# Abstract

The goal of this lab is to transform a grayscale image into a halftone image. This MATLAB script includes a halftone function that creates matrices containing black and white pixels and uses them to replace pixels in a grayscale image whose values vary. This method outputs an image that appears very similar to the input grayscale image, but only uses black and white pixels and no values in between. This report will discuss the techniques used to successfully create halftone transformations and what each output image tells us about the performance of the program.

# Technical Discussion

## How Images are Loaded into the Program

The program starts by reading an image using the imread() function. The images that are used in this lab include “Fig0225(a)(face).tif”, “Fig0225(b)(cameraman).tif”, and “Fig0225(c)(crowd).tif”. The image files are converted to the uint8 type and assigned to a matrix, as shown below in Figure 1.

A picture containing text

Description automatically generated

Figure 1: Importing images and type casting to type uint8

After the matrices are created, they are sent to the halftone function as arguments where they will be processed by an algorithm that performs the halftone transformation. In Figure 2, matrices are set equal to the result of the halftone function. The result is a matrix of type logical.

Text

Description automatically generated

Figure 2: Assigning matrices as the halftone function’s resulting transformation