GROUP 1

Members:

Bárbara Darques Barros - 7243081
Juliana de Mello Crivelli - 8909303
Kaue Ueda Silveira - 7987498

Title: Conversion of static images into parallax animations

Abstract: Starting with a static image we try to retrieve objects at the same depth and separate them into images, each of them representing a different plane from the scene. Then we interpolate the blank spaces caused by the separation, extending the background. In order to produce the parallax effect we overlap the extended planes with slightly different displacements.

Roadmap:

1) Identify and separate planes
   We intend to detect regions applying Region Adjacency Graph (RAG) Merging over the original image. Each region will be put in a separate image, surrounded with transparent pixels.
2) Interpolate the retrieved planes
   Following the professor suggestion, we intend to use the *Papoulis-Gerchberg* in order to fill the areas covered by the foreground planes, avoiding blanks spaces to be displayed to the viewer.
3) Overlap new images with different relative positions
   The extended planes generated in the previous steps will be overlapped with a variable position difference, generating various frames that once put together simulate the relative movement of the scene objects.
4) Generate gif
   We'll use the OpenCV and Matplotlib modules in order to convert the output frames into a gif animation, making use of the function `matplotlib.animation.ArtistAnimation`.

Input images:

The images chosen for input are landscape photos taken by the group students, such as Figure 1, and can be found at the following link: https://drive.google.com/drive/folders/0BxHj0zhF4J8-TXZYMmp3VDNXbW8?usp=sharing.
We selected images with big contiguous regions, like lakes or forests with small variation in color and details.

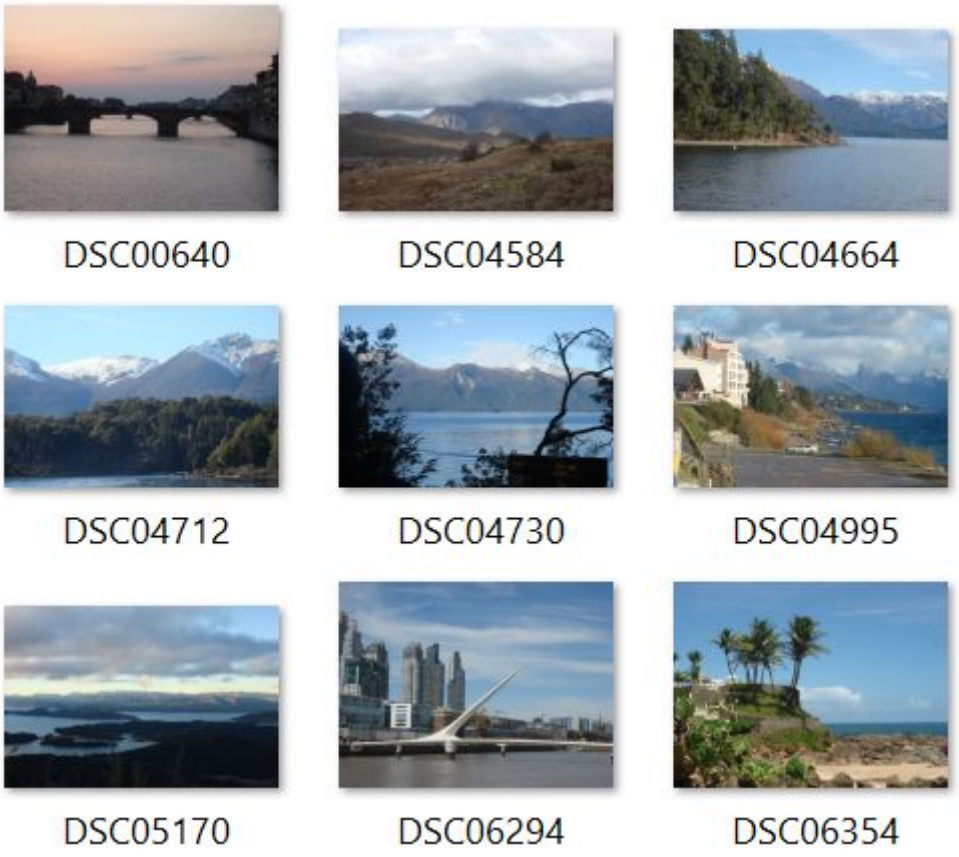

Figure 1: input image samples

First results:

We abandoned the original idea of using Fourier Transform, since it is best suited for black and white images. After several tests, such as watershed transform, a graph-based method proposed by Felzenszwalb & Huttenlocher and Random Walker algorithm (Figure 3), and Otsu-thresholds we were able to obtain a result similar to the desired one, using RAG Merging (Figure 2).

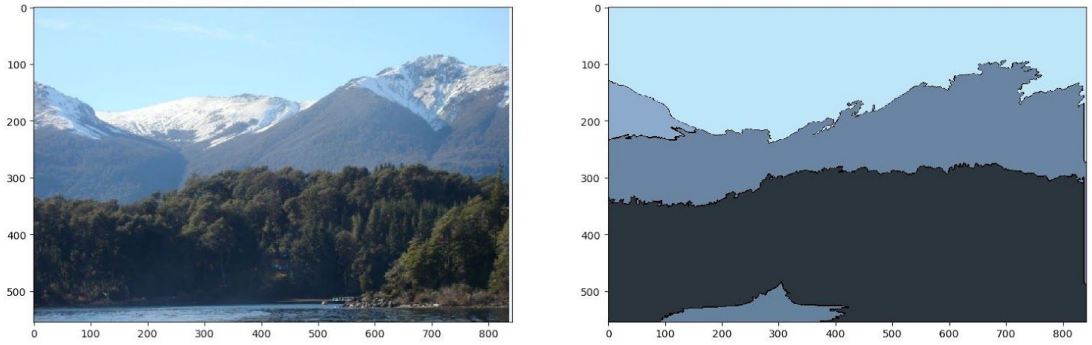

Figure 2: Image segmentated with RAG Merging

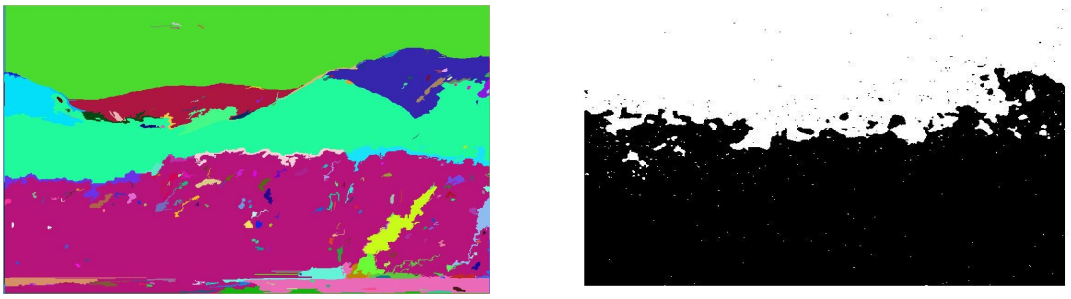

Figure 3: Images segmentated with the Felzenszwalb & Huttenlocher and the Random Walker algorithms

Code:

https://github.com/jumc/img_parallax_gif

References:

https://jaydenossiterghostart.wordpress.com/2016/05/12/the-history-of-the-parallax-effect/ (*The History of the Parallax Effect*)

http://www.nooganeer.com/his/projects/image-processing/making-a-gif-with-opencv-and-scikit-image-in-python (*Making a GIF with OpenCV and Scikit-Image in Python*)

http://sweet.ua.pt/pjf/PDF/Ferreira94e.pdf (*Interpolation and the discrete Papoulis-Gerchberg Algorithm* - Paulo J. S. G. Ferreira)

revistas.ua.pt/index.php/revdeti/article/download/1720/1597 (T*eaching signal and image reconstruction algorithms* - Paulo J. S. G. Ferreira)

http://cs.brown.edu/~pff/papers/seg-ijcv.pdf (*Efficient Graph-Based Image Segmentation* - P. Felzenszwalb, D. Huttenlocher)

http://scikit-image.org/docs/dev/auto_examples/segmentation/plot_rag_merge.html