



Indian Institute of Science Education and Research
Bhopal
Computer Vision(DSE-312/EECS-320)
Assignment-2

Name:

Roll No.:

Time of submission:

Marks Obtained:

Please follow the instructions given in the assignment carefully.

Please provide your detailed answers and any explanations or diagrams directly below each question in the ‘Answers’ section.

1. Implement a face detection algorithm from scratch using Haar-like features and Integral Image computation. (**Marks: 1+2+7**)

- Capture an image of yourself using a webcam or upload a face image.
- Compute the Integral Image to efficiently calculate pixel sums over rectangular regions. Use integral image to detect face using Haar features.

Answer:

Computing Integral Image

I have computed integral image using the following algorithm

$$II[i, j] = I[i, j] + II[i - 1, j] + II[i, j - 1] - II[i - 1, j - 1]$$

where I is the original image and II is the integral image.

Haar-like features

I have used the four type of Haar-like features as shown in the figure below.

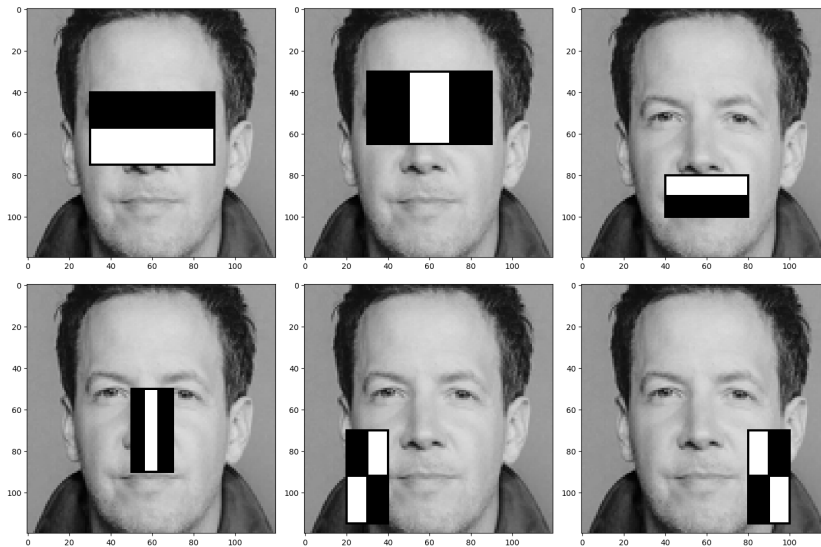


Figure 1: Haar-like features

Face Detection

I have selected first, second and fourth Haar-like features to detect face. I have used the following algorithm to detect face.

- Check if the first haar-like feature evaluate to > 1.8
- Check if the second haar-like feature evaluate to $> .8$
- Check if the fourth haar-like feature evaluate to $> .35$

If all the above conditions are satisfied then the face is detected, else the block is rejected.

Results

Applied my face detection algorithm on image with faces and the results were as follows.

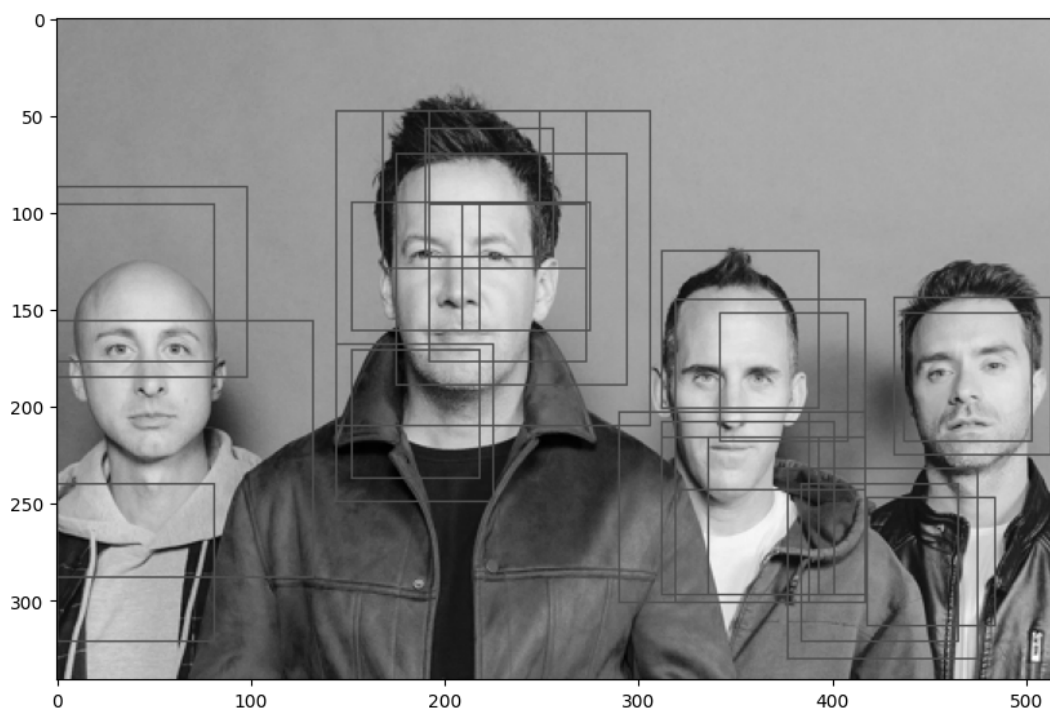


Figure 2: Face detection performance

Though all the four faces are detected, the face detection algorithm is not very accurate. There are too many false positives. This is because I have consider only three features which is not enough to describe the face.

2. Using dataset [link](#), implement a face anti-spoofing model that performs classification based on the below different feature extraction methods to identify fake (spoof) and real image. Compare and analyze your results using metrics accuracy, f1-score, precision, recall and confusion matrix. Document the findings and discuss the failure and success of each method. (*Note: You have to use only 1000 images and not the whole dataset*) (**Marks: 3+3+4**)
- (a) Using the raw pixel values of the face images as features. Train a Support Vector Machine (SVM) classifier on these raw pixel features to perform face recognition. Evaluate and analyze the performance of the model on the dataset.
 - (b) Extract Local Binary Patterns (LBP) features from the face images for feature extraction. Train an SVM classifier using the LBP features to perform face recognition.
 - (c) Compute edge images using any two edge detectors (canny, sobel, prewitt, etc.), then use them as input features independently to train an SVM classifier and perform classification.

Answer:

Loading dataset

I loaded 500 images of real faces and 500 images of spoof faces from the dataset.

Raw pixel values

The raw pixel values of the face images are used as features. I trained a Support Vector Machine (SVM) classifier on these raw pixel features to classify fake and real faces. The performance of the model on the dataset is portrayed in the code.

Local Binary Patterns (LBP) features

Local Binary Patterns (LBP) features are extracted from the face images for feature extraction. I trained an SVM classifier using the LBP features to classify fake and real faces. The performance of the model on the dataset is portrayed in the code.

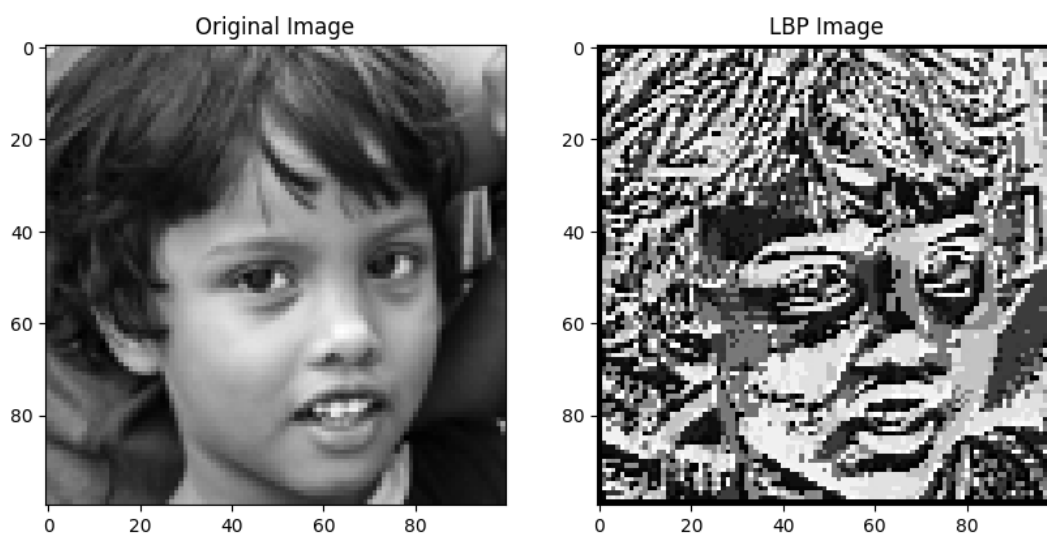


Figure 3: Local binary pattern

Edge methods

Edge images are computed using the Prewitt and Sobel edge detectors. These edge images are used as input features independently to train an SVM classifier and classify fake and real faces. The performance of the model on the dataset is portrayed in the code.

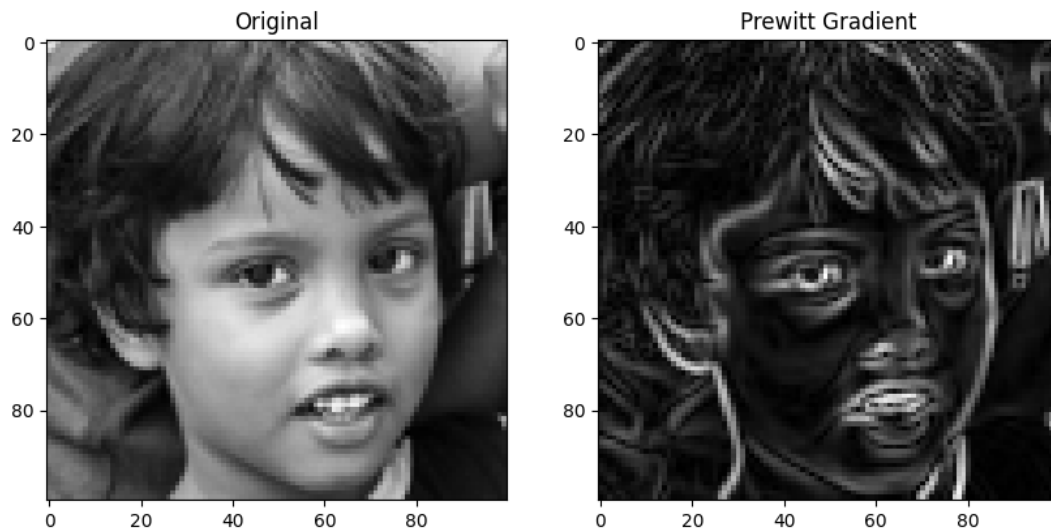


Figure 4: Prewitt filtered image

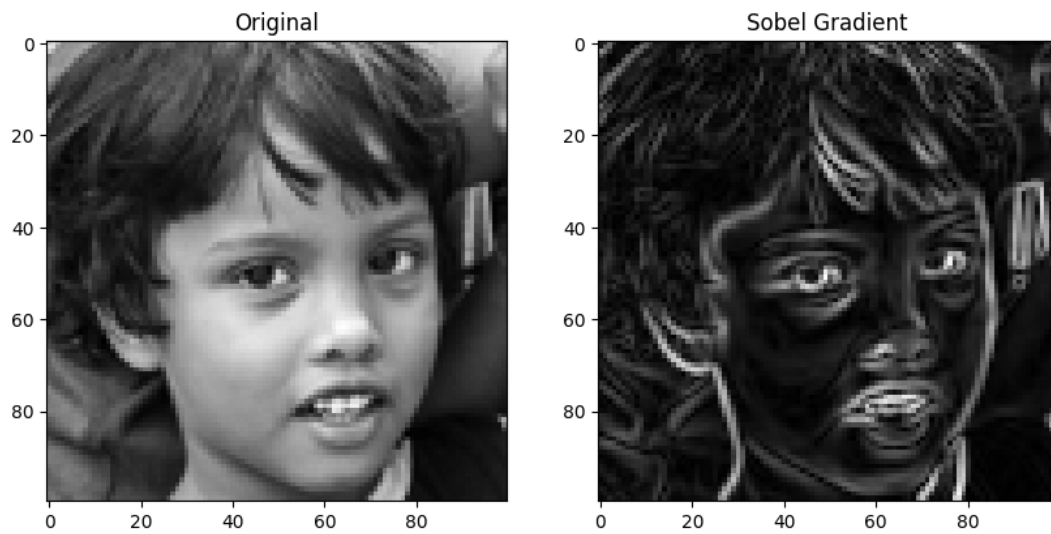


Figure 5: Sobel filtered image