Hybrid images

Faisal Qureshi
Professor
Faculty of Science
Ontario Tech University
Oshawa ON Canada
http://vclab.science.ontariotechu.ca (http://vclab.science.ontariotechu.ca)

Copyright information

License

Outline

- Image pyramids
- Hybrid images

Task

The goal is to laboratory is to construct *hybrid images* as suggested by Oliva et al. in their Siggraph 2006 paper [doi (https://doi.org/10.1145%2F1141911.1141919)]. A hybrid image is static images which changes as a function of viewing distance. Check out the following image taken from Wikipedia article on hybrid image. View this image from up close. You will read "southeast." Now move away from your monitor. As you get farther away from your display, it reads "northwest." How can that be? This image combines low-frequency component of one image (that reads "northwest") and high-frequency component of the other image (that reads "southeast"). Human visual system picks up high-frequency component up close and is unable to pick high-frequency components from a distance. Therefore, from a distance the text reads "northwest."

If you don't feel like moving away from your desk. You can always "simulate" this effect by zooming out the image.

Recipe

Given two images A and B

- Construct laplacian pyramid of image A
- Construct laplacian pyramid of image B
- Construct a new pyramid that takes some levels from A and others from B, recall that different levels of a laplacian pyramid captures signals at different frequencies
- Reconstruct the image from this new pyramid
- Voila

For this to work properly, you will need to fidget how many levels you copy from laplacian pyramid for image A and how many levels you copy from laplacian pyramid for image B. Furthermore, you should pick high-frequency component from images that exhibit higher spatial frequency (i.e., lots of edges and texture).

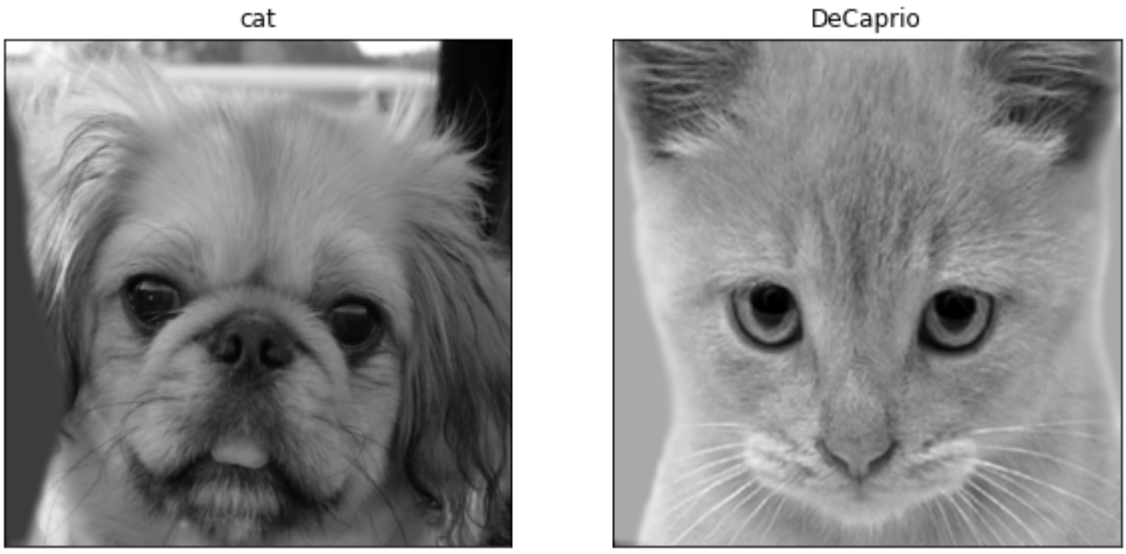
Starter code

Use the code below to get things going

In [290]:
```
import cv2 as cv
import numpy as np
import scipy as sp
from scipy import signal
import matplotlib.pyplot as plt
```

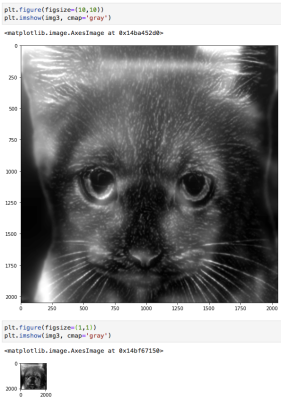
In [291]:

```
img1 = cv.imread('data/dog.bmp')
img1 = cv.cvtColor(img1, cv.COLOR_BGR2GRAY)
img1 = cv.resize(img1, (2048, 2048))

img2 = cv.imread('data/cat1.bmp')
img2 = cv.cvtColor(img2, cv.COLOR_BGR2GRAY)
img2 = cv.resize(img2, (2048, 2048))


plt.figure(figsize=(10,5))
plt.subplot(121)
plt.title('cat')
plt.imshow(img1, cmap='gray')
plt.xticks([])
plt.yticks([]);
plt.subplot(122)
plt.title('DeCaprio')
plt.imshow(img2, cmap='gray')
plt.xticks([])
plt.yticks([]);
```



What to expect

If all goes well, you will get the following effect. The same image when seen at two zoom levels appear to contain a different animal: cat when zoomed in and dog when zoomed out.



In [292]:
```
# Your solution goes here
```

Test on other images

Now construct hybrid images from (h11.jpg, h12.jpg), (h21.jpg h22.jpg), (h31.jpg h33.jpg), and (einstein.bmp, marilyn.bmp) images found in the data folder.

Submission

Include code and all hybrid images in a single jupyter notebook.

Submit via canvas.