A red and black logo

Description automatically generated

Hacettepe University

Computer Engineering Department

BBM415 IMAGE PROCESSING LAB. - 2023 Fall

**Instructor**: Aydın Kaya

**TA**: Burçak Asal

**ASSIGNMENT 2**

Giving Cartoon Effect to Colorful Images

**Student name:**

Hasan Malkoç

**Student Number:**

2200356826

**Using Image Pyramid for Image Blending**

In the image editing tools such as Adobe Photoshop, GIMP, image blending is the one of most fundamental tasks and used for many purposes. For example, you can prepare a poster for an advertisement or a film. The most important thing for image blending methods is to blend images seamlessly. In other words, for a successful image blending method, seams where images or image regions are stitched must be invisible. There are many ways to blend two or more images. One such approach proposed in is Laplacian Pyramid. Accordingly, to this approach, images are first decomposed into their Laplacian pyramids, and then these images are blended in pyramid levels so that they are seamless.

**Approach**

The program will take an image as input and a masked image region from another or the same image, producing a blended image. The process involves the following steps:

* Build Laplacian pyramids for each image.
* Build a Gaussian pyramid for each region mask.
* Blend each level of the pyramid using the region mask from the same level:
* Collapse the pyramid to obtain the final blended image.

**My Implementation:**

First, I take two images with same sizes, first image is the source of the mask, and the second image is the image to apply mask on it. I also set a “mask for blending” image where it contains 0’s and 1’s, where 1’s represents the region of the mask as white pixels, and the rest is black pixels. Example:

A black square with a white square in the middle

Description automatically generated

Shouldn’t be confused with labels **Image1** and **Image2**, it is vice versa in the code.

Functions for this implementation:

def getMask(mask\_image):

def getRegionToMask(input\_image):

def setMaskImage(mask\_image, region\_to\_mask, mask):

For 4 main steps of the implementation my functions are:

def downsample(image):

def upsample(image, target\_shape):

def smooth\_image(image):

def gaussianPyr(image, number\_of\_levels):

def laplacianPyr(image, number\_of\_levels):

def collapsePyr(laplacian\_pyramid):

def blend(image1, image2, mask\_for\_blending, number\_of\_levels):

(Detailed implementation is in code/main.py)

**Outputs of these 4 main steps are as below:** (More examples are after this section)

Laplacian 1

A collage of different images

Description automatically generated

Laplacian 2

A collage of different images

Description automatically generated

Gaussian for Mask

A black square with a white square in the middle

Description automatically generated

Blended Pyramid

A collage of different images

Description automatically generated

Apple And Orange

A close-up of an orange and an apple

Description automatically generated

A black square with a white square in it

Description automatically generated

A close-up of a red apple

Description automatically generated

Just a quick clarification here, my implementation doesn’t apply mask on a circle area, I only choose rectangle area to apply mask so transition on the edges might be little

Laplacian 1

A blue and green sphere

Description automatically generated with medium confidence

Laplacian 2

A close up of a fruit

Description automatically generated

Gaussian of Mask

A black square with a white square in the middle

Description automatically generated

Blended Pyramid

A close up of a circle

Description automatically generated

**Mona Lisa**

A black rectangular object with a white line

Description automatically generated

Image 2 is the replacement of the mask since I used the same image here

Laplacian 1

A collage of different images of a person

Description automatically generated

Laplacian 2

A collage of different images of a person

Description automatically generated

Gaussian of Mask

A black rectangular object with white text

Description automatically generated

Blended Pyramid

A collage of different images of a person

Description automatically generated

Reconstructed Blended Image

A painting of a person with a face

Description automatically generated

**Cat**

**A black square with a white square in the middle

Description automatically generated**

Laplacian 1

A cat with a black and white striped pattern

Description automatically generated with medium confidence

Laplacian 2

A cat with a black and white striped pattern

Description automatically generated with medium confidence

Gaussian of Mask

A black square with a white square in the middle

Description automatically generated

A cat with a white whiskers

Description automatically generated

**Dog**

**A black square with a white square in the middle

Description automatically generated**

Laplacian 1

A screenshot of a video game

Description automatically generated

Laplacian 2

A close up of a dog

Description automatically generated

Gaussian of Mask

A black square with a white square in the middle

Description automatically generated

Blended Pyramid

A screenshot of a video game

Description automatically generated

A close-up of a dog

Description automatically generated

Let’s Increased the Number of Pyramid Levels until the final Pyramid level size is scaled down to minimum dimension.

Blended Pyramids:

A screenshot of a game

Description automatically generated

A comparison of images of a dinosaur

Description automatically generated

A close up of a dog

Description automatically generated

A collage of images of two people

Description automatically generated

A screenshot of a video game

Description automatically generated

\*I will add the results to google drive to keep this file low size.

Generally higher level of pyramid level blends more details so it gives smoother image than the lower level of pyramid level.