

# CSCE-689 Computational Photography

## Programming Assignment 2

Sriram Chakravarthy  
UIN:629009746

February 27, 2022

### Task 1: Hybrid Images

This task involves combining two images where one is low passed (contains only low frequencies) and the other is high passed (contains only high frequencies). The result creates the effect that the image with high frequencies is visible prominently when looked at from a close distance while the image with low frequencies is visible when viewed from a distance. In this part of the task, grayscale images are considered.

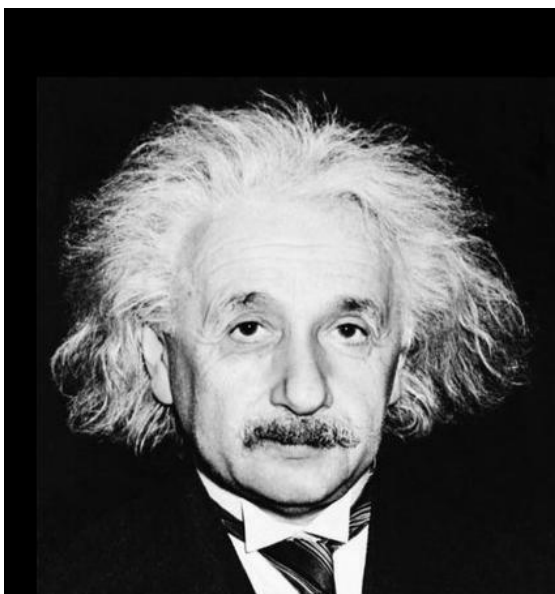
There are mainly two factors that affect the quality of the hybrid images, given a pair of input images.

1. Alignment of the two images: Before doing the frequency filtering, the images are aligned by choosing two points on each image. To create a good effect, it is important to choose good alignment points. For example, when two inputs are of human faces, the images are aligned keeping eyes as a reference.
2. Standard deviation of the filters: Gaussian filters are used to create the low frequency and high frequency images. The extent of filtering is decided by the standard deviation of the gaussian filter. This is varied depending on the input images.

In the implementation, gaussian filter was implemented and convolved with each image to create low pass and high pass effects. The challenge was to find the standard deviation that produced a good result. Below are few results from my experiments.



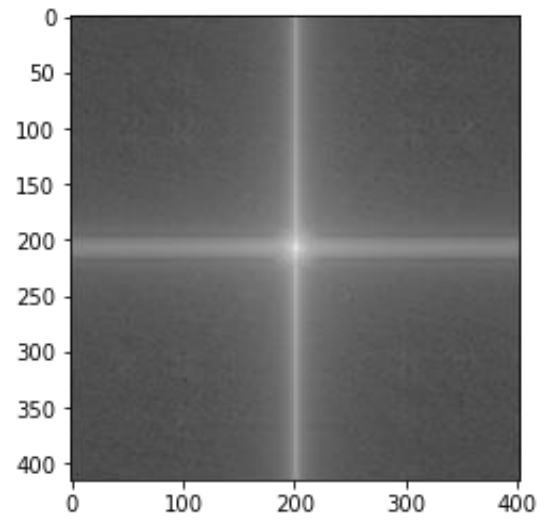
Monroe.jpg



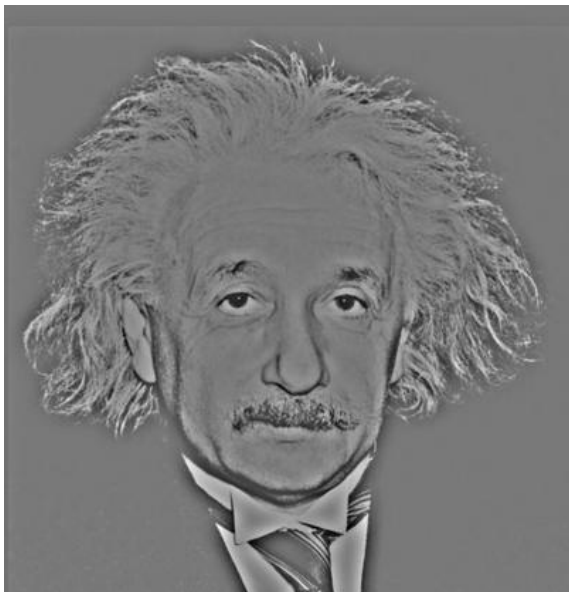
Einstein.jpg



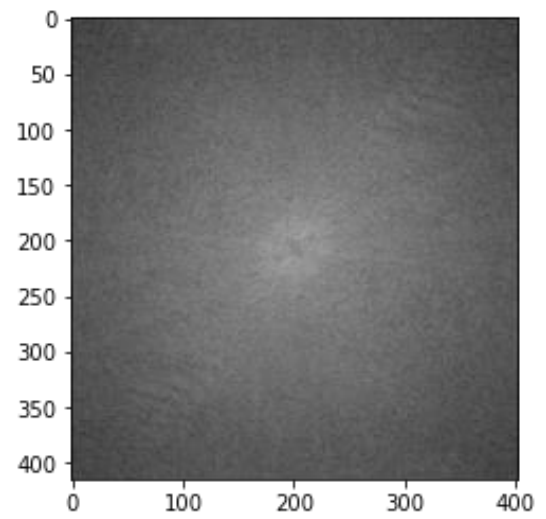
Low Passed Version of Monroe



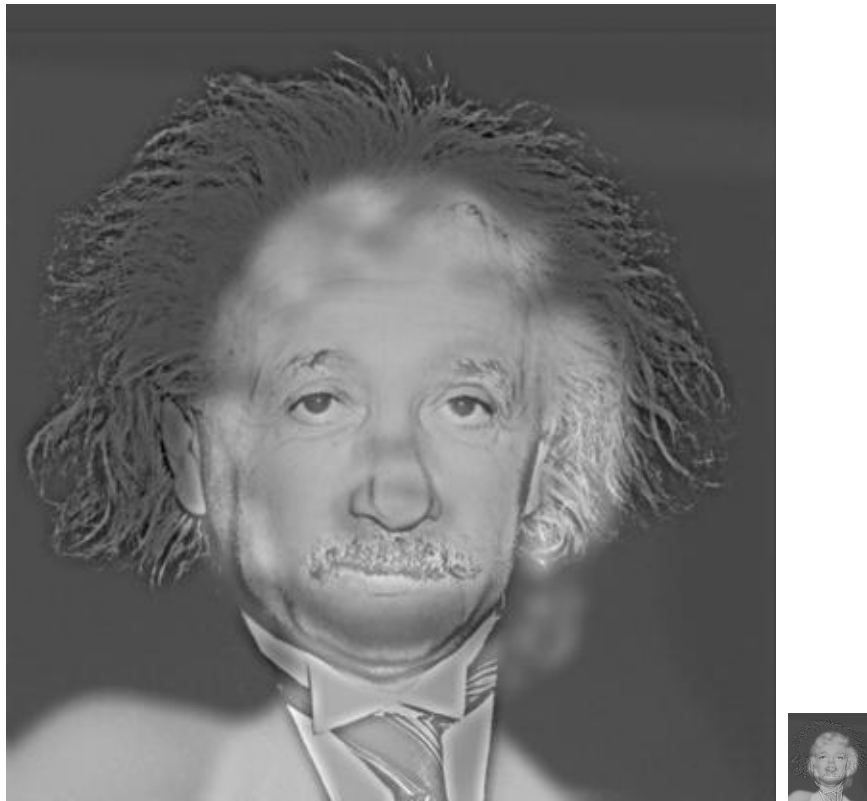
Freq Response



High passed version of Einstein

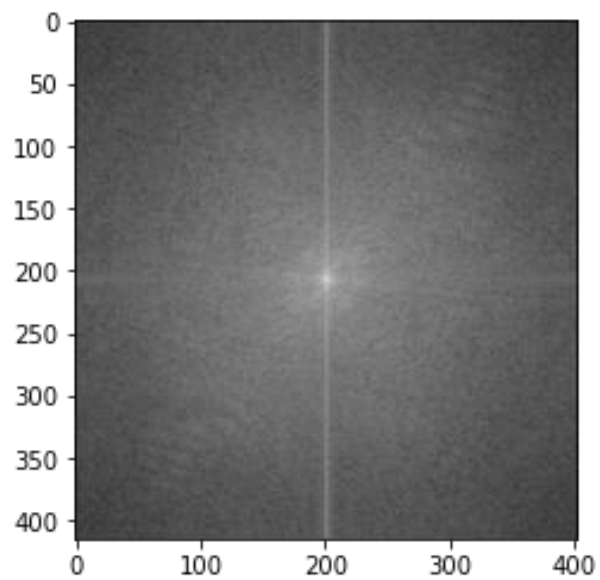


Freq Response



Hybrid Image zoomed in & zoomed out ( sigma = 7.0)

Resulting FFT:



2) Success Example : Brock Lesnar + Tiger



Brock Lesnar



Tiger



Hybrid Image zoomed in & zoomed out (sigma = 7.0)



3) Success Example : Lion + Tiger



Lion



Tiger



Hybrid Image zoomed in & zoomed out (sigma = 7.0)

#### 4) Failure Example: Zebra + Leopard

Based on my observations, choosing images which are too similar to each other doesn't create a good hybrid image. If the images have enough distinguishing perceptual features in different frequency regions, then the task is about finding the right alignment and cutoff frequencies to create a good visual effect. Otherwise, the result of combining the images will not be clearly seen in both zoomed in and zoomed out views.

For example, consider combining a zebra and a leopard. Apart from the shapes, the distinctive features ( stripes and spots) are both corresponding to high frequency. If the zebra image is low passed and leopard is high passed, leopard will be clearly visible in the zoomed in view. However, it won't be clear in the zoomed out view since its distinctive feature of stripes are no longer present.



Zebra



leopard



Hybrid Image zoomed in & zoomed out(unclear zoomed out version)

Varying the extent of smoothing the zebra image also doesn't help. If smoothing is reduced, quality is compromised in the zoomed in view where the zebra stripes interfere with the leopard spots.



Hybrid Image zoomed in & zoomed out (unclear zoomed in version)



## Using Color in hybrid images

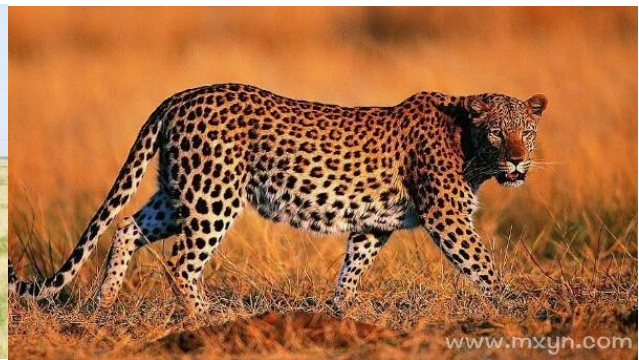
To enhance the hybrid image effect, I tried to add color in three ways :

1. Colored low pass image + Gray high pass image
2. Gray low pass image + Colored high pass image
3. Colored low pass image + Colored high pass image

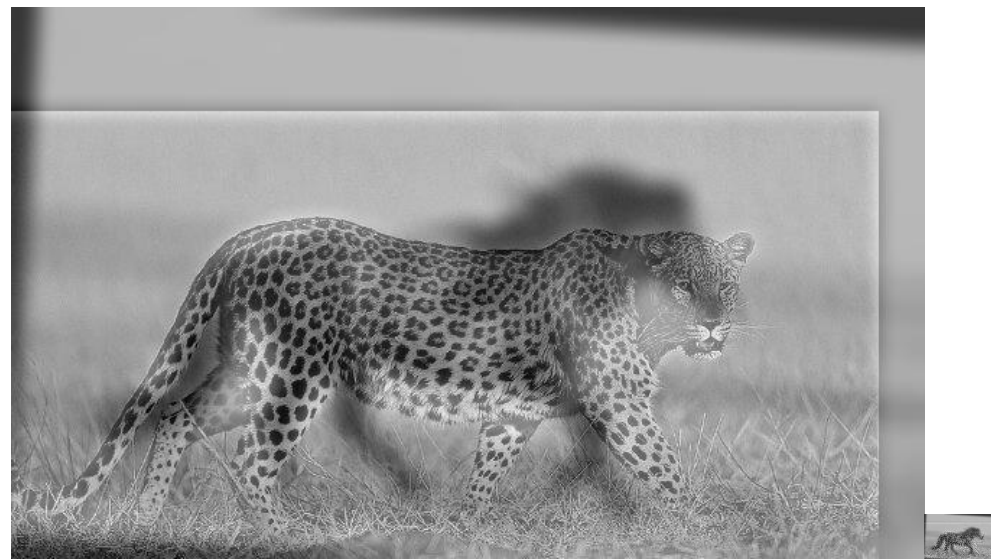
I chose the below images as inputs to see the effect of color as the images used in the previous section had atleast one image which was not colorful.



Horse



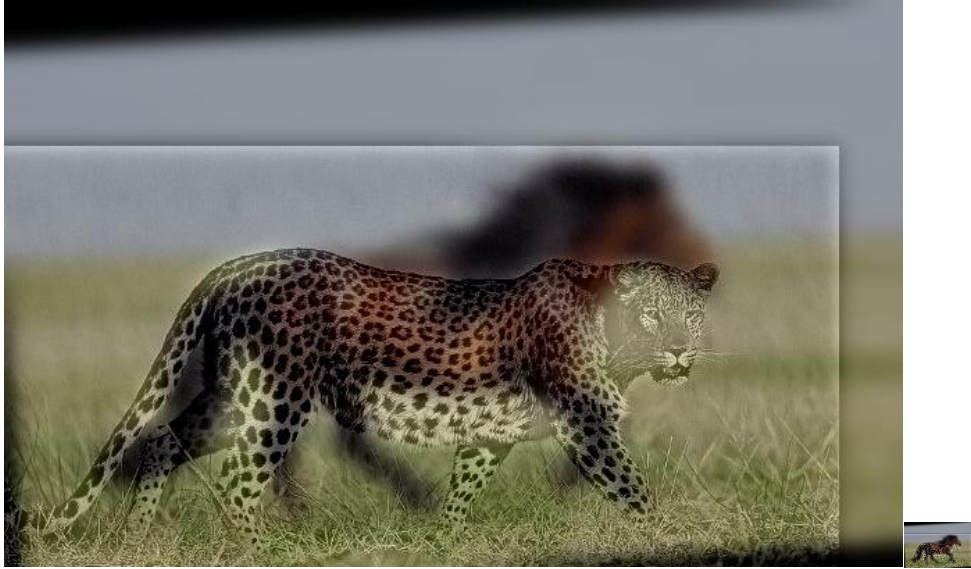
cheetah



Both gray

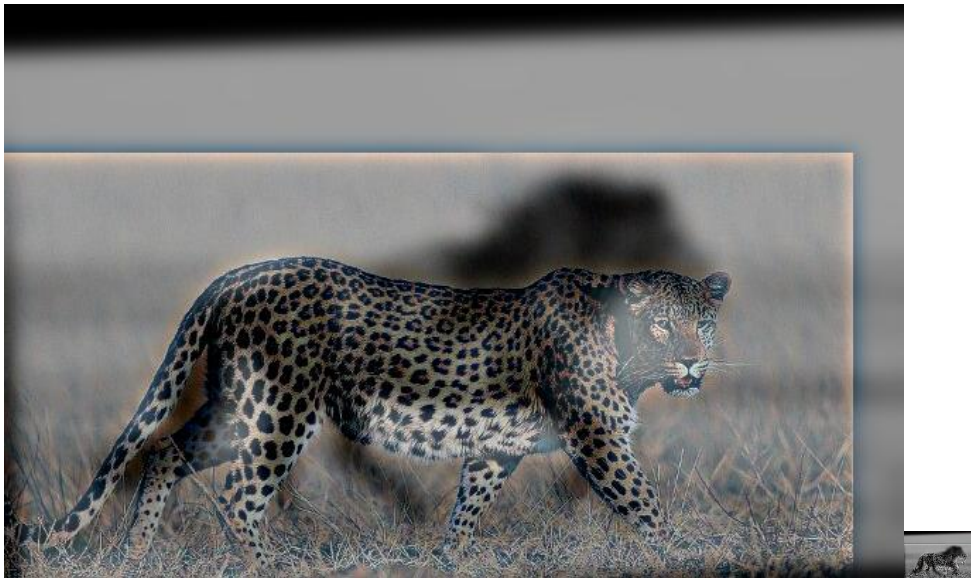
The gray version looks good. In following images, I tried to enhance it using colors.





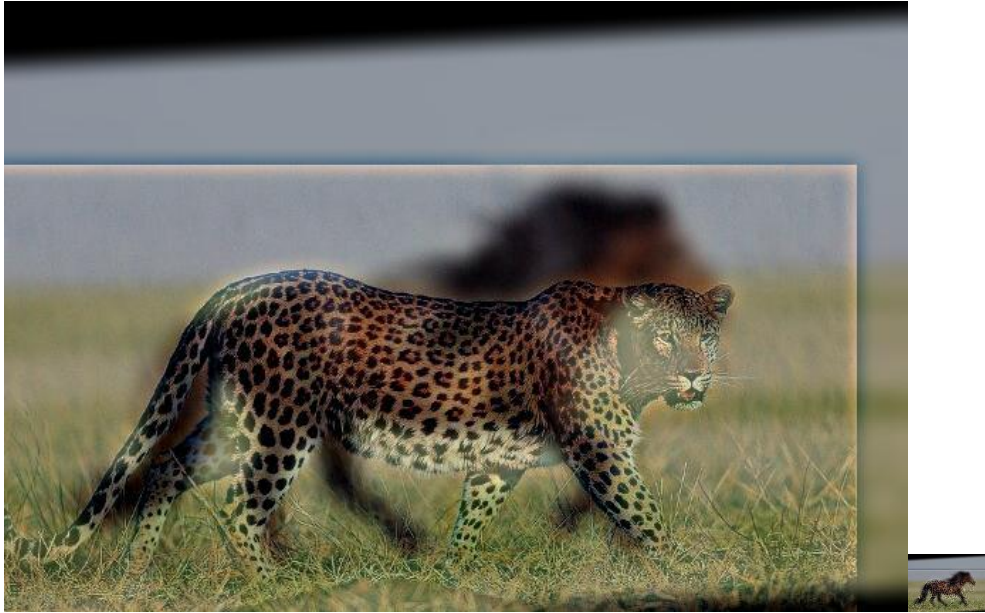
Colored horse + Gray cheetah

In this image, the low pass image of horse is colored, and high pass image of cheetah is gray. This looks particularly good in the zoomed-out image but in the zoomed in version, the color of the horse seems to draw attention away from the cheetah.



Gray horse + Colored cheetah

In this image, the low pass image of horse is gray and high pass image of cheetah is colored. This looks good overall. In the zoomed in image, the cheetah looks more prominent than the other images and in the zoomed out version, horse looks fine.



Both Colored

In this image, the low pass image of horse is colored and high pass image of cheetah is also colored. This is similar to the second result. In the zoomed in version, the color of the horse seems to draw attention away from the cheetah.

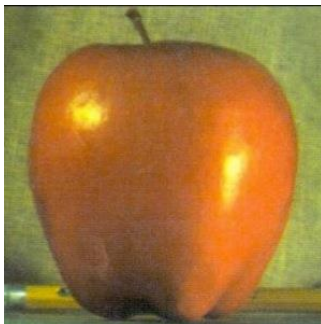
In my opinion, the third image with low pass image as gray and high pass image as colored creates the best visual effect.

## Task 2: Pyramid Blending

In this task, two color images(source and target) are combined according to a mask using Gaussian and Laplacian pyramids. This creates a much smoother blending than just combining the images using the mask.

Firstly, I tried to combine the apple and orange images using the naïve method i.e.,

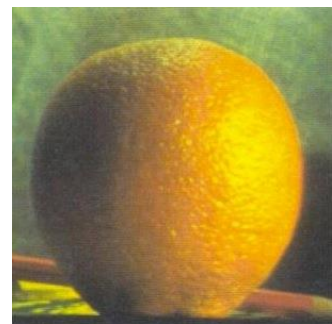
$$\text{Resulting image} = \text{Source} \times \text{Mask} + (1 - \text{Mask}) \times \text{target}$$



Apple(source)

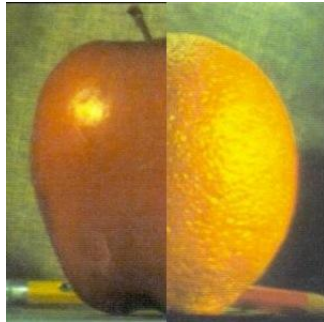


mask

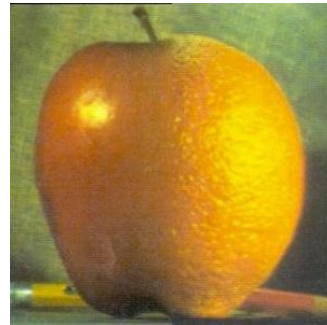


Orange(target)

In the naïve blending image, the transition from left image(apple) to the right image(orange) is not smooth and seems sudden.



Result of naïve blending



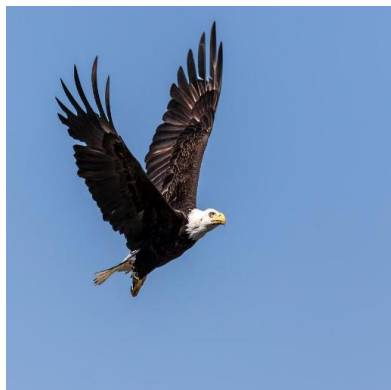
Result of pyramid blending

This is not the case with pyramid blending. There is a gradual fading of the color from left to right which produces a more natural looking blending effect as seen in the above figure.

The steps are as follows:

- Compute Laplacian pyramids  $LS$  and  $LT$  from the source and target images and a Gaussian pyramid  $GM$  from the mask.
  - Here 6 levels were used.
  - The input images' dimensions were set to next closes power of 2 by adding black pixels at the borders.
  - At each layer, the filtering and downsampling/upsampling were performed using *skimage.transform.resize*.
- Use the Gaussian pyramid to combine the source and target Laplacian pyramids as follows:
$$LC(l) = GM(l) \times LS(l) + (1 - GM(l)) \times LT(l)$$
where  $l$  is the layer index.
- Collapse the blended pyramid  $LC$  to reconstruct the final blended image and remove the borders added earlier.

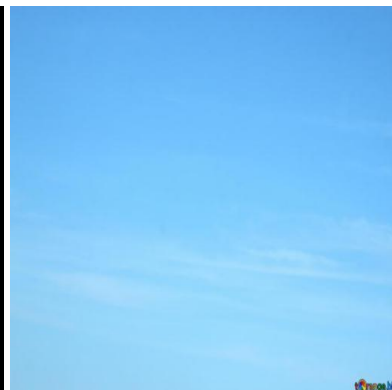
More examples are shown below.



Eagle(source)



mask



Sky(target)

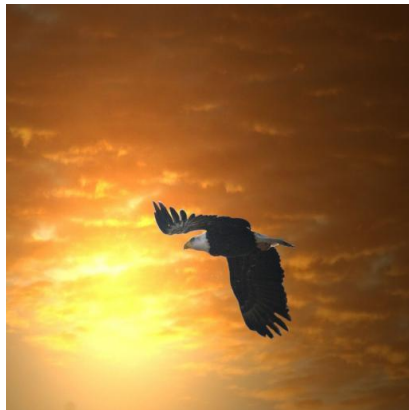


*Result of naïve blending*

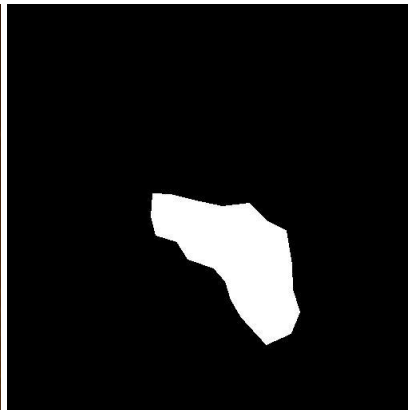


*Result of pyramid blending*

Failure example:



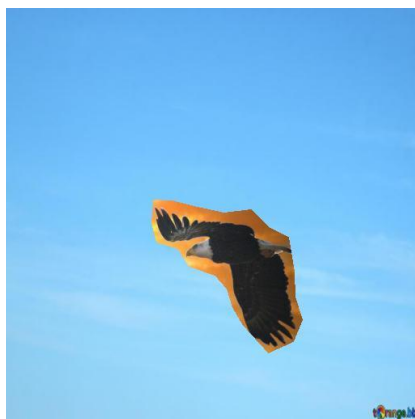
Eagle in sunset(source)



mask



Blue Sky(target)



*Result of naïve blending*



*Result of pyramid blending*

In this case, the backgrounds of the images are too contrasting. In other words, there is a high color gradient between the two image backgrounds. This causes the pyramid blending technique to be ineffective resulting in the eagle appearing to be on fire due to its sunset background in the source image. A suitable approach for blending such images would be gradient based blending.