Logbook References

|  |  |
| --- | --- |
| p.15 | Kumar, P. K., Das, S., & Yegnanarayana, B. (2000). One-Dimensional Processing for Edge Detection using Hilbert Transform. *In Proceedings of the Indian Conference on Computer Vision, Graphics and Image Processing,* ICVGIP, 25-31. |
| p.17 | Canny, J. (1986). A Computational Approach to Edge Detection*. IEEE Transactions on Pattern Analysis and Machine Intelligence*, PAMI-8(6), 679–698. doi:10.1109/TPAMI.1986.4767851 |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |

Common used commands

# To run “edges.r”:

my.var <- edges("C:/Users/Victor/Dropbox/University of Strathclyde/Summer Project/Data/edges.csv")

# Loading functions in R:

pathnames <- list.files(pattern="[.]R$", path="C:/Users/Victor/Dropbox/University of Strathclyde/Summer Project/Functions/", full.names=TRUE)  
sapply(pathnames, FUN=source)

# Importing a .csv dataset to a variable in R:

my.data <- data.frame(read.csv("C:/Users/Victor/Dropbox/University of Strathclyde/Summer Project/Data/RED\_000184DE.csv", header = TRUE))

# Exporting a variable to a .csv file in R:

write.csv(my.data, "C:/Users/Victor/Dropbox/University of Strathclyde/Summer Project/Data/name.csv")

# Adding dots to a graph in R:

plot(Pressure.spectrum[[1]], Pressure.spectrum[[2]], type='n', xlab="Frequency [Hz]", ylab="Amplitude", main="Spectrum of Original Pressure Data")

curriculum

lines(Pressure.spectrum[[1]], Pressure.spectrum[[2]], type='l', xlab="Frequency [Hz]", ylab="Amplitude", main="Spectrum of Original Pressure Data")

# Plotting a rectangle in Matlab:

rectangle('Position',[0.59,0.35,3.75,1.37],...

'Curvature',[0.0,0.0],...

'LineWidth',2,'LineStyle','--')

daspect([1,1,1])

# Filter design in Matlab:

% Example 1:

% For data sampled at 1000 Hz, design a 9th-order highpass

% Butterworth filter with cutoff frequency of 300Hz.

Wn = 300/500; % Normalozed cutoff frequency

[z,p,k] = butter(9,Wn,'high'); % Butterworth filter

[sos] = zp2sos(z,p,k); % Convert to SOS form

h = fvtool(sos); % Plot magnitude response

% Example 2:

% Design a 4th-order butterworth band-pass filter which passes

% frequencies between 0.15 and 0.3.

[b,a]=butter(2,[.15,.3]); % Bandpass digital filter design

h = fvtool(b,a); % Visualize filter

filtered\_averaged\_pressure = (filter(b,a,average\_pressure))

ylabel('abs(Pressure)');

xlabel('Time [0.1 s]');

title('Edge detector over Absolute Filtered (2) Averaged Pressure Data');