# ELL715: Assignment-5
# Facial image analysis

**Instructor**: Prof. Monika Agrawal
Prowfessor, C.A.R.E., Indian Institute of Technology Delhi
**Total Marks: 260 marks (160 + 100 (bonus))**
**Deadline**: 21st Nov'25

October 29, 2025

## Instructions

- This assignment can be done in teams. Your team should comprise of $\leq$ smallest even natural number of members.

- Entire AI generated submissions shall attract a zero score in the assignment. You may use agents for partial coding and reference. You'll have to specify at what all places agentic coding has been employed.

- You'll have to submit the code and an informal report. Your code repository and reported results should be reproducible. You'll have to attach a `README` file which should have instructions to run the code.

- Both report and code has to zipped inside a zip file. The zip file should be named as per the following scheme: $< \text{Entry no. student\#1} > - < \text{Entry no. student\#2} > \_ < \text{Name of student\#1} > - < \text{Name of student\#2} >.$

- `Python` is recommended language for implementation. You are free to use utilize any open source package/library of your choice.

- If you have any queries, you can meet the TAs in their office hours.

- We will conduct demos for grading.

- We shall run `moss` for plagiarism check. Based upon the amount of plagiarism adequate penalty will be applied.

- Grading will entirely be on how much efforts you have put in. We will keep correctness secondary.

## 1 Introduction

This assignment has two parts, the first one is mandatory and the second is bonus. In the first part you are required to implement Viola Jones algorithm from scratch for face detection. While in the second part, you are required to present a comparative analysis between two face identification algorithms. You should use *Faces94* dataset.

**Viola Jones paper:** https://www.cs.cmu.edu/ efros/courses/LBMV07/Papers/viola-cvpr-01.pdf

`Link to` *Faces94* `dataset:` **Dataset**

## 2 Part-1: Viola-Jones Face Detector *(160 marks)*

You are supposed to implement the famous Viola-Jones face detector from scratch. You can use external libraries for implementing sub-functionalities and matrix manipulations, but direct usage of functions/classes is strictly prohibited (except for reading the images). Implement the following:

1. **Dataset generation**: There are three folders in the dataset, use the `maleStaff` and `female` for training Viola-Jones classifier, and `male` for testing. For ground truths, you can select $16 \times 16$ patch from the center of the image. This will serve as the 'face' class. Then, you can extract 5 other $16 \times 16$ random patches from the image. These will tagged as 'not-a-face' class. Repeat this process for all the images in the training and testing set. *(20 marks)*

2. **Haar features**: Next, Haar features are to be extracted. For this consider: horizontal, vertical, and diagonal Haar filters of multiple scales. *(20 marks)*

3. **Integral image**: To extract Haar features quickly, you should implement Integral image based feature extraction. Refer the Viola Jones paper for more details. *(20 marks)*

4. **Adaboost algorithm**: Next to classify an image as 'face' or 'not-a-face' implement Adaboost algorithm from scratch. *(40 marks)*

5. **Cascade of classifiers**: Finally, arrange these classifiers in a cascade, as described in the original paper. *(20 marks)*

Following are the deliverables *(40 marks)*:

1. Final test accuracy.

2. Face detection results on a couple of images with multiple faces. You can use images from internet.

3. A well documented codebase and an informal report. Put up all the results in the report.

# 3   Part-2 (Bonus): Face Identification Results *(100 marks)*

This is a bonus part. For the this part, you should use initial 75% images for each subject in `maleStaff` & `female` folders as the gallery. Remaining 25% images per subject are to be used as probes. After the data split, implement the following:

1. **EigenFaces**: Implement the `EigenFaces` algorithm. You are free to use any external package for `PCA`. *(40 marks)*

2. **Wavelets**: This part is open ended. Use different wavelets to construct a feature vector per image. You are free to use any wavelet configuration, and can implement them with any external package. An example of good texture based wavelets are `Gabor` wavelets. *(40 marks)*

3. Finally, compare the identification performance of the two methods. Try plotting `tSNE` plots of the gallery feature spaces so as to understand subject level separability. *(20 marks)*

As mentioned in the instructions, grading will first account for efforts, and then correctness! All the best!