

EE 604: Digital Image Processing

Course Projects

Please read the instructions carefully:

- Here is the [link](#) for selecting projects.
- The course project weightage is 20%.
- The spreadsheet is comment only. So please add your name in the comment section next to the project you would like to work on. Each project should have a maximum of two presenters.
- The project allocation is on a first-come-first-serve basis.
- Once a project is claimed by a group (or an individual) (i.e. there are existing comments in column 'B' and 'C'), you are not allowed to overwrite it. In case such a situation is noted, the individuals violating it will be allotted a project of our choice. Also, each group (or individual) can claim only one project.
- The deadline for choosing your project and hence commenting on the spreadsheet is 11:59 PM on 29th September. Failing to do so, we will take the liberty to form groups at random and assign projects on our own volition. It is highly advised to contact the TAs and discuss the project topic(s), either before or after choosing.
- The final evaluation of the completed project will be done tentatively in the 1st week of November (announcements will be made 2 weeks in advance).
- Proper credit(s) must be given if you are using part(s) of some other work in your project. In the case of plagiarism, zero marks will be assigned.

1. Red-eye correction :

You are given a picture corrupted with 'red eye' or take your own picture. Your program should be able to make the eye color black. The link to download the image is [here](#).

2. HoG feature extraction :

Input will be a picture of a person and output will be an HoG (Histogram of Oriented Gradients) features of the picture. (You may find this video useful: <https://www.youtube.com/watch?v=4ESLTAd3IOM>)

3. Face detection using SVM :

Take some dataset of 'face' and 'non-face' images. Extract features and perform SVM (support vector machine) to classify the image as 'face' or 'non-face'.

4. Face recognition using eigenface :

Take some dataset of 'face' images. Find eigenfaces and use those to recognize. (You may find this video useful: <https://www.youtube.com/watch?v=jQOZrXZTXcw>)

5. Handwritten digit recognition :

Load mnist dataset. It is a dataset comprising of 60k training examples and 10k test examples of handwritten digits. There are 10 classes (0-9). Train a neural network to identify digits.

(Advised to use python, you may use Keras. Two class classification is sufficient, in that case, take '0' and '1'. You may find this link useful: <https://keras.io/#getting-started-30-seconds-to-keras>).

6. Counting Coins :

Take images with (at least) two different kinds of coins. Your program should be able to detect and extract the count of coins in each category which can be touching and/or overlapping.

7. Structural Similarity Index (SSIM) :

Degrade an image with Gaussian noise and impulse noise. Calculate the SSIM index with respect to the reference image.

8. Image stitching :

Take two images of the same place i.e. two images have sufficient overlap region. Stitch those images to get one panoramic image. The link to download the image is [here](#).

9. Blend one image into another :

For a given pair of images blend one image into another i.e. mix two images so that they combine together as a mass, the one like given [here](#).

10. Count scattered matchsticks :

Take images (by your mobile camera) of scattered matchsticks and count the number of random scatter of matchsticks.

11. Hand gesture recognition :

Build a model to recognize different hand gestures using real-time data from the webcam.

12. Fingertip detection in a video :

The objective is to create an application that will detect the tips of the finger in a video clip. (Sample videos will be provided for testing and development).

13. Blur background of an image :

Make a GUI application that takes some user input defining foreground and background in an image and blur the background.

14. Human Iris segmentation and analysis :

The MATLAB GUI application will take the image of an individual and segment out the iris from the eye and present a color analysis of the same (Percentage of blue, black, brown, gray, green, red).

15. Human Iris segmentation and editing :

The MATLAB GUI application will take the image of an individual and segment out the iris from the eye and add functionality to change the color to any of the mentioned colors: blue, black, brown, gray, green, red, which would be provided as input.

16. License Plate Detection :

Implement an algorithm to detect the license plate in an image, if present.

17. Image Compression using Run Length Encoding :

Implement RLE based image compression and show results on images of Baboon, Lena, and Pepper.

18. LSB Steganography :

Implement LSB Steganography on pairs of images from images of Baboon, Lena, and Pepper. Results include PSNR and MSE of steganography with respect to cover.

19. Haze/fog removal from images :

The MATLAB GUI application must take input of an image of a hazy or foggy scene and give a clear image as output. The level of haze/fog must be moderate so the background information is not lost in the original image.

20. DCT Image Compression (used in JPEG) :

Implement DCT based image compression and show results on images of Baboon, Lena, and Pepper. Compare the results with standard JPEG compression.

21. Measure the Diameter of an Object in an Image :

Implement the segmentation of the object and calculate its diameter (the object will be circular and exactly one in count).

22. Geometric Shape Detection :

Implement a program to recognize geometrical shaped objects like circles, rectangles, and squares from the input image.

23. Heat Map Generation (pseudo coloring) :

Implement heatmap generation (based on color) on the image.

24. Foreground extraction by focus :

Write an algorithm to compute the depth map of a scene. Here you have to extract the foreground by focus on the given image and compute its depth.

25. Head-Pose Detector :

Develop an application using MATLAB GUI which will receive the image of an individual as input, detect the face/head and also be able to turn the face/head slightly in all four directions according to user input (make four keys in the GUI to input which direction to turn the head).

26. Cartooning of an Image :

Input will be an image and produce the output which will be turned into a cartoon image.

27. OMR Scanning and Evaluation :

Implement an algorithm to scan the OMR sheet, evaluate it and print the results.

28. Extraction of Text from Images :

To detect and extract characters from the scanned documents/images. To make editable documents from existing paper documents or image files. Also considering detecting text and extracts them regardless of the orientation is proposed, if possible.

29. Removal of High-Density Salt-and-Pepper Noise :

Detect noisy pixels in an image and denoise it using any proposed filters. Also, compute the Peak Signal-to-Noise Ratio (PSNR).

30. Motion deblurring by pseudoinverse :

Here, one has to implement the pseudo-inverse filter. Take a blurred input image and deblur it using the pseudo-inverse technique.

31. Real-time Edge Detection in a video :

To perform edge detection of images in real-time (use popular canny edge detection to detect a wide range of edges in images).

32. Background and foreground detection of real-time moving object :

There's will be a real-time video accessed using a webcam. Perform the detection and extraction of the moving objects from the video.

33. Counting the number of fingers :

Count the given numbers on your fingers and find the correct finger on which the number ends.

34. Shadow Removal :

The objective is to remove shadows or non-uniform illumination from faces in a portrait. Develop an application that will take an image of an individual with shadows cast over the face and the expected output is the image with the face uniformly illuminated or shadows removed.

35. Template matching :

To find an object in an image using Template Matching. Here, you have to find the patch/template in an image i.e. the location of a template image in a large crowd image. Take two input images: Source Image which is the image to find the template in and Template Image which is the image that is to be found in the source image.

36. Foreground and Background Separation using flash :

The objective is to perform the extraction of the foreground layer using flash/no-flash image pairs, which we call flash cut. Flash cut is based on the simple observation that only the foreground is significantly brightened by the flash and the background appearance change is very small if the background is distant.

37. Changing the Background of a Subject :

Develop an application that will receive two input images from the user, crop/segment the subject/person in the first image and incorporate the subject in the second image. Here, the purpose is to change the background of the subject in the image.

38. Image Forgery Detection :

Here you to detect a forgery in a given image that is tampered and manipulated. Develop techniques that verify the integrity and authenticity of digital images. Some known most common type of image forgery is a copy-move forgery in digital image or exposing and detecting duplicated image regions.

39. Eye-tracking in real-time video :

Develop an eye-tracking application which will take the real-time video as input through webcam and in the video frame, mark the position at which the eye of the subject is looking at.

40. Clouds segmentation :

Perform segmentation of cloud in a given image.