FOURTH LABORATORY REPORT

A PREPRINT

Santiago E. Cortes[*]
Department of Mathematics
Los Andes University
Bogota, Cun
secg95@gmail.com

February 22, 2019

ABSTRACT

fourth laboratory

***K*eywords** First keyword · Second keyword · More

1 Introduction

This a report for a process to generate Hybrid images and image blending.

2 Headings: first level

For the laboratory there were chosen 4 images, only the one with the moon was modified with horizontal cropping. To combine two images of different sizes the mean of the heights and widths were used.

3 code snippets

Example of how the resized was done

```
img1 =  cv2.imread('fish.bmp')
img2 =  cv2.imread('plane.bmp')

x = int(np.mean([img1.shape[1],img2.shape[1]]))
y = int(np.mean([img1.shape[0],img2.shape[0]]))
img1 = cv2.resize(img1,(x,y))
img2 = cv2.resize(img2,(x,y))
```

This is how for the low pass filter the negatives were handled:

```
LowPassFilterTemp =  cv2.GaussianBlur((img1/opacador).astype('uint8'),(5,5), sigmaX=5,sigma
HighPassFilter = (img1/opacador).astype('uint8') - LowPassFilterTemp
```

This is how the two pyramids were computed:

[*] Use footnote for providing further information about author (webpage, alternative address)—*not* for acknowledging funding agencies.

```
def Gaussian_pyramid(img):
    ListaImagenes = [img]
    while True:
        if(img.shape[0] <= 4 or img.shape[1] <= 4):
            break
        lower_reso = cv2.pyrDown(img)
        ListaImagenes.append(lower_reso)
        img = lower_reso

    return ListaImagenes


def Laplace_pyramid(img):
    ListaImagenes = [img]
    while True:
        if(img.shape[0] <= 4 or img.shape[1] <= 4):
            break
        def substract(imag1,imag2):
            x = min(imag1.shape[1],imag1.shape[1])
            y = min(imag1.shape[0],imag1.shape[0])
            imag1 = cv2.resize(imag1,(x,y))
            imag2 = cv2.resize(imag2,(x,y))
            Laplace = imag1 - imag2
            negativos = imag1 < imag2
            Laplace[negativos] = 0
            return Laplace
        lower_reso = cv2.pyrDown(img)
        ListaImagenes.append(substract(img,cv2.pyrUp(lower_reso)))
        img = lower_reso
    return ListaImagenes
```

and finally how the blended image was done :

```
gaus = Gaussian_pyramid(Tarea2pubnto)
lap = Laplace_pyramid(Tarea2pubnto)

upsampl = gaus[5]

for i in range(1,5):
   Laplace = lap[5-i]
   x = int(np.mean([upsampl.shape[1],Laplace.shape[1]]))
   y = int(np.mean([upsampl.shape[0],Laplace.shape[0]]))
   upsampl = cv2.resize(upsampl,(x,y))
   Laplace = cv2.resize(Laplace,(x,y))
   temp = upsampl+ Laplace
   upsampl = temp
```

4 Results

4.1 Figures

References

[1] George Kour and Raid Saabne. Real-time segmentation of on-line handwritten arabic script. In *Frontiers in Handwriting Recognition (ICFHR), 2014 14th International Conference on*, pages 417–422. IEEE, 2014.

[2] George Kour and Raid Saabne. Fast classification of handwritten on-line arabic characters. In *Soft Computing and Pattern Recognition (SoCPaR), 2014 6th International Conference of*, pages 312–318. IEEE, 2014.

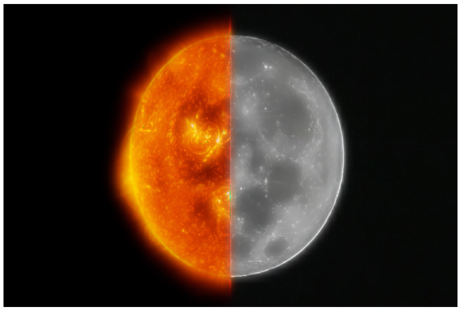

Figure 1: HybridImage.

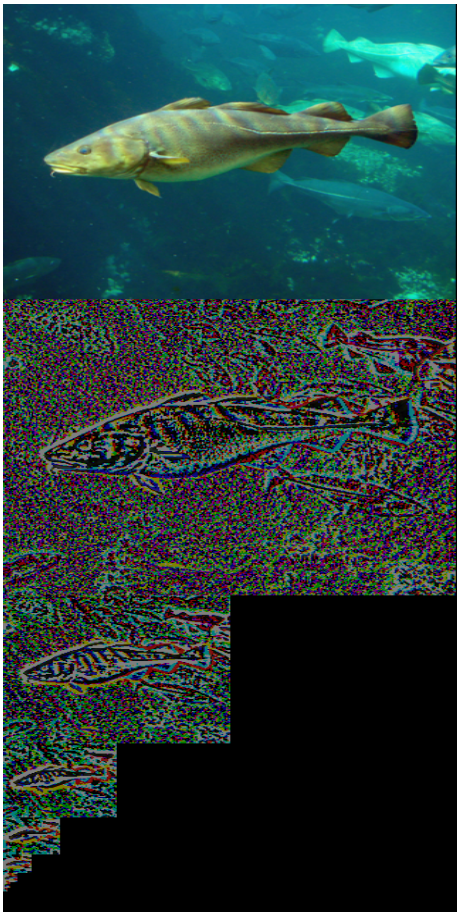

Figure 2: pyramid.

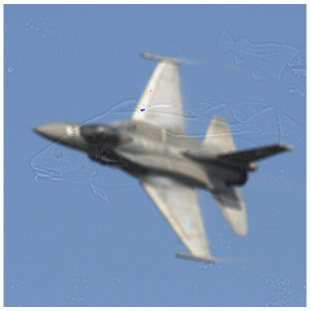

Figure 3: Hybrid.

[3] Guy Hadash, Einat Kermany, Boaz Carmeli, Ofer Lavi, George Kour, and Alon Jacovi. Estimate and replace: A novel approach to integrating deep neural networks with existing applications. *arXiv preprint arXiv:1804.09028*, 2018.