University Of The Punjab

Gujranwala Campus

Department Of Information Technology

Computer Vision Assignment

Matlab code execution

Prepared By:

Alisha Naeem

Roll No: BIT21013

7th Semester (Morning)

Submitted To:

Maam Fouqia

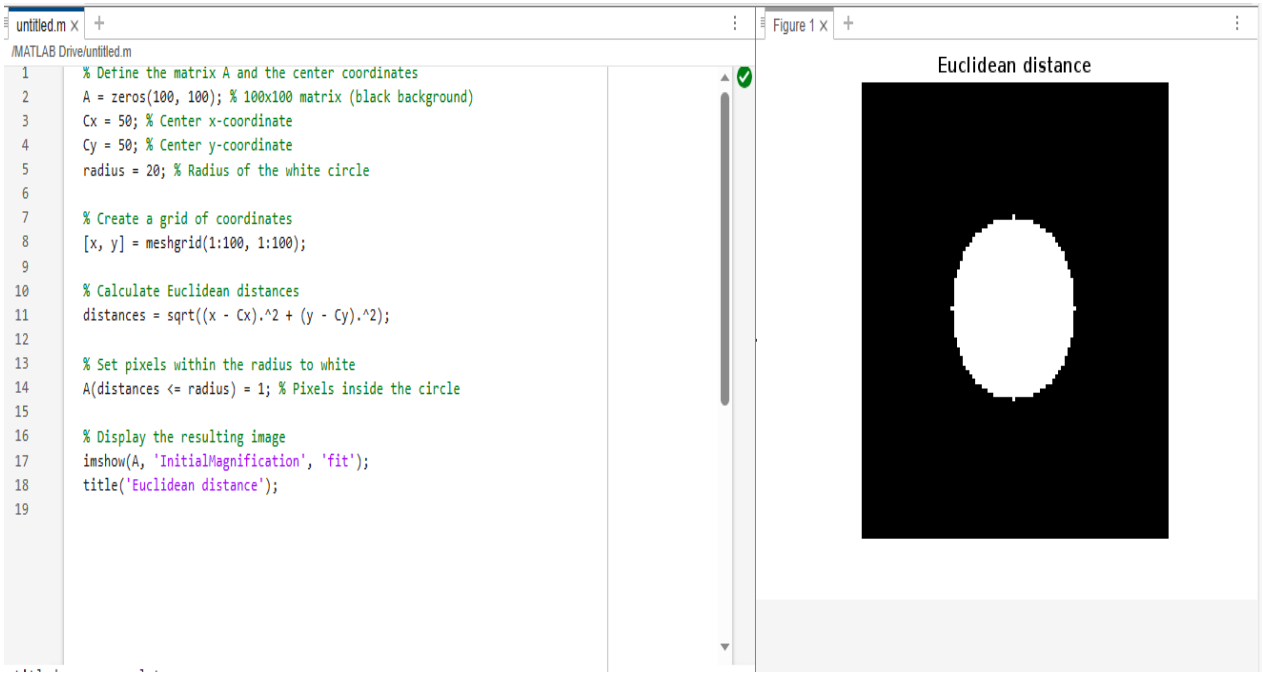
Dated:

01/20/2024

Code # 1: Euclidean distance

```
% Define the matrix A and the center coordinates
A = zeros(100, 100); % 100x100 matrix (black background)
Cx = 50; % Center x-coordinate
Cy = 50; % Center y-coordinate
radius = 20; % Radius of the white circle
% Create a grid of coordinates
[x, y] = meshgrid(1:100, 1:100);
% Calculate Euclidean distances
distances = sqrt((x - Cx).^2 + (y - Cy).^2);
% Set pixels within the radius to white
A(distances <= radius) = 1; % Pixels inside the circle
% Display the resulting image
imshow(A, 'InitialMagnification', 'fit');
title('Euclidean distance');
```

Output:

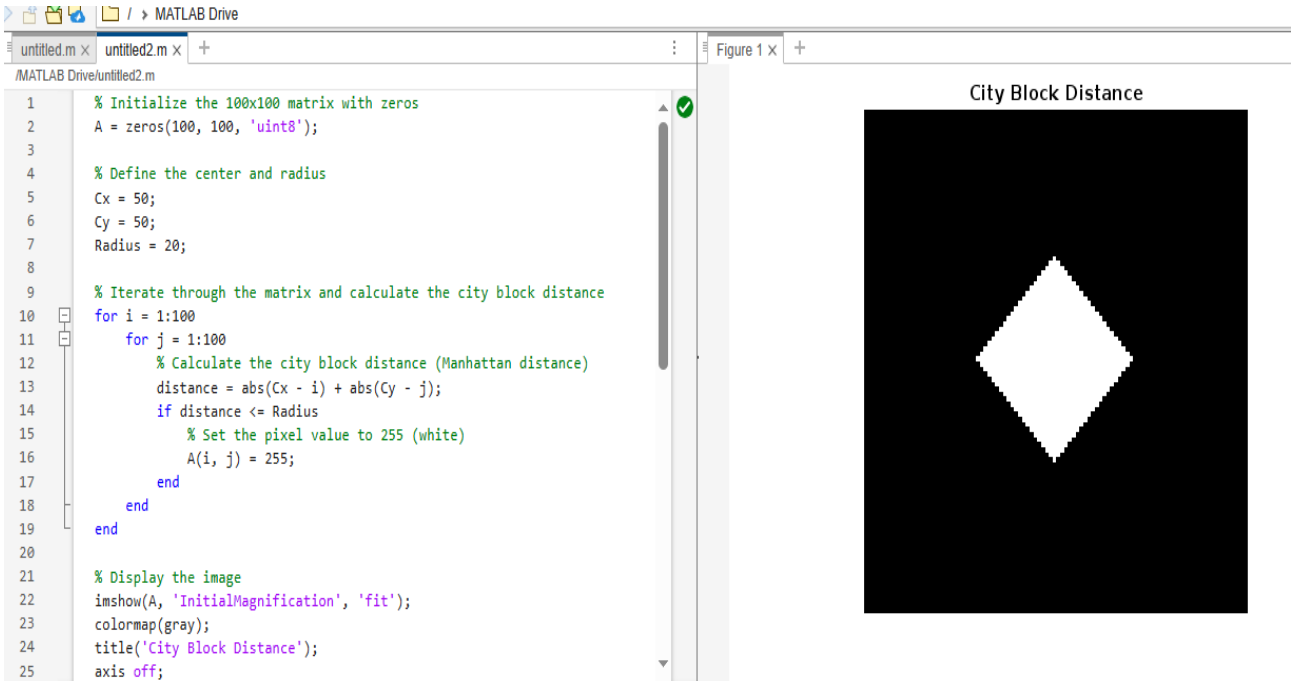


Code # 2: City Block Distance

```
% Initialize the 100x100 matrix with zeros
A = zeros(100, 100, 'uint8');
% Define the center and radius
```

```
Cx = 50;
Cy = 50;
Radius = 20;
% Iterate through the matrix and calculate the city block distance
for i = 1:100
   for j = 1:100
      % Calculate the city block distance (Manhattan distance)
      distance = abs(Cx - i) + abs(Cy - j);
      if distance <= Radius
         % Set the pixel value to 255 (white)
         A(i, j) = 255;
      end
   end
end
% Display the image
imshow(A, 'InitialMagnification', 'fit');
colormap(gray);
title('City Block Distance');
axis off;
```

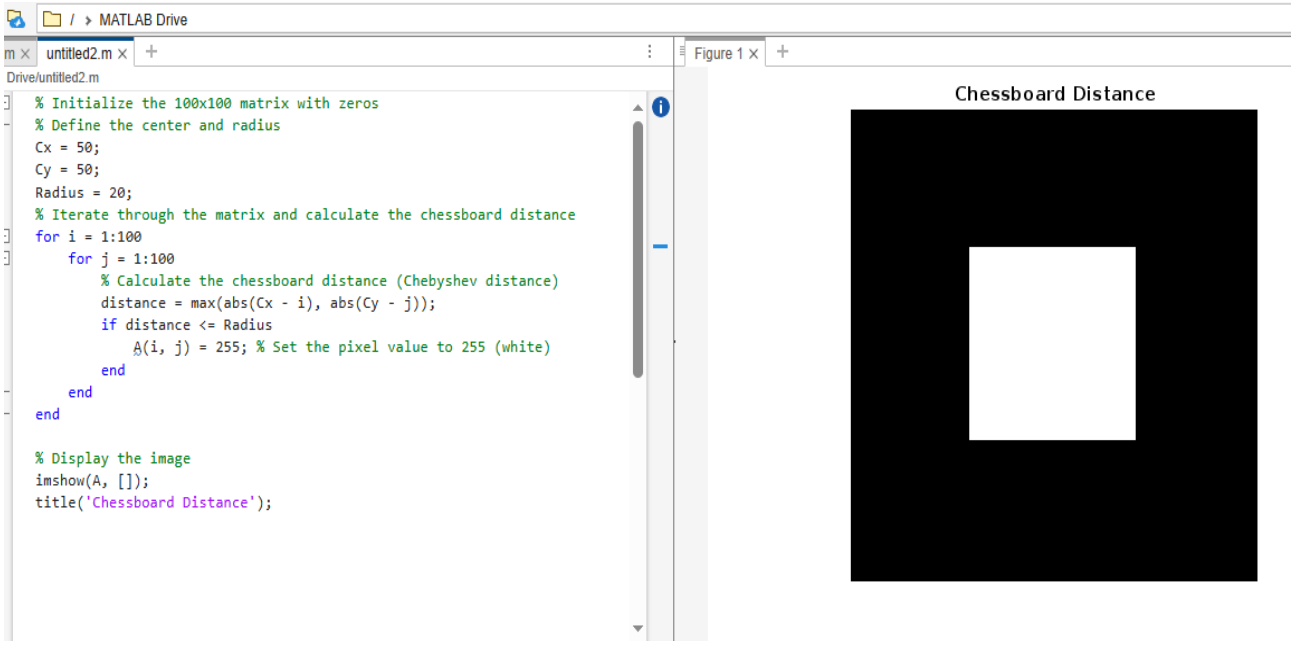
Output:



Code # 3: Chessboard Distance

```
% Initialize the 100x100 matrix with zeros
% Define the center and radius
Cx = 50;
Cy = 50;
Radius = 20;
% Iterate through the matrix and calculate the chessboard distance
for i = 1:100
   for j = 1:100
      % Calculate the chessboard distance (Chebyshev distance)
      distance = max(abs(Cx - i), abs(Cy - j));
      if distance <= Radius
         A(i, j) = 255; % Set the pixel value to 255 (white)
      end
   end
end
% Display the image
imshow(A, []);
title('Chessboard Distance');
```

Output:



Code # 4: Diamond Shape (Manhattan Distance)

```
A = zeros(100, 100, 'uint8');   % Initialize the 100x100 matrix with zeros
Cx = 50;     % Define the center and radius
Cy = 50;
Radius = 20;
for i = 1:100    % Iterate through the matrix and calculate the Manhattan distance
   for j = 1:100
      distance = abs(Cx - i) + abs(Cy - j);    % Calculate the Manhattan distance
      if distance <= Radius
         A(i, j) = 255;       % Set the pixel value to 255 (white)
      end
   end
end
imshow(A, 'InitialMagnification', 'fit');  % Display the image
colormap(gray);
title('Diamond Shape (Manhattan Distance)');
axis off;
```

Output: