function batchProcessImages()

% Let the user select multiple image files

[fileNames, filePath] = uigetfile({'\*.png;\*.jpg;\*.jpeg', 'Image Files (\*.png, \*.jpg, \*.jpeg)'}, 'Select image files', 'MultiSelect', 'on');

% If the user cancels, exit the function

if isequal(fileNames, 0)

disp('User canceled the operation');

return;

end

% If only one file is selected, convert it to a cell array with one element

if ischar(fileNames)

fileNames = {fileNames};

end

% Iterate over each selected image file

for i = 1:length(fileNames)

% Read the image

img = imread(fullfile(filePath, fileNames{i}));

% Display the image

imshow(img);

title('Drag to create a rectangular region');

% Create an interactive rectangle object using imrect

h = imrect;

% Wait for the user to finish drawing; press delete when done

wait(h);

% Get the position of the rectangle

rectPosition = getPosition(h);

% Create a mask based on the rectangle position

rectMask = createMask(h);

% Close the figure window

close(gcf);

% Extract the channel information from the selected region

selected\_region = img .\* uint8(rectMask);

% Perform channel analysis on the selected region (using grayscale here)

intensity = double(rgb2gray(selected\_region));

% Create X and Y grids

[rows, cols] = size(intensity);

[X, Y] = meshgrid(1:cols, 1:rows);

% Flip the intensity data horizontally

intensity\_flipped = flip(intensity, 2);

% Plot a 3D figure

hFigure = figure;

% Set the size of the figure window

screenSize = get(0, 'ScreenSize');

figureWidth = screenSize(3) \* 0.6;

figureHeight = screenSize(4) \* 0.8;

set(hFigure, 'Position', [screenSize(3)/2 - figureWidth/2, screenSize(4)/2 - figureHeight/2, figureWidth, figureHeight]);

% Create a progress bar

hWaitbar = waitbar(0, 'Processing...', 'Name', 'Progress Bar');

% Update the progress bar

waitbar(i/length(fileNames), hWaitbar, sprintf('Processing... (File %d/%d)', i, length(fileNames)));

% Plot a 3D figure

surf(X, Y, intensity\_flipped);

axis auto;

shading interp;

colormap("jet");

view(-104, 76);

xlabel('X');

ylabel('Y');

zlabel(''); % Set the z-axis label to an empty string to hide it

colorbar('Location', 'SouthOutside');

% Add contour lines

hold on

contour3(X, Y, intensity\_flipped, 10, 'k', 'LineWidth', 0.5);

hold off

% Update the progress bar

waitbar((i+1)/length(fileNames), hWaitbar, sprintf('Processing... (File %d/%d)', i+1, length(fileNames)));

% Add legend and title

[~, baseFileName, ~] = fileparts(fileNames{i});

title(strrep([baseFileName, '\_matlab'], '\_', ' ')); % Replace underscores with spaces

xlabel('X');

ylabel('Y');

% Enable interactive 3D rotation

rotate3d on;

% Update the progress bar

waitbar(1, hWaitbar, 'Processing complete');

% Get the current figure title

figureTitle = get(get(gca,'title'), 'String');

% Use the title as the file name and add a .png extension

outputFileName = [figureTitle, '.png'];

% Save the figure as a PNG file

saveas(hFigure, outputFileName);

% Close the progress bar

close(hWaitbar);

end

end