Melbourne School of Computing and Information Systems

COMP90086 Machine Learning - Assignment 1:

Image formation & Filtering

Tuan Khoi Nguyen - 1025294

Word Count: 342 (Q.1), 475 (Q.2)

Introduction

The purpose of the report is to be able to work out the real world position estimation from an image, as well as being able to understand how filtering can have effect on pictures, show specifically through the following tasks:

- **Question 1** - Find the real world coordinates of point A and B in Image 1.
- **Question 2** - Use Fourier Transform and filters to create an embedded text to an image background.

Figure 1: The original image for Question 1

Question 1: Mapping between world and image coordinates `[Word Count:342]`

Image coordinate extraction

To be able to find which pixel the points lie on, or where the coordinate origin - the pinhole position is set, appropriate color extraction is done. With the red marks being manually added, it is feasible to find group of pixels that share the same red color.

To make it easy to extract the color group, the image is converted to HSV color, so that the same color tones are grouped into the same amount of Hue. For the bright red color of the marks, a strict range of Hue - 0 to 5 is applied.

By applying the color filter, the masks are retrieved as shown in Figure 2, with the white area representing the pixels with the indicated color.

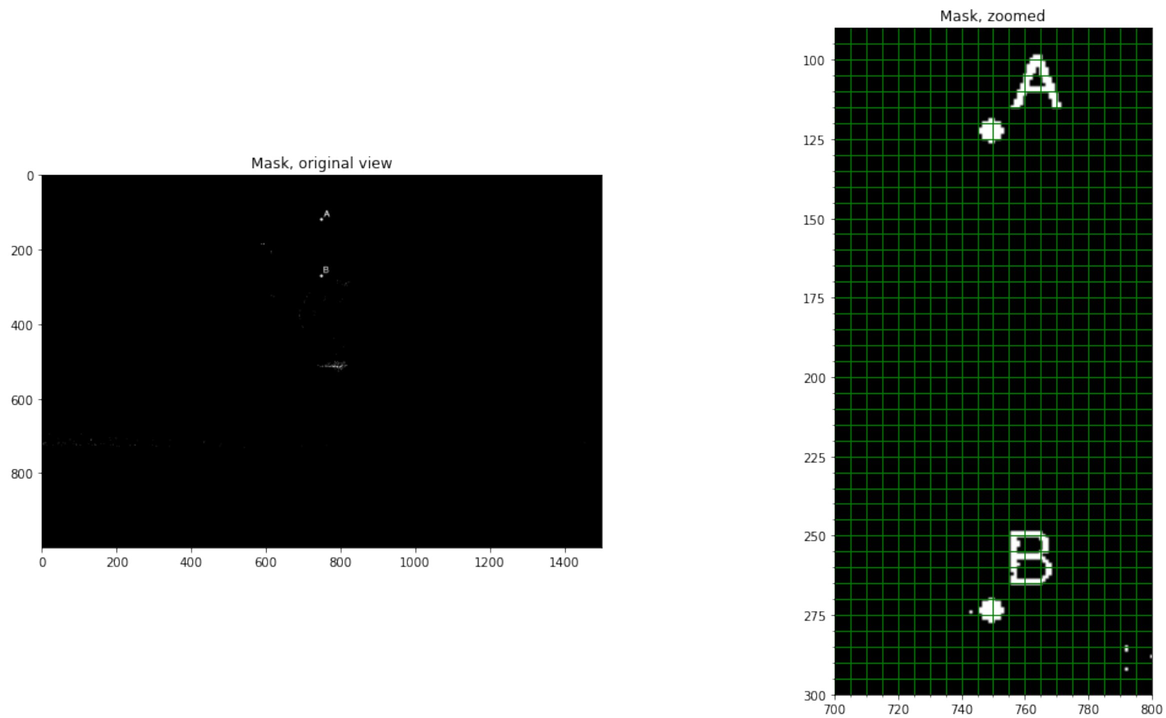

Figure 2: Color mask evaluated on the original image, and their close-up views

As the camera aperture is assumed to be at $(0, 0, 0)$ and the table surface is the x-axis, it is feasible to conclude that the y-axis lies in the middle of the image width and x-axis lies in the middle of the image height, that is, $x_0 = width(image)/2 = 750$ and $y_0 = height(image)/2 = 500$.

A and B: From observation, the points lie somewhere in $x_{image} = [700, 800]$ and $y_{image} = [100, 300]$. Zooming to the approximated range, it is clear that the marks align perfectly at y-axis ($x_{image} = 750$). Therefore, at $x_{image} = 750$, continuous white pixels at size larger than 3 are considered as the dot point. As expected, 2 pixel groups that represent the points are found, with lower y_{image} corresponds to the point with higher position in the image, which is A:

- A: $y_{image} = [119, 125]$, $y_{centroid(image)} = 122 \longrightarrow A_{image} = (750, 122)$
- B: $y_{image} = [271, 276]$, $y_{centroid(image)} = 273.5 \longrightarrow B_{image} = (750, 273.5)$

Mapping to real world coordinates

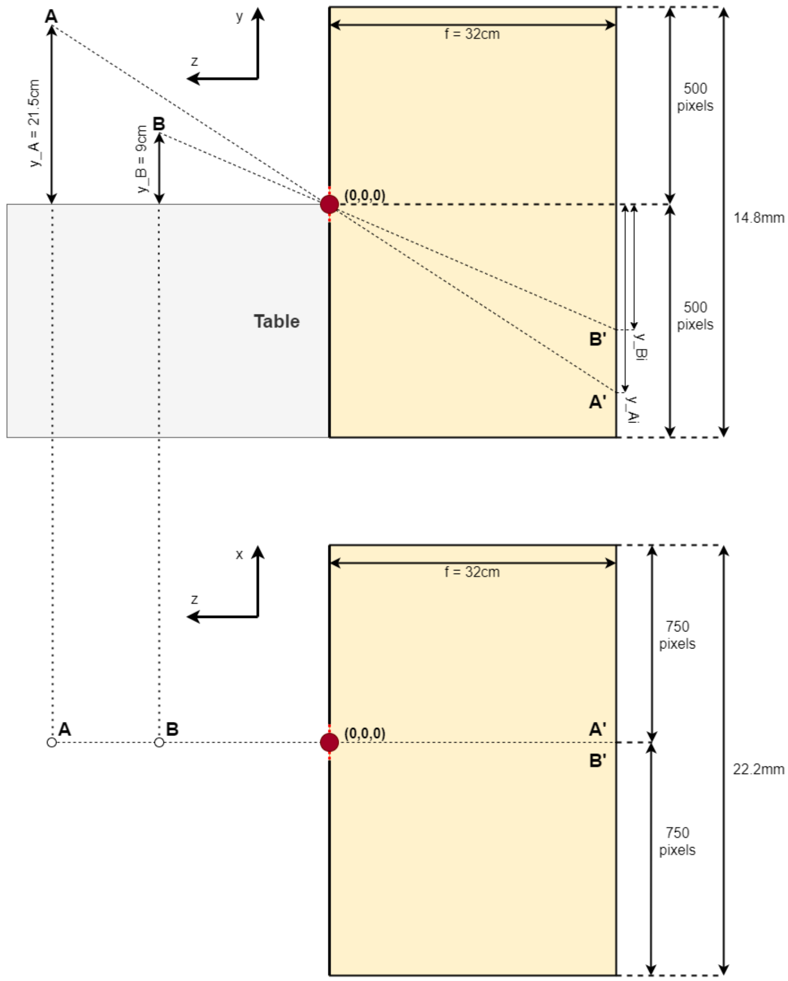

Figure 3: The working diagram of the pinhole model

Given that the image sensor is 1500×1000 pixels and has dimension of $22.2mm \times 14.8mm$, dimensions in the image can be converted from pixels to millimeters, using $1mm = \frac{1500}{22.2} = \frac{1000}{14.8} \approx 67.57\, pixels$

With the x and y image coordinates of A and B found, their real world coordinate in pixels and millimeters, can be calculated as follows:

- $x_{Ai} = |750 - 750| = 0\, pixels = 0mm$
- $y_{Ai} = |500 - 122| = 378\, pixels \longrightarrow y_{Ai} = 378/67.57 = 5.59mm$
- $x_{Bi} = |750 - 750| = 0\, pixels = 0mm$
- $y_{Bi} = |500 - 273.5| = 226.5\, pixels \longrightarrow y_{Bi} = 226.5/67.57 = 3.35mm$

With the table surface aligning to the x-axis and predefined height values are given, real world y-coordinates of points A and B can be retrieved:

- $y_A = 2150mm$
- $y_B = 900mm$

Given the pinhole formula $x = x_i \frac{z}{f}$, then $x_A = x_B = 0cm$. Using the pinhole formula $z = f \frac{y}{y_i}$:

- $z_A = 32 \times \frac{2150}{5.59} = 12307.69mm \approx 123.08cm$
- $z_B = 32 \times \frac{900}{3.35} = 8597.01mm \approx 85.97cm$

Overall, the resultant real world coordinate found for the required points are:

- $A = (0.0, 21.5, 123.08)\ cm$
- $B = (0.0, 9.0, 85.97)\ cm$

Question 2: Secret messages with bandpass filters `[Word Count:475]`

To create the secret message, a background image and a text image, both having the size of 500×500, will be used for creation process. They will be converted to grayscale and undergo Fourier Transform, and then get multiplied with a filter in Fourier Domain, specifically:

- Gaussian low-pass filter for background image.
- Gaussian high-pass filter for text image.

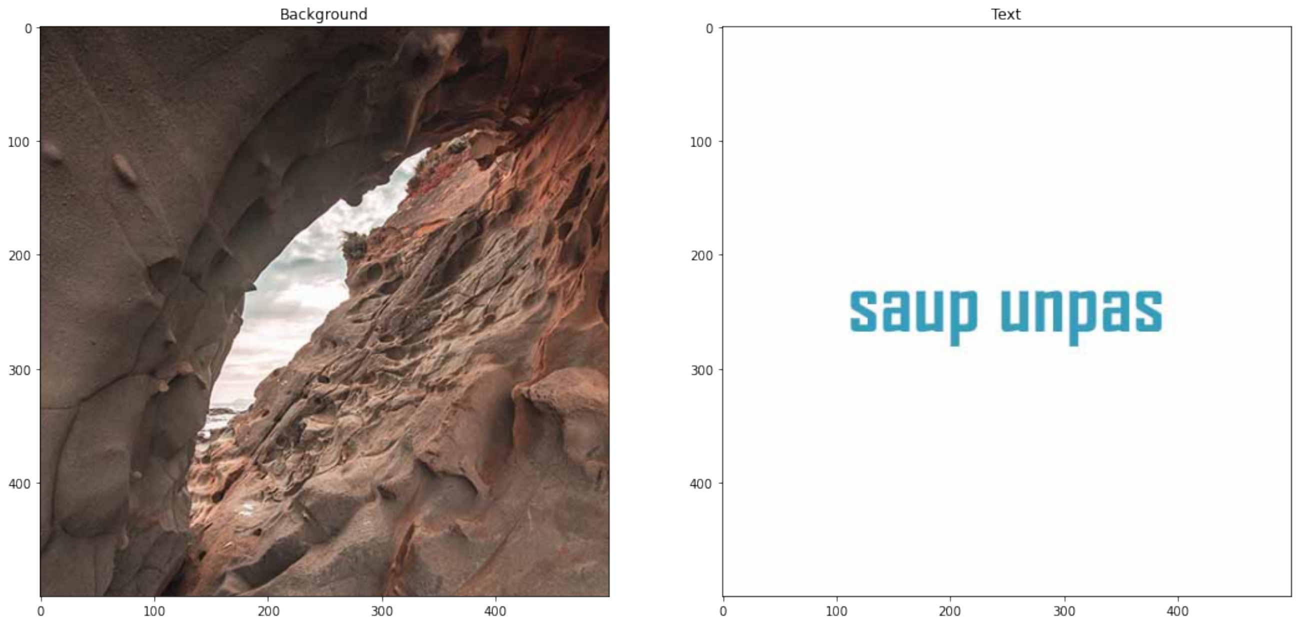

Figure 4: Original 500×500 images created by the author to be used for processing

A Gaussian filter has the following formulation:

$$g_\sigma(x,y) = \begin{cases} exp(-\frac{x^2+y^2}{2\sigma^2}) & \text{if Low-pass} \\ 1 - exp(-\frac{x^2+y^2}{2\sigma^2}) & \text{if High-pass} \end{cases}$$

(x,y: Point coordinate of pixel from image center)

σ is the standard deviation that represents the radius of the filter. For this question, a range of different σ values will be applied, and from there evaluate how this variable can change the behavior of outcome.

Evaluation on effect of σ

Plotted results from Figure 5 show the following for σ_{low} - the radius of low-pass filter used on the background and σ_{high} - the radius of high-pass filter used on the text:

- $\sigma_{low} = 0$: All pixels of background are flattened to the same color.
- As σ_{low} increases, background will become less blurry, and will eventually come to its original sharpness, text will also become harder to recognize.
- $\sigma_{high} = 0$: All pixels of text image retained their color and embedded clearly into background.
- As σ_{high} increases, the text will lose details, and will eventually flatten all pixels.

Explanation for these behaviours will be listed on next page.

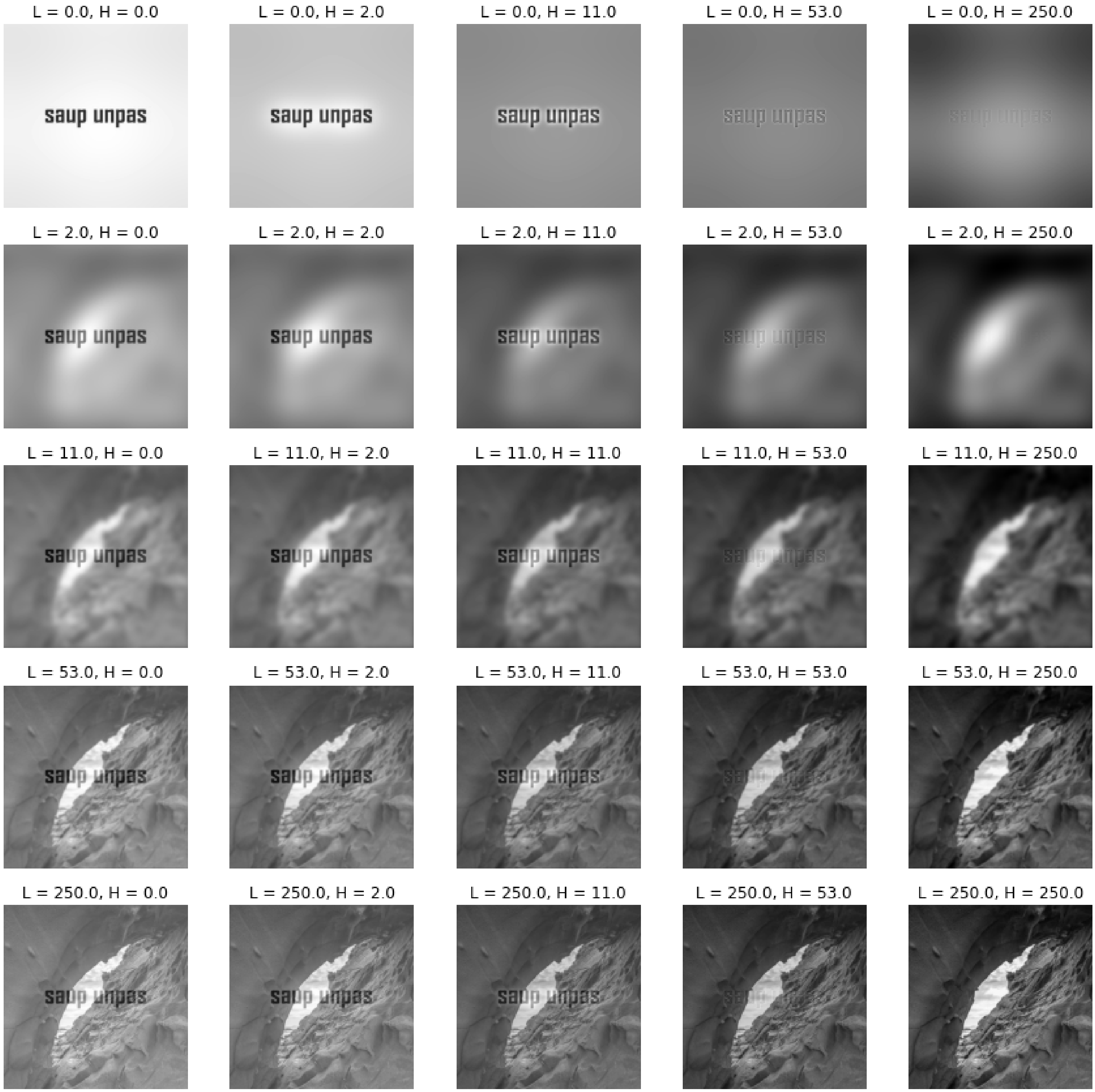

Figure 5: Histogram for `'class'=bridge` coordinates.

The behaviour of low-pass filter is that it tends to filter out high frequencies, which corresponds to small details of an image. Therefore, the larger the radius, the higher the threshold, and smaller background details will be passed on, which might cover out the smaller frequency details of the text.

Contrary to the above, high-pass filter keeps higher frequencies. This means it only keeps small details over its threshold, and will eventually filter out the whole text when details are not small enough to lie above threshold frequency.

The more detail, the better?

With lower σ_{low} corresponds to more detailed background, suggestion to maximize the detail by setting higher radius is feasible. Testing this hypothesis shows the following results:

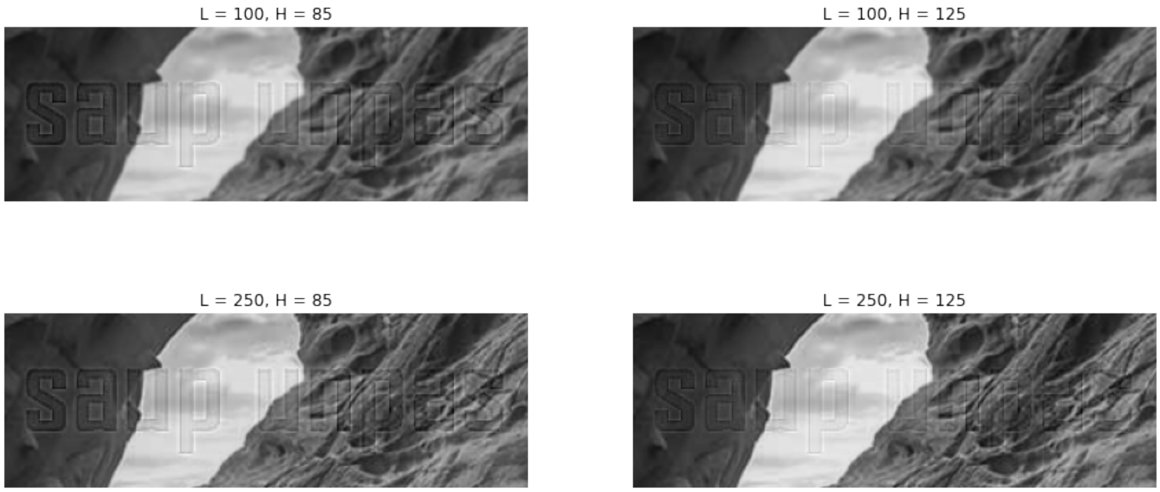

Figure 6: Close-up comparison of different sharpness

The results on $\sigma_{low} = 250$ shows that there is uneven clearness in the text. This is because as higher frequencies get in the image, discontinuity in frequency across the image will appear. The areas with lower frequency will let the text noticeable, but as higher frequencies appear, they will blend out the text's lower frequency pixels within their area. Therefore, an in-between value of $100 < \sigma_{low} < 250$ is considered suitable to display the text, as it crosses out the small details, but still keep most visible features while making the text able to appear more simultaneous.

Result

As seen in Figure 6, at $\sigma_{low} = 250$, the image may still be visible at $\sigma_{high} = 85$, and become hard to notice clearly at $\sigma_{high} = 125$, hence the former value is chosen. Setting an in-between value of $\sigma_{low} = 125$ returns the following result on Figure 7.

The image now satisfies the requirements: visible on close-up while remain unnoticeable when looked from far away. Therefore, $\sigma_{low\ pass} = 125$ for background image, and $\sigma_{high\ pass} = 85$ for text image is suitable for the author's choice of images, and Figure 7 will be used as the resultant image for this question.

Figure 7: The final result image for Question 2