Programming Assignment 1: Hybrid Images

Due Date: 11th September 2014

• Introduction

The goal of this assignment is to create hybrid images using a modification of the approach described in the SIGGRAPH 2006 paper by Oliva, Torralba, & Schyns. Hybrid images are static images that change in interpretation as a function of the viewing distance. The basic idea is that high frequency tends to dominate perception when it is available, but, at a distance, only the low frequency (smooth) part of the signal can be seen. By blending the high frequency portion of one image with the low-frequency portion of another, you get a hybrid image that leads to different interpretations at different distances.

• Assignment

Part I:

- **Correlation** and **Convolution**: Your first task is to generate your own correlation and convolution functions in Matlab/Octave. Each one of these functions take as input an image as well as a mask, and outputs the results of the corresponding processing as follows:

```
ImageConv = MyConv(ImageIn, Mask);
ImageCorr = MyCorr(ImageIn, Mask);
```

- **Mask**: Implement a function in Matlab/Octave that computes a Gaussian mask, with parameters **σ** and the size of the mask as follows:

```
GuassMask = MyGauss(Sigma, Size);
```

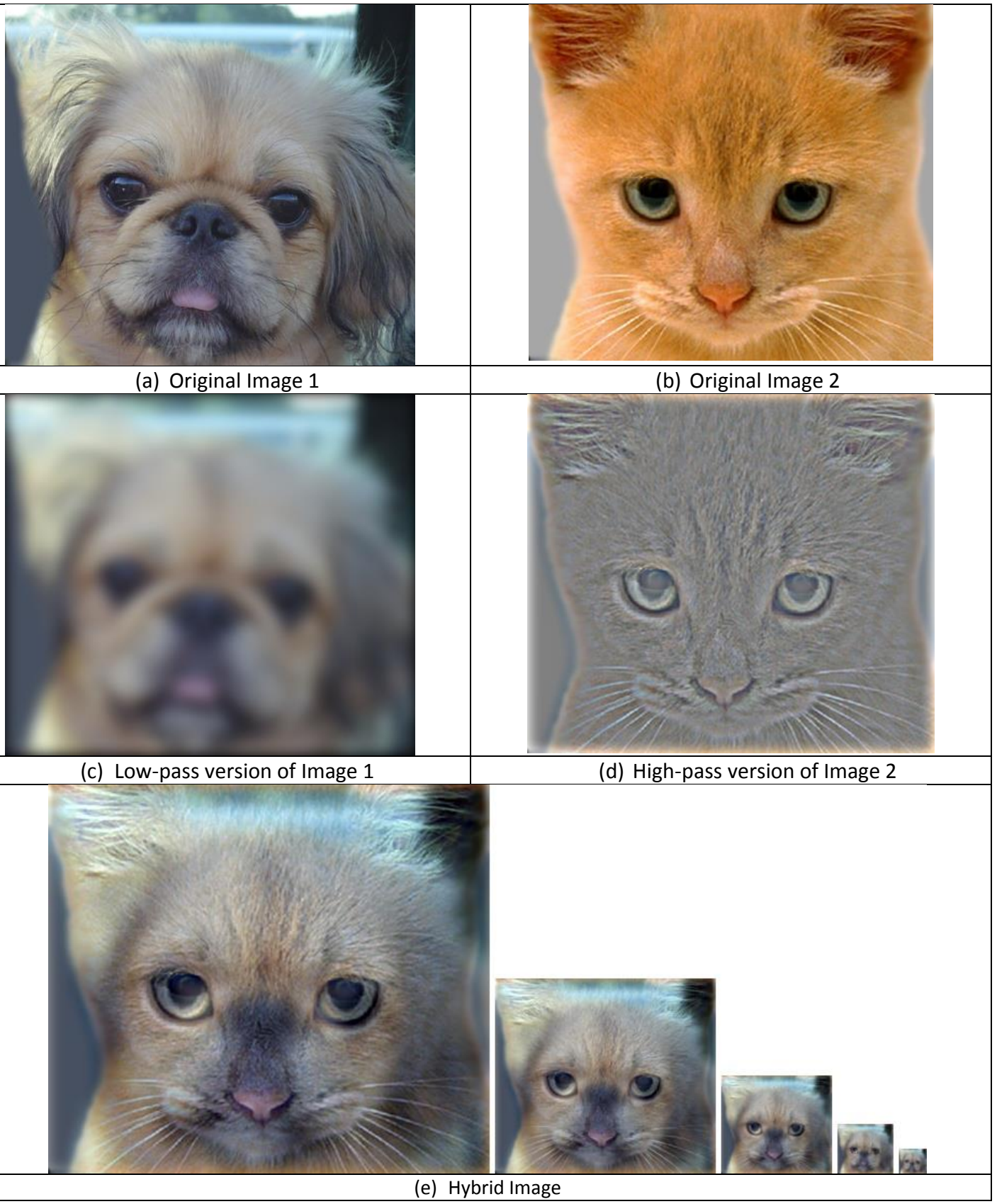
- **Gaussian Pyramid**: Take your favorite image and construct a Gaussian Pyramid. Display the results. What is the minimum size necessary to store it?

```
MyGaussPyramid(ImageIn);
```

Part II:

- **Hybrid Image:** Is the sum of a low-pass filtered version of the one image and a high-pass filtered version of a second image. There is a free parameter, which can be tuned for each image pair, which controls how much high frequency to remove from the first image and how much low frequency to leave in the second image. This is called the "cutoff-frequency". In the paper it is suggested to use two cutoff frequencies (one tuned for each image) and you are free to try that, as well.
    - You are provided with a few sample pairs of images (*that are aligned!!*) that you can make into hybrid images for testing purpose. You should provide your own images as part of results.

Figure 1: Example of Hybrid Image Creation (Source: N. Snavely)



(a) Original Image 1

(b) Original Image 2

(c) Low-pass version of Image 1

(d) High-pass version of Image 2

(e) Hybrid Image

- The paper suggests using the standard 2D Gaussian filter for low-pass filter and using impulse filter minus the Gaussian filter (which can be computed by subtracting the Gaussian-filtered image from the original) for high-pass filter. The cutoff-frequency of each filter should be chosen with some experimentation.
- You are **NOT ALLOWED** to use the following in-built functions from Matlab/Octave:
  - `fspecial(),imfilter(),filter2(),conv2(),nlfilter(),colfilt(), corr2(),etc.`
- If you're having trouble seeing the multiple interpretations of the image, a useful way to visualize the effect is by progressively down-sampling the hybrid image as is done above.
- Implement your Hybrid function such that it takes two images and corresponding cut-off values as input and outputs the hybrid image as follows:

```
HybridImage = MyHybrid(Image1, Image2, CutOff1, CutOff2);
```

- **Submit**

1. Complete source code, sample images, and README file as a tar-gzipped archive.
   a. When unzipped, it should create a directory with your ID. Example: **P2008CS1001** (NO OTHER FORMAT IS ACCEPTABLE!!! Case sensitive!!!)
   b. ***Negative marks if the TA has to manually change this to run his scripts!!***
2. Source Code should include the following files:
   a. `MyConv.m, MyCorr.m, MyGauss.m, MyGaussPyramid.m, MyHybrid.m`
   b. `TestScript.m`: This file will be used by the TA to test your project. You should include commands in this file that showcase your Hybrid Image function (along with sample images and corresponding cut-off values) and Gauss Pyramid (run on your chosen image)
   c. Case-Sensitive file/function names!!
   d. ***Negative marks for any problems/errors in running them***
3. In addition to the sample images provided, you need to submit two pairs of images of your choice for testing hybrid image formation. Make sure the images are aligned!!
4. Include a README file to convey any details about the project to the TA!
5. Submit/Upload it to moodle.

- **Grading**

1. Correct implementation of `MyConv, MyCorr, MyGauss`: 5 points each (15 Points)
2. Correct implementation of `MyGaussPyramid`: 10 points
3. Correct implementation of `MyHybrid`: 20 points
4. Showcasing all the functionality using `TestScript`: 10 points
5. Correct output for your test files: 10 points
6. Correct output for instructor's test files: 10 points
7. Total = 75 (Late Submission: 10% Penalty per extra day)
8. **Can discuss ideas, but STRICTLY no copying/sharing or source code.**