```
% % Image preprocessing

ImagePath = '/Users/macintosh/Desktop/Engineering/Image Process/
images_DE';
FileExt = fullfile(ImagePath,'*.jpg');
ImageData = dir(FileExt); % Preallocating the directory for image data.


% different folder
Image_Preprocessed = '/Users/macintosh/Desktop/Engineering/Image Process/
images_DE/Image_Preprocessed';
if ~exist(Image_Preprocessed,"dir") % allocating directory to terminate the
process to retrieve images.
   mkdir(Image_Preprocessed); % Generating directory
end

for X = 1:length(ImageData) % Initiating loop
   File_Name = ImageData(X).name;
   Full_File = fullfile(ImagePath,File_Name); % loading the full file to repeat the
code for each and every image one by one
   Array_Image = imread(Full_File); % Reading the directory

   Image_Processed = Array_Image;
   imshow(Image_Processed);
   Image_Resized = imresize(Image_Processed,[500,500]); % Transforming the
images into 500x500 pixels
   imshow(Image_Resized);
   Image_Gray = rgb2gray(Image_Resized); % Transforming the images from
RGB to gray scale
   imshow(Image_Gray);
   imwrite(Image_Gray,fullfile(Image_Preprocessed,
['Preprocessed_',File_Name])); % Writing the whole file data and transfering to
images in other foldder
   Denoised_Image = imgaussfilt(Image_Gray,2); % Apply the denoise filter
from gray image
   imshow(Denoised_Image);
   imwrite(Denoised_Image,fullfile(Image_Preprocessed,
['Denoised_',File_Name])); % % Writing the whole file data and transfering to
images in other foldder
end

% PART 2
% %Feature Annotation

% Define image data using cell arrays
Annotations = {
   'https://github.com/Nkukadiya05/Preprocessed_images/blob/main/
```

```
Albatros.jpg', {'Bird','Blue','Albatros'}, 'A big bird flying on the sea';
    'https://github.com/Nkukadiya05/Preprocessed_images/blob/main/
Avocado.jpg', {'Fruit','Avocado','Healthy'}, 'Most favourite creamy fruit';
    'https://github.com/Nkukadiya05/Preprocessed_images/blob/main/
BlueJay.jpg', {'Bird','Blue','Bluejay'}, 'A beautiful bird sitting on the branch';
    'https://github.com/Nkukadiya05/Preprocessed_images/blob/main/Boat.jpg',
{'Boat','Wood','Water'}, 'A wooden boat floating on the water';
    'https://github.com/Nkukadiya05/Preprocessed_images/blob/main/Cat.jpg',
{'Animal','Cat','Tiny'}, 'A blue-eyed cat staring something';
    'https://github.com/Nkukadiya05/Preprocessed_images/blob/main/
DragonFruit.jpg', {'Fruit','Red','DragonFruit'}, 'Red coloured very healthy fruit';
    'https://github.com/Nkukadiya05/Preprocessed_images/blob/main/
Flamingo.jpg', {'Animal','Pink','Flamingo'}, 'Pink animal on the wallpaper';
    'https://github.com/Nkukadiya05/Preprocessed_images/blob/main/Fox.jpg',
{'Animal','Wild','Fox'}, 'Wild Fox in the snow';
    'https://github.com/Nkukadiya05/Preprocessed_images/blob/main/
Grapes.jpg', {'Fruit','Juicy','Grapes'}, 'Very juicy fruit hanging with the shrub';
    'https://github.com/Nkukadiya05/Preprocessed_images/blob/main/Hen.jpg',
{'Bird','Hen','White'}, 'A red-beaked bird staring something';
    'https://github.com/Nkukadiya05/Preprocessed_images/blob/main/Lady.jpg',
{'Human','Lady','Dress'}, 'A beautiful lady standing on the heels';
    'https://github.com/Nkukadiya05/Preprocessed_images/blob/main/
Orange.jpg', {'Fruit','Orange','Juicy'}, 'A very juicy orange fruit';
    'https://github.com/Nkukadiya05/Preprocessed_images/blob/main/Owl.jpg',
{'Bird','Owl','Big'}, 'A big white bird sitting';
    'https://github.com/Nkukadiya05/Preprocessed_images/blob/main/
PassionFruit.jpg', {'Fruit','Passion','Sweet'}, 'A very sweet fruit in the bowl';
    'https://github.com/Nkukadiya05/Preprocessed_images/blob/main/
Peacock.jpg', {'Bird','Peacock','Big'}, 'A very big colourful bird';
    'https://github.com/Nkukadiya05/Preprocessed_images/blob/main/
Pigeon.jpg', {'Bird','Pigeon','Gray'}, 'A blue-necked bird';
    'https://github.com/Nkukadiya05/Preprocessed_images/blob/main/
Pixabay.jpg', {'Bird','Pixabay','Hue'}, 'A beautiful bird sitting on the tree';
    'https://github.com/Nkukadiya05/Preprocessed_images/blob/main/
Pomegranate.jpg', {'Fruit','Pomegranate','Red'}, 'Very tasty fruit on the tree';
    'https://github.com/Nkukadiya05/Preprocessed_images/blob/main/
Shoes.jpg', {'Shoes','Legs','White'}, 'A lady wearing white shoes';
    'https://github.com/Nkukadiya05/Preprocessed_images/blob/main/
Spinach.jpg', {'Veg','Spinach','Green'}, 'Very healthy leafy vegetable';
    'https://github.com/Nkukadiya05/Preprocessed_images/blob/main/
Sunflower.jpg', {'Flower','Sunflower','Yellow'}, 'A big yellow beautiful flower';
    'https://github.com/Nkukadiya05/Preprocessed_images/blob/main/Swan.jpg',
{'Bird','Swan','White'}, 'A big white bird in the snow';
    'https://github.com/Nkukadiya05/Preprocessed_images/blob/main/
Tomato.jpg', {'Veg','Tomato','Red'}, 'A ripe red tomato ready to be picked';
    'https://github.com/Nkukadiya05/Preprocessed_images/blob/main/Tram.jpg',
{'Travel','Tram','Small'}, 'A yellow tram on the street';
```

```
    'https://github.com/Nkukadiya05/Preprocessed_images/blob/main/Yacht.jpg',
{'Boat','Yacht','White'}, 'A white yacht on the water'
};

metadataFolder = 'metadata';
if ~exist(metadataFolder, 'dir') % Preallocating the directory to save metadata if
it doesnt exist alreay
    mkdir(metadataFolder); % Generating directory
end


for i = 1:size(Annotations, 1) % Initiating the loop to display the annotation
    metadata = struct('ImageID', Annotations{i, 1}, 'Keywords', Annotations{i, 2},
'Description', Annotations{i, 3}); % Setting the structure to display vertically.
    metadataFileName = fullfile(metadataFolder, sprintf('metadata_%d.mat', i));
% Generating the different folder to save metadata  and split them into the 25
differen files.
    save(metadataFileName, 'metadata'); % Saving the metadata into the
different file and allocating the data in the specific file.
end


metadata = cell(size(Annotations, 1), 1); % initialising the metadata with cell
array


for i = 1:size(Annotations, 1) % initialising the loop
    metadata{i} = struct('ImageID', Annotations{i, 1}, 'Keywords', Annotations{i,
2}, 'Description', Annotations{i, 3}); % Populating the metadata with the
stucture for each image one by one
end
disp(metadata); % Display the metadata


% % Feature Extraction

imageFiles = {'Albatros.jpg', 'Avocado.jpg', 'BlueJay.jpg', 'Boat.jpg', 'Cat.jpg',...
    'DragonFruit.jpg', 'Flamingo.jpg', 'Fox.jpg', 'Grapes.jpg', 'Hen.jpg',...
    'Lady.jpg', 'Orange.jpg', 'Owl.jpg', 'PassionFruit.jpg', 'Peacock.jpg',...
    'Pigeon.jpg', 'Pixabay.jpg', 'Pomegranate.jpg', 'Shoes.jpg','Spinach.jpg',...
    'Sunflower.jpg','Swan.jpg','Tomato.jpg','Tram.jpg','Yacht.jpg',};

outputFolder = 'Images_Preprocessed'; % Specifying the folder

if ~exist(outputFolder, 'dir') % Creating the directory folder if it doesn't already
exist
    mkdir(outputFolder);
```

```
end

for i = 1:length(imageFiles) % Initialising the loop through each image
    currentImage = imread(imageFiles{i}); % Reading the images
    figure(i);
    imshow(currentImage); % Displaying the images

    if size(currentImage, 1) > 500 || size(currentImage, 2) > 500 % Resizing the
images
        currentImage = imresize(currentImage, [500, 500]);
    end

    % Extracting the colour moments for each image
    meanRed = mean2(currentImage(:,:,1));
    stdDevRed = std2(currentImage(:,:,1));
    meanGreen = mean2(currentImage(:,:,2));
    stdDevGreen = std2(currentImage(:,:,2));
    meanBlue = mean2(currentImage(:,:,3));
    stdDevBlue = std2(currentImage(:,:,3));

    % Creating the table to save the moments
    momentsTable = table(...
        meanRed, stdDevRed, meanGreen, stdDevGreen, meanBlue,
stdDevBlue, ...
        'VariableNames',{'Mean_Red','StdDev_Red','Mean_Green','StdDev_Green',
'Mean_Blue','StdDev_Blue'});

    grayImage = rgb2gray(currentImage); % Transforming the original RGB
images to gray
    figure(i + length(imageFiles)); % Numbering the images
    imshow(grayImage);
    title(['Gray ', imageFiles{i}(1:end-4)]); % Giving the name to images from 1 to
25

    Wavelength = 4; % Setting the Wavelength
    orientation = 45; % Setting the Orientation range
    [mag, phase] = imgaborfilt(grayImage, Wavelength, orientation); % Applyng
the Gabor filter

    figure(i + 2 * length(imageFiles)); % Displaying the gabor filter result images
    subplot(1,3,1); % Setting the direction in the figure
    imshow(grayImage);
    title('GrayScale Image');

    subplot(1,3,2); % Setting the direction in the figure
    imshow(mag, [0, 555]); % setting the magnitude for gabor filter
    title('Gabor Magnitude');
    subplot(1,3,3); % Setting the direction in the figure
    imshow(phase); % Setting the phase for gabor filter
    title('Gabor Phase');

    subsetMag = mag(1:10, 1:10);  % Adjusting the indices to 10:10

    ConnectedComponents = bwconncomp(grayImage); % Displaying the
properties of image
    propertiesCC = regionprops(ConnectedComponents); % Displaying the
shape feature of the image

    % Creating the structure for the images to display metadata
    currentImageStruct = struct(...
        'Image', i, ...
        'ColorMoments', struct(...
            'Mean_Red', meanRed, 'StdDev_Red', stdDevRed, ...
            'Mean_Green', meanGreen, 'StdDev_Green', stdDevGreen, ...
            'Mean_Blue', meanBlue, 'StdDev_Blue', stdDevBlue), ...
        'GaborResults', struct(...
            'Magnitude', subsetMag, 'Phase', phase), ...
        'ConnectedComponents', struct(...
            'Centroid', cat(1, propertiesCC.Centroid)));

    jsonStrCurrentImage = jsonencode(currentImageStruct); % Encoding the
JSON file
    jsonFilenameCurrentImage = fullfile(outputFolder, ['image_' num2str(i)
'_features.json']);  % Creating the JSON file and saving them in the different
folder
    fidCurrentImage = fopen(jsonFilenameCurrentImage, 'w'); % Opening the
JSON file
    fprintf(fidCurrentImage, '%s', jsonStrCurrentImage); % Saving the image
features to JSON file in the specified folder
    fclose(fidCurrentImage); % Closing and saving the file in the folder
end
```