightarrow Path to mpremap). All options together with their values are displayed by selecting the menu-item Help \rightarrow About.

Each menu-item is explained in the online-help accessible through the Help-menu. This menu also provides manuals for the helper applications. In this article we will only present a few examples included in the distribution which may serve as starting points for your own projects and which tests your installation. Key to all tranformations is a corresponding scriptfile which specifies the details. Please consult the online-help for details and further options. The source media files and

Figure 1: MPRemap main window displaying status information



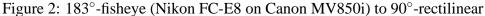
the script files can be downloaded from my website

<http://www.hs-furtwangen.de/~dersch>.

Conversion starts after pressing the 'Start'-button. The status window displays messages from MPRemap and the helper applications. Conversion is finished when the message 'Conversion finished" is displayed. Canversion may be interrupted by pressing 'Cancel'.

3 Examples

3.1 Fisheye to Rectilinear/Cylinder







The video is made with a Camcorder Canon MV850i and a Nikon Fisheye lens FC-E8 (left image in figure 3.1). To create a rectilinear view like the image to the right, use this script

```
p f0 w640 h480 v160 o f10 r-8 v173
```

and set the options:

```
input=...fisheye.mpg
output=...ar.mpg
```

```
script=...script_fisheye2rectilinear
extract=false
motionblur=1
use_ffmpeg_for_input=true
deinterlace=false
use_ffmpeg_for_output=true
output_format= -b 2000kb -vcodec mpeglvideo
audio= -acodec copy
```

Figure 3: 183°-fisheye (Nikon FC-E1 on Canon MV850i) to 160°-cylinder



To remap the fisheye view to cylindrical projection like the image to the right (figure 3.1, use this script

```
p f1 w640 h480 v160 o f10 r-8 v173
```

and set the options:

```
input=...fisheye.mpg
output=...ac.mpg
script=...script_fisheye2cylinder
extract=false
motionblur=1
```

```
use_ffmpeg_for_input=true
deinterlace=false
use_ffmpeg_for_output=true
output_format= -b 2000kb -vcodec mpeglvideo
audio= -acodec copy
```

Notice that the scripts also rotate the image by 8° to get a straight horizon.

3.2 Perspective Correction



Figure 4: Correcting falling lines and barrel distortion

The video is made with a Canon Powershot TX1 and MPRemap is used to perform a perspective correction, i.e. to straighten vertical lines, and correct lens distortions.

To repeat the conversion use this script

```
p f0 v60 w823 h1029 C84,724,127,607
```

o f0 v50 y0 p10 r2 b-0.01

and these options

```
input=...perspect.mpg
output=....ap.mpg
```

```
script=...script_perspect_movie
extract=false
quality=82
motionblur=1
use_ffmpeg_for_input=true
deinterlace=false
use_ffmpeg_for_output=true
output_format= -b 2000kb -vcodec mpeglvideo
audio= -acodec copy
```

Notice that this script creates large target frames (823x1029 pixels) and later crops these to 640x480. This simulates the action of a shift lens.

3.3 Panorama to Grid

In this example MPRemap creates a series of prerendered views from a static panoramic image. These views are equally spaced regarding to pan and tilt angles, and can be used together with an object-viewer (eg PTViewerJS) to view interactive panoramas on limited devices.

To repeat the conversion use this script. Make sure the target directory for the jpeg-encoded images exists.

```
p f2 v360

GRID(-90,90,15,0,355,20)
o f0 w180 h120 v75 r0 y0 p0
  and these options
  input=...panorama.jpg
  output=...target/im%d.jpg
  script=...script_panorama2grid
  extract=true
  quality=82
  motionblur=1
  use_ffmpeg_for_input=false
  use_ffmpeg_for_output=false
```

Figure 5: Creating equally spaced views from a panoramic image

3.4 Panorama to Movie

MPRemap can be used to create movies from static panoramic images. Arbitrary many pan/tilt/rotation/zoomlevels are smoothly interconnected by a spline function to simulate a real panning camera. Smooth start and stop actions provide a realistic viewing experience, which can be further enhanced by adding motion blur to fast movements or zooming. These are a few example frames from a virtual pan.

To repeat the conversion use this script

```
p f2 v360

o f0 w320 h240 y70 p0 r0 v90
SPLINE(100,100,30,30)
o f0 w320 h240 y200 p20 r10 v120 nsame
o f0 w320 h240 y360 p0 r0 v60 nsame
o f0 w320 h240 y360 p0 r180 v120 nsame
o f0 w320 h240 y360 p0 r360 v60 nsame
o f0 w320 h240 y360 p0 r360 v60 nsame
and these options
input=...panorama.jpg
output=...am.mpg
script=...script_panorama2movie
extract=true
motionblur=10
use_ffmpeg_for_input=false
use_ffmpeg_for_output=true
```

output format = -b 2000kb -vcodec mpeglvideo



3.5 How to find suitable parameters

In the case of video input, use single frames to experiment and find suitable parameters. This command may be used to extract a single frame from a movie:

In the case of panoramic input, use extract-mode. Then you can use PTViewer to select suitable viewing parameters which PTViewer displays after pressing 'v'.

Otherwise, choose one of the example scripts as starting point, keep the script open in a plain-texteditor, and make changes to the parameter. See the online help in MPRemap for the meaning of each option. Then create a single target frame using MPRemap and judge the result using an imageviewer. Cycle through these steps until you are satisfied with the result, and then convert the whole movie.

4 Sources

The distribution contains the C-sources for the mpremap helper application. To build the application, you also need the Panorama Tools library pano12 (e.g. from Sourceforge) and the dependencies of this library (libjpeg, libpng, libtiff). The Makefile has targets for Windows (Mingw), X86-Linux and PPC-Linux. Other targets can be easily generated.