Documentation

Automatic generation of perspective stereo pairs from half equirectangular images

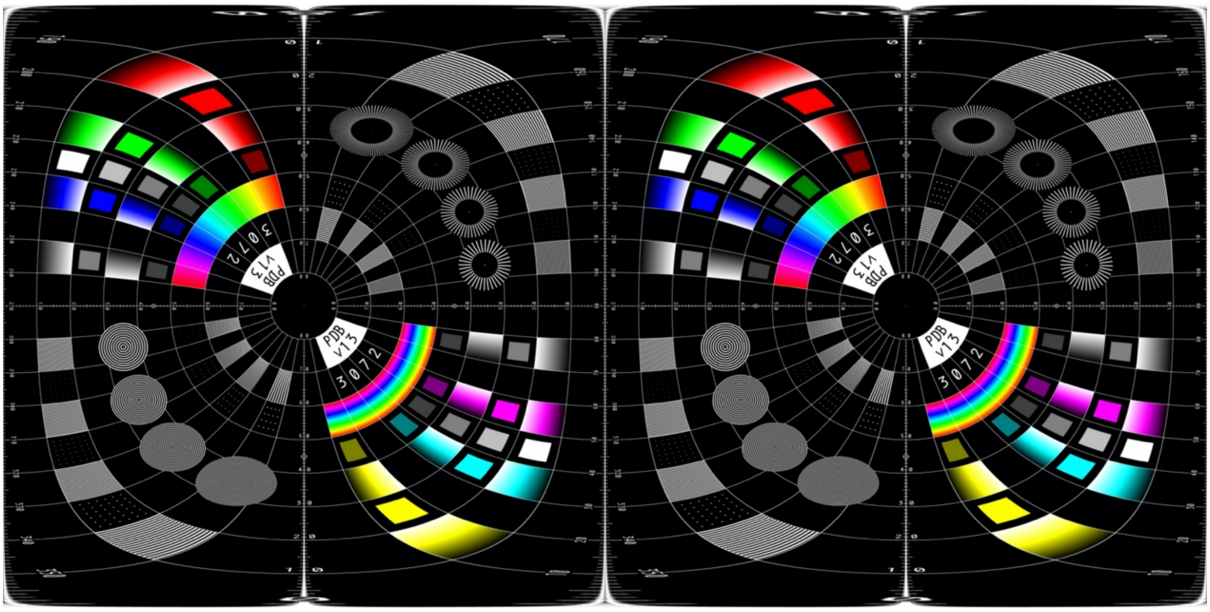
Paul Bourke

7 February 2021

Introduction

In the following a procedure is described for taking the stereoscopic videos from the Insta360 Evo camera when in the VR180 configuration and converting those into stereoscopic perspective videos. Since the Evo VR180 camera captures a 180 degree field of view, the perspective representation will necessarily be of a narrower field of view.

The Evo camera consists of two fisheye lenses, but the device converts these each into a 180degree by 180degree equirectangular projection and places the left and right eye side by side in a 5760x2880 frame. A fisheye grid would look like the following.



The task then is to apply an equirectangular to perspective projection, once for each left-right eye half of the image from the camera. This is performed by the authors sphere2fish software, more details can be found here

http://paulbourke.net/dome/2fish/

This utility allows one to totally control the perspective view extracted. Specifically it allows one to adjust the horizontal field of view, the pan and tilt angles, the aspect ratio and the lens offset. The sphere2fish software only converts one image, but it also allows one to create the remap filter files for ffmpeg thus allowing one to convert movies.

Remap files

The reader is directed to the ffmpeg manual for details on the remap filter, see the following

https://trac.ffmpeg.org/wiki/RemapFilter

In essence the remap files inform ffmpeg what pixel to use from the source image (or movie frame) for each pixel in the destination image (or movie frame). For a single remap filter conversion two files are required, one that specifies the source horizontal coordinate for each output pixel and another that specifies the source vertical coordinates for each output pixel. These are normally referred to as the “x” and “y” remap files. These two files are encoded in the PGM image format, a rather antiquated format but simple and adequate for the task at hand.

The main limitation is that a remap file are fixed rather than parameterised in any way, that is, a different pair of remap files are required for each remapping variation.

For the exercise here the following variations have been created

- 4:3 and 16:9 aspect ratio.
- Perspective tilt angles of 0, 15 and 25 degrees. This allows one to orientate the (virtual) perspective camera at different vertical angles to center on the key subject matter.
- Perspective horizontal offsets of 0 and 5 degrees. This allows one to control the zero parallax distance, a value of 0 places the zero parallax at infinity, 5 (which corresponds to a shift of 5% of the image width) will place the zero parallax distance closer to the camera. This is an important control for easy to view screen based stereoscopy.

The set above consists of 2 * 3 * 2 remap variations, each of which consists of 2 pgm files (x and y) and one for each left and right eye. This is a total of 48 pgm files.

A naming convention was chosen by using the letters “o”, “tilt”, “43” or “169”, “left” or “right” and “x” and “y”. For example, the following 4 PGM files

o5_tilt25_169_leftx.pgm
o5_tilt25_169_rightx.pgm

o5_tilt25_169_lefty.pgm

o5_tilt25_169_righty.pgm

would be required to remap a stereoscopic movie to a perspective movie with a 5% horizontal lens offset, a 25 degree tilt and a 16:9 aspect ratio.

Note the remap files are also dependent on the input image or movie width, 5760x2880 is assumed here.

Specifics

A command line script called “convertmovie” is provided that will take an input movie from the Insta360 Evo and create the left and right eye movie, a side-by-side version at full resolution, and a side-by-side version at half the horizontal resolution. All 4 of these outputs are created using ffmpeg. The reader is expected to inspect and understand how this script works.

It should be called with two command line parameters, the first is the prefix of the movie filename to be converted and the second is the first portion of the pgm file.

For example, if the input movie is called sample.mp4 and one wishes to transform it using the pgm files above (5% lens offset, 25 degree tilt and 16:9 aspect) one would use the command line

```
convertmovie sample o5_tilt25_169
```

There are some assumptions made here as to where the script, the ffmpeg program, the movie file and pgm files are stored. I will assume ffmpeg is on the general command line application path. If the “convertmovie” script is also not on the command path then one might need to run as

```
./convertmovie sample o5_tilt25_169
```

if the “convertmovie script is in the current directory or

```
/pathtoconvertmovie/convertmovie sample o5_tilt25_169
```

Similarly one might need to specific the paths to the movie and the pgm files.