Assessment (ECE2111)

Lab 02 Results Document

Name: Tan Jin Chun

Student ID: 32194471

ECE2111 lab2 results document:

Name: Tan Jin Chun
Names of all students you discussed this lab with: Loh Jia Quan, Kelvin Ku Yew Siang, Chong Yen Juin, Huan Meng Hui

Section 1:
Output from script in item 2:

```
% Question 2(c)
y =

     1     2     3     3     2     1


ny =

    -2    -1     0     1     2     3

% Question 2(d)
y0 =

     1     2     3     3     2     1


ny0 =

    -2    -1     0     1     2     3

% Question 2(e)
y1 =

     1     3     6     8     8     6     3     1


ny1 =

     0     1     2     3     4     5     6     7


y2 =

     1     3     6     8     8     6     3     1


ny2 =

     0     1     2     3     4     5     6     7
```

Question 2(c): What does `[y, ny] = myconv(x,nx,h,nh)` *return? Why do you expect this to happen?*

```
% Question 2(c)
y =

     1     2     3     3     2     1


ny =

    -2    -1     0     1     2     3
```

The output would be the impulse response of the system.

For vector y, it is the result of the convolution between x and h (the inputted vector and the impulse response vector). Vector y is essentially the scalar product of vector x and vector h and will be of the same length with vector ny.

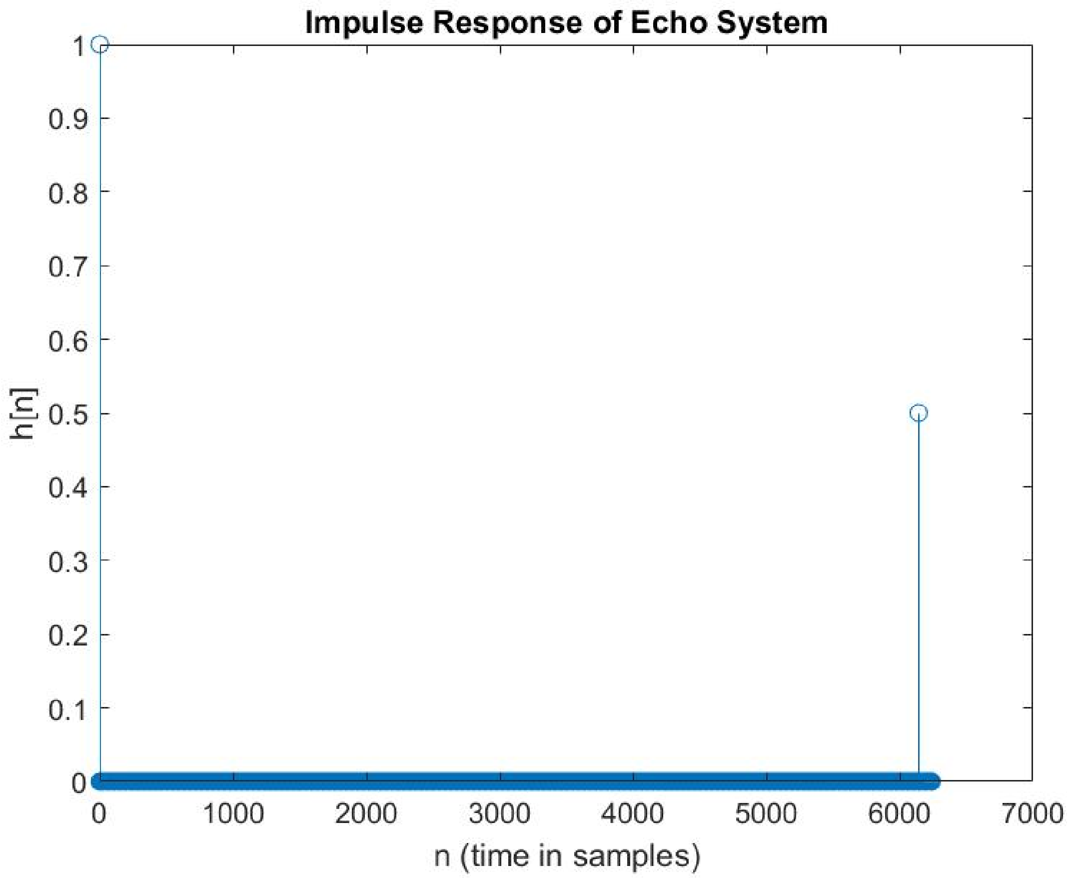
For vector ny, the boundary will be -2 and 3 (lower boundary and upper boundary).

Question 2(d): What relationship do you expect between `myconv(x,nx,h,nh)` *and* `myconv(h,nh,x,nx)`*?*

Based on the output produced in the command window, the output results will be the same. This is because the commutative principle holds true for the convolution operation that has been carried out. This means that $x(n)*h(n) = h(n)*x(n)$

Section 2:

Include your figure for item 2(b) below.



Question 2(e): What happens if you increase or decrease the parameters D and alpha?

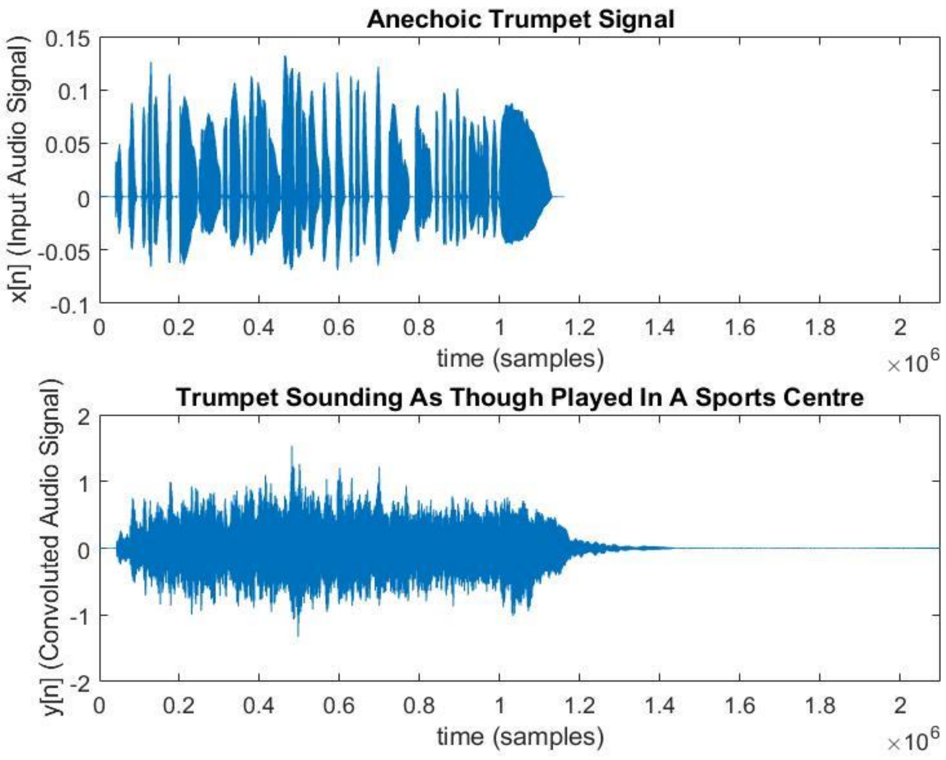
When D is increased, the time delay between the echoes will increase.
When D is decreased, the time delay between the echoes will decrease.
When alpha is increased, the amplitude of the echoes will increase.
When alpha is decreased, the amplitude of the echoes will decrease

Question 2(f): How many echoes do you hear?

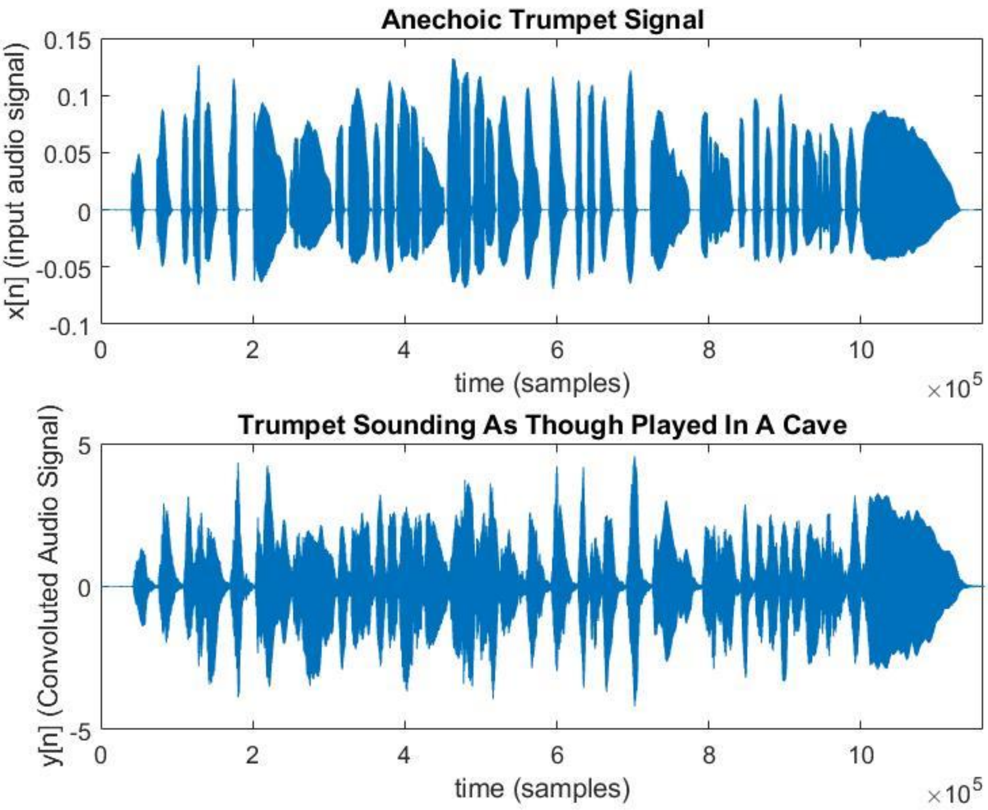
2 echoes

Section 3:

Include your figure from item 2(c) below:

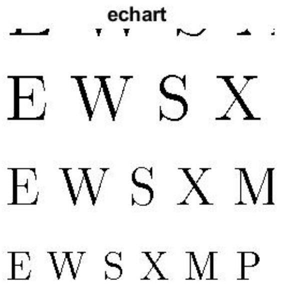


Include your figure from item 3(c) below:

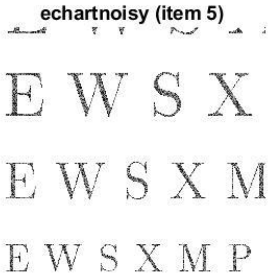


Section 4

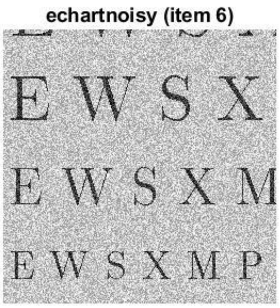
Include your image from item 2 below:



Include your image from item 5 below:



Include your image from item 6 below:

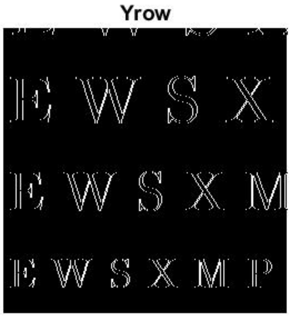


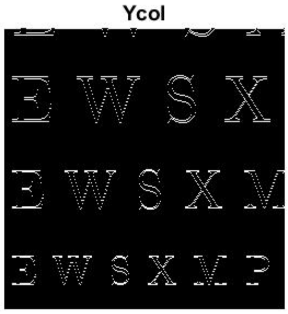
Question 3: How many pixels wide and how many pixels high is the image?
The image is 256 pixels wide and 257 pixels high

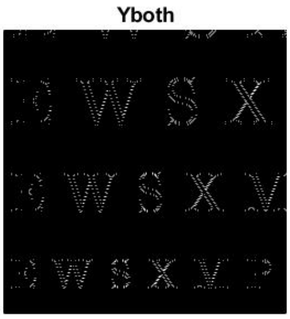
Question 7: What is the difference between the displayed images in items 5 and 6?
Items 5 has a white background while item 6 has a grey background. This is because the image in item 5 displays the image uses the default range, where the value smaller than 0 are black while the value larger than 1 is white. The image in item 6 however is rescaled in such a way that the smallest matrix entry displays black and the largest matrix entry displays white. Values in the middle would be scaled and this will show shades of greys.

Section 5:

Include the images from item 1(d) below



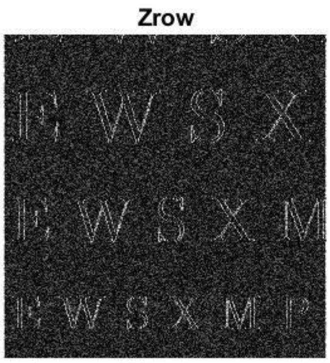




Question 1(e): Based on the images you made in part (d), briefly explain how convolution of the rows and/or columns of the image with h, followed by an absolute value operation, enhances the edges of an image.

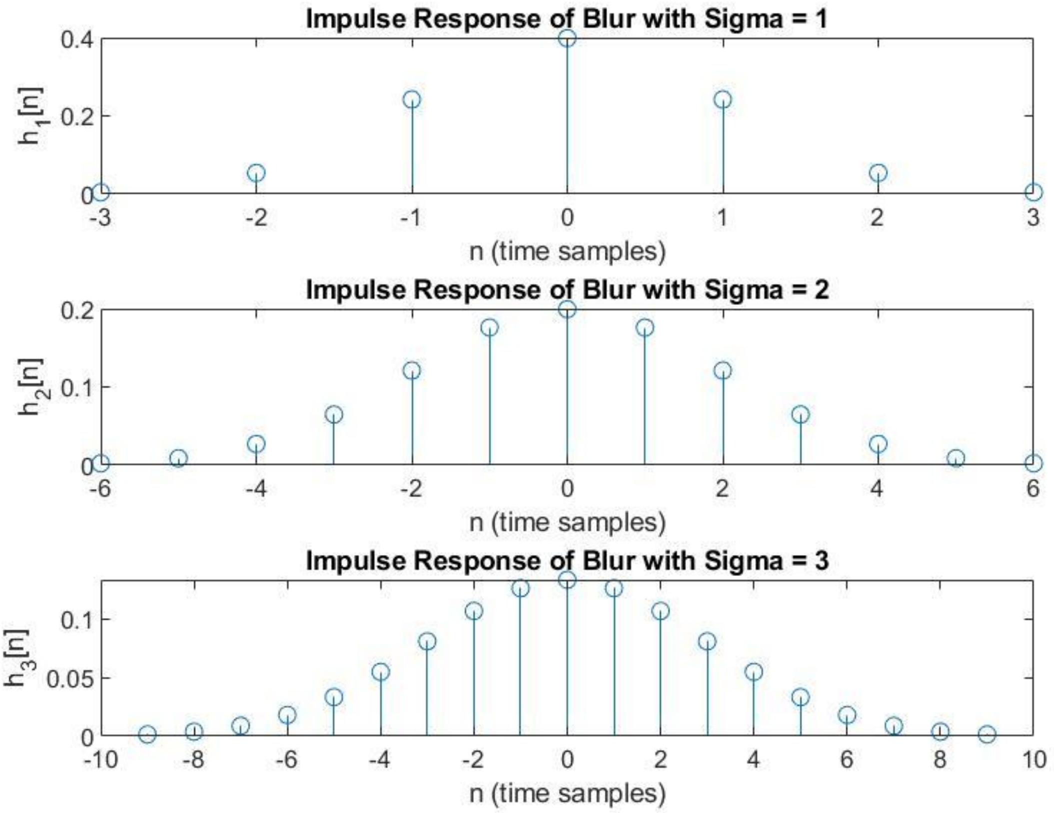
Convolution of the rows will refine the vertical edges in the image and the convolution of the column will refine the horizontal edges in the image. If an absolute value operation is followed, the values closer to zero will be darker and the value straying away from 1 will be whiter. The display range would be in between 0 and 1 and this would result in the enhancement of the edges of an image.

Include the image from item 2(b) below



Question 2(c): How does noise affect the result of the edge enhancement operation?
The noise will negatively affect the convolution process which would result in a disruption of the edge enhancement operation. Addition of noise would cause the edges to be blurred as the convolution operation is being carried out.

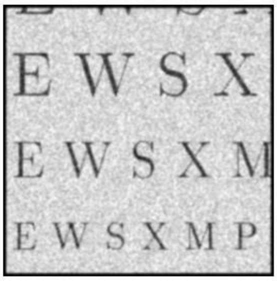
Include your plots from item 3(b) below:



Question 3(b): What is the effect of changing σ?
The impulse response will be more spread out and covers a larger number of samples when sigma increases. The increase of the range of the impulse response will filter out a larger range of noise but further blurs the image out.

Include the images from item 3(c) below

Zblur1



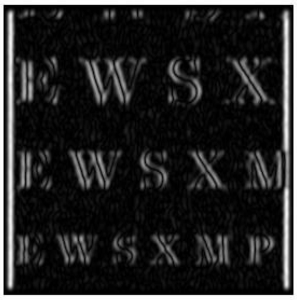
Zblur2

Zblur3

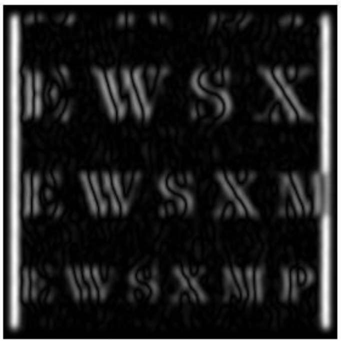
Include the images from item 3(d) below

Zblur1 after h filter

Zblur2 after h filter



Zblur3 after h filter



Question 3(e): How does first applying a Gaussian blur (with different values of σ) affect the performance of subsequent edge enhancement filtering?

When a Gaussian blur is applied, the subsequent edges will become more smoothed out. With higher values of Gaussian blur, the gaussian blur will spread wider which makes the edges much smoother (more refined) but it subsequently blurs the image. This would result in a blur that reduces noise, details and preserving the edges. In a nutshell, the Gaussian blur increases the performance of subsequent edge enhancement filtering.

Code for section 1:

Code from section 1 item 1:

Paste your code for myconv in here.

```
% Written by Tan Jin Chun
% Last Modified : 11/8/2021
% Lab02T01Part1

% The name of this function is myconv
% The purpose of this function is to take four row vectors x, nx (DT
Signal x)
% , h and nh (DT signal h) and carry out the convolution for both of
the
% signals provided.
% The inputs are x, nx, h and nh
% Tne outputs are y and ny

% Assumptions Made
% Assume that nx = ax:bx for integers ax and bx and that nh = ah:bh
for
% integers ah and bh

function [y, ny] = myconv(x, nx, h, nh)

    % Adding the two vectors together
    ny = (min(nx) + min(nh)):(max(nx) + max(nh));

    % Using matlab built in conv() function
    y = conv(x, h);

end
```

Code from section 1 item 2:

Paste your script in here.

```
% Written by Tan Jin Chun
% Last Modified : 11/8/2021
% Lab02T01

clear all;close all;clc

% Question 2(a)
% Calling the dtimpulse() function
[x,nx] = dtimpulse(0, 0, 0);

% Question 2(b)
% Initialising the variable
h = [1:1:3, 3:-1:1];
nh = -2:3;

% Question 2(c)
% Calling the myconv() function
[y, ny] = myconv(x, nx, h, nh)

% Question 2(d)
% Callling thr myconv() function again, this time with the order of
the
% input flipped
[y0, ny0] = myconv(h, nh, x, nx)

% Both of the output has the same values
% This means that the commutative property holds true
% x(n)*h(n) = h(n)*x(n)

% Question 2(e)
% Initialising the variable
x = [1 1 1];
nx = 2:4;

% Output
[y1, ny1] = myconv(x, nx, h, nh)
[y2, ny2] = myconv(h, nh, x, nx)

% Both of the output has the same values
% This means that the commutative property holds true
% x(n)*h(n) = h(n)*x(n)
```

Code for section 2

Code from section 2 item 1:

Paste your echoIR function code here.

```
% Written by Tan Jin Chun
% Last Modified : 11/8/2021
% Lab02
% The name of this function is echoIR()
% The purpose of this function is to create an impulse response with
the
% inputted value of D (in samples) and alpha
% Input: D and alpha
% Output: h (impulse response signal) and nh (time indices)

function [h,nh] = echoIR(D, alpha)

    % Checking the inputted values into the function
    if isscalar(alpha) == 0
        error("The argument must be a scalar")
    end

    % Checking if the value of alpha is between 0 and 1
    if (alpha < 0 || alpha > 1)
        error("The value of alpha must be in between 0 and 1")
    end

    % Method 1
    % Calling the dtimpulse response function
    % d = dtimpulse(0,0,D);
    % ad = dtimpulse(D, 0, D);
    % h = d + alpha * ad;
    % nh = 1:length(h);

    % Method 2
    % Creating a vector
    nh = 0:1:D+100;
    h = (nh == 0).*1 + (nh == D).*alpha;

end
```

Code from section 2 item 2:

Paste your script in here.

```
% Written by Tan Jin Chun
% Last Modified: 11/8/2021
% Lab02T02

clear all;close all;clc

% Question 2(a)
% Initialising the variable
D = 8192 * 0.75;
alpha = 0.5;

% Calling the function
[h,nh] = echoIR(D, alpha);

% Question 2(b)
% Making a stem plot
stem(nh,h);
xlabel("n (time in samples)")
ylabel("h[n]")
title("Impulse Response of Echo System");

% Question 2(c)
% Loading handel
load handel;

% Sound
ny = 0:(length(y)-1);

% Question 2(d)
% Calling the myconv() function
[yecho, nyecho] = myconv(y, ny, h, nh)

% Checking the graph
% figure
% plot(nyecho, yecho);

% Question 2(e)
% Playing the sound
soundsc(yecho, Fs);
pause;

% Decrease D
D = 8192 * 0.1;
alpha = 0.5;

% Calling back the functions
[h, nh] = echoIR(D,alpha);
[yecho, nyecho] = myconv(y, ny, h, nh);
soundsc(yecho, Fs);
pause;

% Decrease alpha
D = 8192 * 0.75;
```

```
alpha = 0.1;


% Calling back the function
[h, nh] = echoIR(D,alpha);
[yecho, nyecho] = myconv(y, ny, h, nh);
soundsc(yecho, Fs);
pause;

% Question 2(f)
D = 8192 * 0.75;
alpha = 0.5;

% Calling back the function and passing the yecho back into the
function
[h, nh] = echoIR(D,alpha);
[yecho, nyecho] = myconv(y, ny, h, nh);
[yecho2, nyecho2] = myconv(yecho, nyecho, h, nh);
soundsc(yecho2, Fs);
```

Code for section 3

Code from section 3.1:

Paste your script in here.

```
% Written by Tan Jin Chun
% Last Modified : 11/8/2021
% Lab02T03

clear all;close all;clc

% Question 3(a)
% Loading an audio file
[trumpet, trumpetFs] = audioread("trumpet.wav");

% Getting the length of the signal and the sampling frequency of the
signal
len_trumpet = length(trumpet);
freq_trumpet = trumpetFs;
soundsc(trumpet);

% Question 2
[hSports, hSportsFs] = audioread('sportscentre.wav');
freq_hSports = hSportsFs;

% Question 2(a)
% Calling the function
trumpetSports = conv(trumpet, hSports);

% Question 2(b)
% Playing the resulting output sound
soundsc(trumpetSports, trumpetFs);

% Question 2(c)
x_trumpet = 0:len_trumpet - 1;
x_trumpetSports = 0:length(trumpetSports) - 1;

% Plotting the subgrids
% New figure
figure;

% The first subplot
subplot(2,1,1);
plot(x_trumpet, trumpet);
title('Anechoic Trumpet Signal');
xlabel('time (samples)');
ylabel('x[n] (Input Audio Signal)');
xlim([0 2101423]);

% The second subplot
subplot(2,1,2);
plot(x_trumpetSports, trumpetSports);
title('Trumpet Sounding As Though Played In A Sports Centre');
xlabel('time (samples)');
ylabel('y[n] (Convoluted Audio Signal)');
xlim([0 2101423]);
```

```
% Question 3
[hCave, hCaveFs] = audioread('cavemono.wav');

% Question 3(a)
% Calling the built-in function
trumpetCave = conv(trumpet, hCave);

% Question 3(b)
% Playing the sound
soundsc(trumpetCave, trumpetFs);
x_trumpetCave = 0:length(trumpetCave) - 1;

% Question 3(c)
% Plotting the graph
% New figure
% The first subgrid
figure;
subplot(2,1,1);
plot(x_trumpet, trumpet);
xlabel('time (samples)');
ylabel('x[n] (input audio signal)');
title('Anechoic Trumpet Signal');
xlim([0 len_trumpet - 1]);

% The second subgrid
subplot(2,1,2);
plot(x_trumpetCave, trumpetCave);
xlabel('time (samples)');
ylabel('y[n] (Convoluted Audio Signal)');
title('Trumpet Sounding As Though Played In A Cave');
xlim([0 len_trumpet - 1]);
```

Code for section 4:

Code from section 4.2:

Paste your script in here.

```
% Written by Tan Jin Chun
% Last Modified : 11/8/2021
% Lab02T04

clear all;close all;clc

% Question 1
% Loading the image matrix echart into the workspace
load echart;

% Question 2
% Displaying the image
imshow(echart);

% Question 3
% Getting the size of the image
% The image is 257 pixels high and 256 pixels wide as shown in the
command
% window
size(echart)

% Question 4
% Adding noise to the image
echartnoisy = echart + 0.8*rand(size(echart));

% Question 5
% Displaying the image
figure
imshow(echartnoisy);

% Question
% Displaying the image again
figure
imshow(echartnoisy,[]);

% One image has a black background and another image has a white
% blackground
% The difference between the image in item5 and item6 is item5
% is normal
% noise added echart image and item6 is the gray image of the noise
added
% image of the echart image matrix
```

Code for section 5:

Code from section 5.1:

Paste your script in here.

```
% Written by Tan Jin Chun
% Last Modified : 11/8/2021
% Lab02T05

clear all;close all;clc

% Question 1
% Loading echart
load echart;

h = [-1 1];
h_transpose = h';

% Question 1(a)
Yrow = conv2(1, h, echart);

% Question 1(b)
Ycol = conv2(h_transpose, 1, echart);

% Question 1(c)
Yboth = conv2(h_transpose, h, echart);

% Question 1(d)
% Displaying the image for Yrow
figure;
imshow(abs(Yrow), []);
title('Yrow');

% Displaying the image for Ycol
figure;
imshow(abs(Ycol), []);
title('Ycol');

% Displaying the image for Yboth
figure;
imshow(abs(Yboth), []);
title('Yboth');

% Question 2
% Getting echartnoisy
echartnoisy = echart + 0.8 * rand(size(echart));

% Question 2(a)
Zrow = conv2(1, h, echartnoisy);

% Question 2(b)
figure;
imshow(abs(Zrow), []);
title('Zrow');

% Question 3
```

```
% Question 3(a)
% Sampling Gaussian Function

% Creating one anonymous function
% Based on the equation from the lab notes
gauss_filter = @(n,sigma) (1./(sqrt(2*pi) * sigma)) .* (exp((-
n.^2)./(2*sigma.^2)));

% Blurring Filters
n1 = -3:1:3;
n2 = -6:1:6;
n3 = -9:1:9;

% Creating signals h1,h2 and h3
h1 = gauss_filter(n1,1);
h2 = gauss_filter(n2,2);
h3 = gauss_filter(n3,3);

% Question 3(b)
figure;

% First subplot
subplot(3,1,1);
stem(n1, h1);
xlabel('n (time samples)');
ylabel('h_1[n]');
title("Impulse Response of Blur with Sigma = 1");

% Second subplot
subplot(3,1,2);
stem(n2, h2);
xlabel('n (time samples)');
ylabel('h_2[n]');
title("Impulse Response of Blur with Sigma = 2");

% Third subplot
subplot(3,1,3);
stem(n3, h3);
xlabel('n (time samples)');
ylabel('h_3[n]');
title("Impulse Response of Blur with Sigma = 3");

% Question 3(c)
Zblur1 = conv2(h1', h1, echartnoisy);
Zblur2 = conv2(h2', h2, echartnoisy);
Zblur3 = conv2(h3', h3, echartnoisy);

% First Image
figure;
imshow(Zblur1,[]);
title('Zblur1');

% Second Image
figure;
imshow(Zblur2,[]);
```

```
title('Zblur2');

% Third Image
figure;
imshow(Zblur3,[]);
title('Zblur3');

% Question 3(d)
Zblur1 = conv2(1, h, Zblur1);
Zblur2 = conv2(1, h, Zblur2);
Zblur3 = conv2(1, h, Zblur3);

% First Image
figure;
imshow(abs(Zblur1),[]);
title('Zblur1 after h filter');

% Second Image
figure;
imshow(abs(Zblur2),[]);
title('Zblur2 after h filter');

% Third Image
figure;
imshow(abs(Zblur3),[]);
title('Zblur3 after h filter');
```