

Monitoring Mobile usage in Classroom

Architha Ramesh
Bangalore Institute of Technology

Abstract -This paper describes work in progress regarding mobile monitoring which helps in the current scenario of usage of smartphones during a lecture or a conference, this would have potential undesirable effects if utilized in restricted premises. Our approach treats face recognition as a tool along with object detection, taking advantage of these two techniques in this paper. This considers the facts that a student yawning, using phone, bowing down head etc. are some of the gestures signifying that a student is not attentive in the class being held. Our paper includes ubiquitous computing using facial and object detection algorithm. This paper helps to analyze student's behavior. Combining the above two approaches help in achieving the stated problem in classroom.

1. INTRODUCTION

Nowadays mobile phone is a very important communicational device. This cable less device reduces the hazards of the land phone. In spite of its various advantages, it has some disadvantages also. Misusing of mobile phone increases the disadvantages of this device day by day. Modern cell phones enable users to access a variety of electronic media at almost any time and any place. At present, the cell phone is likely to be on hand while university students are in class and studying. Thus, the main purpose of the present proposed work is to investigate the usages pattern of a cell phone in the classroom and to reduce the usage which in turn increases attentiveness of the students in the classroom. Mobile phones can be a source of great disruption in workplaces as well as in classrooms. Now almost every mobile phone provides individuals with access to texting, games, social media and the internet. So, they have reduced the attention of students in classes and can, therefore, be injurious to learning. The students are using mobile phones for playing games, sending messages, calling even when the class is in progress..

Biometrics is used in the process of authentication of a person by verifying or identifying that a user requesting a network resource is who he, she, or it claims to be, and vice versa. It uses the property that a human trait associated with a person itself like structure of data with the incoming data we can verify the identity of a particular person.. There are many types of biometric system like detection and recognition, iris recognition etc., these traits are used for human identification in surveillance system, criminal identification, face details etc. By comparing the existing fingerprint recognition. Advancements in computing capability over the past few decades have enabled comparable recognition capabilities from such engineered systems quite successfully. Early face recognition algorithms used

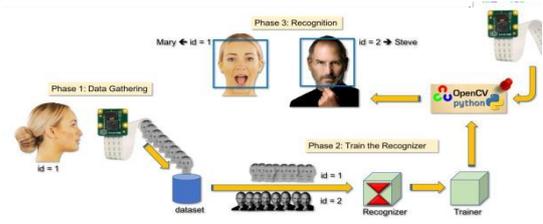
simple geometric models, but recently the recognition process has now matured into a science of sophisticated mathematical representations and matching processes. Major advancements and initiatives have propelled face recognition technology into the spotlight. Face recognition technology can be used in wide range of applications. Computers that detect and recognize faces could be applied to a wide variety of practical applications including criminal

identification etc. Face detection and recognition is used in many places nowadays, verifying websites hosting images and social networking sites. Face recognition and detection can be achieved using technologies related to computer science. Features extracted from a face are processed and compared with similarly processed faces present in the database. If a face is recognized it is known or the system may show a similar face existing in database else it is unknown. In surveillance system if a unknown face appears more than one time then it is stored in database for further recognition. These steps are very useful in criminal identification. In general, face recognition techniques can be divided into two groups based on the face representation they use appearance-based, which uses holistic texture features and is applied to either whole-face or specific face image and feature-based, which uses geometric facial features (mouth, eyebrows, cheeks etc), and geometric relationships between them.

Background and related work

In [1], an efficient algorithm using open source image processing framework called Open CV. Modified version of Haar-Classfier proposed by Viola-Jones used for face detection. LBP histogram for face recognition. In [2], Viola-Jones estimation is done along with processor controller and common USB camera for controlling the electrical motor. In [3], an eigen face is the name given to a set of eigenvectors hence used in the computer vision problem of human face recognition. The approach of using eigenfaces for recognition was developed by Sirovich and Kirby and used by Matthew Turk and Alex Pentland in face classification. In linear algebra, an eigenvector or characteristic vector of a linear transformation is a nonzero vector that changes at most by a scalar factor when that linear transformation is applied. In [4], Image Acquisition is the creation of a digitally encoded representation of the visual characteristics of an object, such as a physical scene or the interior structure of an object. Face detection is a computer technology being used in a variety of applications that identifies human faces in digital images. Face detection also refers to the psychological process by which humans locate and attend to faces in a visual scene. Region detection : This phase detects all the important

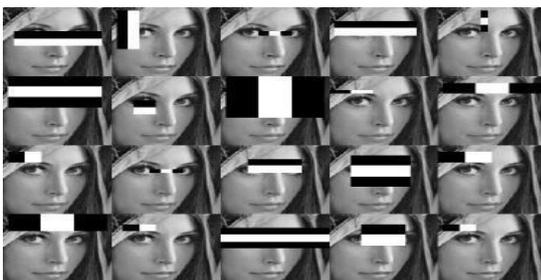
regions in the image from which features can be extracted in the subsequent stages. The Local Binary Pattern Histogram (LBPH) algorithm is a simple solution on face recognition problem, which can recognize both front face and side face. To solve this problem, a modified LBPH algorithm based on pixel neighborhood gray median (MLBPH) is proposed. [6], Image capturing followed by its enhancements, such as Histogram normalization, Median filter. In [7], a two-step mechanism is used. First comes to be face detection for which Viola Jones face detection algorithm is used then followed by face recognition using hybrid algorithm from PCA and LDA. In [8], Image Capture, Face Detection, Pre-processing, Database Development, Feature Extraction and Classification are include



Finding face and facial features

The initial step is detecting the face for which we will use Histogram of Oriented Gradients (HOG) which is a type of “feature descriptor”. The HOG person detector uses a sliding detection window which is moved around the image. At each position of the detector window, a HOG descriptor is computed for the detection window. This descriptor is then shown to the trained SVM, which classifies it as either “person” or “not a person”.

.A Haar-like feature considers adjacent rectangular regions at a specific location in a detection window, sums up the pixel intensities in each region and calculates the difference between these sums. This difference is then used to categorize subsections of an image. For example, let us say we have an image database with human faces. It is a common observation that among all faces the region of the eyes is darker than the region of the cheeks. Therefore a common Haar feature for face detection is a set of two adjacent rectangles that lie above the eye and the cheek region. The position of these rectangles is defined relative to a detection window that acts like a bounding box to the target object (the face in this case).



The rectangle which is drawn on the face gives the centroid in the pixel esteemed. Face location is finished by image processing with python in OpenCV library. The camera will persistently capture video. This video is handled by processor. In the processor the prepared haar course xml record is put away then the processor tries to do he component extraction in video, real highlights in the face are eyes, nose, mouth. If the most element matches, it distinguishes it as a face.

Behavior analysis:

Behavior analysis is a natural science that seeks to understand the behavior of individuals, That is, behavior analysts study how biological, pharmacological, and experiential factors influence the behavior of humans and animals. So in this project we are doing behavioral analysis by body detection and motion recognition techniques. This is possible by Haar cascade.

The proposed solution consists of two steps:

Face Recognition:

Face detection is done using Principal Component Analysis. Principal component analysis (PCA) is a statistical procedure that uses an orthogonal transformation to convert a set of observations of possibly correlated variables into a set of values of linearly uncorrelated variables called principal components. This transformation is defined in such a way that the first principal component has the largest possible variance, and each succeeding component in turn has the highest variance possible under the constraint that it is orthogonal to the preceding components. The resulting vectors are an uncorrelated orthogonal basis set. PCA is sensitive to the relative scaling of the original variables.

Behavioral analysis

Fisher faces approach uses Linear Discriminate Analysis (LDA) method to reduce the variance between image space classes, and tightly group those similar groups together to maximize the separation between different projected classes. Firstly we locate faces in an image, then we extract facial features from the detected face regions and analyzing the motion of facial features and classifying this information in some facial interpretative expressions such as facial muscle

activations like smile, frown, emotion categories. In our project we will interpret it as using phone by considering various gestures. Facial landmark

detection algorithms aim to automatically identify the locations of the facial key landmark points on facial images or videos. Those key points are either the dominant points describing the unique location of a facial component (e.g., eye corner) or an interpolated point connecting those dominant points around the facial components and facial contour. Formally, given a facial image denoted as I , a landmark detection algorithm

predicts the locationsof D landmarks $x = \{x_1, y_1, x_2, y_2, \dots, x_D, y_D\}$ $x = \{x_1, y_1, x_2, y_2, \dots, x_D, y_D\}$, where $x.x.$ and $y.y.$ resentmentthe image coordinates of the facial landmarks. Facial landmark detection is challenging for several reasons. First, facial appearance changes significantly across subjects under different facial

expressions and head poses. Second, the environmental conditions such as the illumination would affect the appearance of the faces on the facial images. Third, facial occlusion by other objects or self-occlusion due to extreme head poses would lead to incomplete facial appearance information. Over the past few decades, there have been significant developments of the facial landmark detection algorithms. The early works focus on the less challenging facial images without the aforementioned facial variations. Later, the facial landmark detection algorithms aim to handle several variations within certain categories, and the facial images are usually collected with “controlled” conditions. For example, in “controlled” conditions, the facial poses and facial expressions can only be in certain categories. More recently, the research focuses on the challenging “in-the-wild” conditions, in which facial images can undergo arbitrary facial expressions, head poses, illumination, facial occlusions, etc. In general, there is still a lack of a robust method that can handle all those variations.

Facial landmark detection algorithms can be classified into three major categories: the *holistic methods*, the *Constrained Local Model (CLM) methods*, and *regression-based methods*, depending on how they model the facial appearance and facial shape patterns. The facial appearance refers to the distinctive pixel intensity patterns around the facial landmarks or in the whole face region, while face shape patterns refer to the patterns of the face shapes as defined by the landmark locations and their spatial relationships. As summarized in Table 1, the holistic methods explicitly model the holistic facial appearance and global facial shape patterns. CLMs rely on the explicit local facial appearance and explicit global facial shape patterns. These regression-based methods use holistic or local appearance information and they may embed the global facial shape patterns implicitly for joint landmark detection. In general, the regression-based methods show better performances recently (details will be discussed later). Note that, some recent methods combine the deep learning models and global 3D shape models for detection and they are outside the scope of the three major categories.



model_filename f : (REQUIRED) A path to binary file storing the trained model which is to be loaded [example - /data/file.dat]

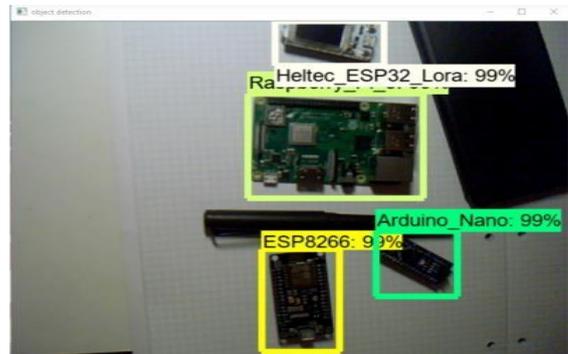
- **image i** : (REQUIRED) A path to image in which face landmarks have to be detected.[example -

/data/image.jpg]

- **face_cascade c** : (REQUIRED) A path to the face cascade xml file which you want to use as a face detector.

Object Detection is the process of finding real-world object instances like car, bike, TV, flowers, and humans in still images or Videos. It allows for the recognition, localization, and detection of multiple objects within an image which provides us with a much better understanding of an image as a whole. It is commonly used in applications such as image retrieval, security, surveillance, and advanced driver assistance systems (ADAS).

Tensorflow is Google’s Open Source Machine Learning Framework for dataflow programming across a range of tasks. Nodes in the graph represent mathematical operations, while the graph edges represent the multi-dimensional data arrays (**tensors**) communicated between them.



Cell Phone Detection:

Deep Learning has emerged as a new area in machine learning and is applied to a number of signal and image applications. The main purpose of the work presented in this paper, is to apply the concept of a Deep Learning algorithm namely, Convolutional neural networks (CNN) in image classification. The algorithm is tested on various standard datasets, like remote sensing data of aerial images (UC Merced Land Use Dataset) and scene images from SUN database. The performance of the algorithm is evaluated based on the quality metric known as Mean Squared Error (MSE) and classification accuracy. The graphical representation of the experimental results is given on the basis of MSE against the number of training epochs. The experimental result analysis based on the quality metrics and the graphical representation proves that the algorithm (CNN) gives fairly good classification accuracy for all the tested datasets. Working of CNN algorithm This section explains the working of the algorithm in a brief manner. The detailed explanation is available in [7]. The input to the network is a 2D image. The network has input layer which takes the image as the input, output layer from where we get the trained output and the intermediate layers called as the hidden layers. As stated earlier, the network has a series of convolutional and sub-sampling layers. Together the layers produce an approximation of input image data’s. CNNs exploit spatially local correlation by enforcing a local connectivity pattern between neurons of

adjacent layers [8]. Neurons in layer say, 'm' are connected to a local subset of neurons from the previous layer of (m-1), where the neurons of the (m-1) layer have contiguous receptive fields, as shown in figure (2a).

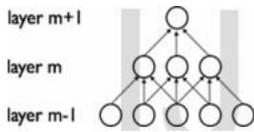


Figure 2(a): Graphical flow of layers showing connection between layers [4]

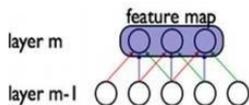
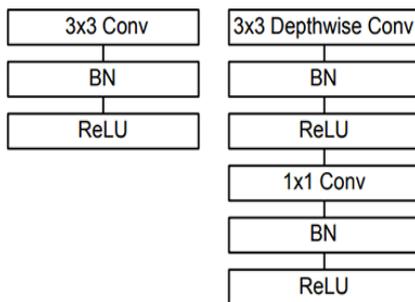


Figure 2(b): Graphical flow of layers showing sharing of weights [4]

In the CNN algorithm, each sparse filter is replicated across the entire visual field. These units then form a feature maps, these share weight vector and bias. Figure (2b), represents three hidden units of same feature map. The weights of same color are shared, thus constrained to be identical.

Methodology For Cell Phone Detection



When building object detection networks we normally use an existing network architecture, such as VGG or ResNet, and then use it inside the object detection pipeline. The problem is that these network architectures can be very large in the order of 200-500MB. Network architectures such as these are unsuitable for resource constrained devices due to their sheer size and resulting number of computations. We call these networks "MobileNets" because they are designed for resource constrained devices such as your smartphone. MobileNets differ from traditional CNNs through the usage of *depthwise separable convolution* (Figure above).

The general idea behind depthwise separable convolution is to split convolution into two stages:

1. A 3x3 depthwise convolution.
2. Followed by a 1x1 pointwise convolution.

OPEN-CV MODULE

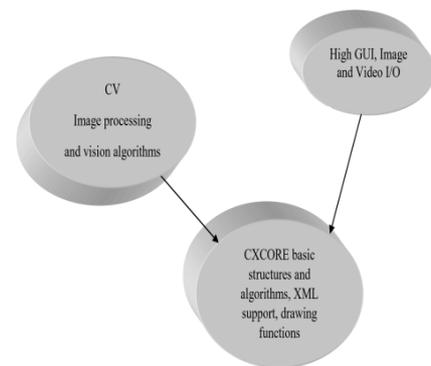
OpenCV (Open Source Computer Vision) is a library of programming functions mainly aimed at real-time computer vision. Originally developed by Intel, it was

later supported by Willow Garage then Itseez (which was later acquired by Intel). The library is cross platform and free for use under the open-source BSD license. OpenCV supports deep learning frameworks TensorFlow, Torch/Py Torch and Caffe.

It has C++, Python, Java and MATLAB interfaces and supports Windows, Linux, Android and Mac OS. OpenCV leans mostly towards real-time vision applications and takes advantage of MMX and SSE instructions when available. A full-featured CUDA and OpenCL interfaces are being actively developed right now. There are over 500 algorithms and about 10 times as many functions that compose or support those algorithms. OpenCV is written natively in C++ and has a templated interface that works seamlessly with STL containers.

In 1999, the OpenCV project was initially an Intel Research initiative to advance CPU intensive applications, part of a series of projects including real-time ray tracing and 3D display walls. The main contributors to the project included a number of optimization experts in Intel Russia, as well as Intel's Performance Library Team. In the early days of OpenCV, the goals of the project were described as:

- Advance vision research by providing not only open but also optimized code for basic vision infrastructure. No more reinventing the wheel.
- Disseminate vision knowledge by providing a common infrastructure that developers could build on, so that code would be more readily readable and transferable.
- Advance vision-based commercial applications by making portable, performance optimized code available for free – with a license that did not require code to be open or free itself.

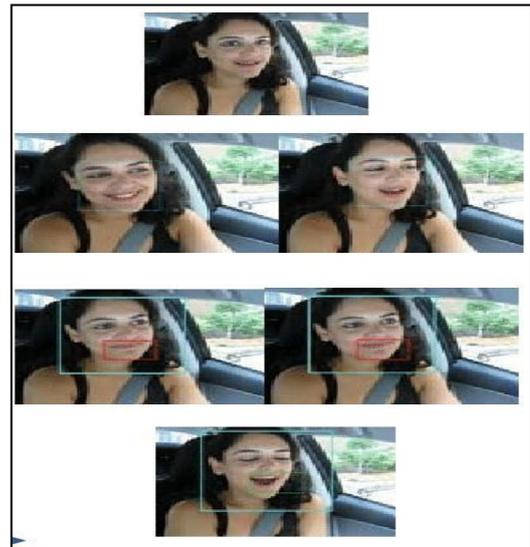
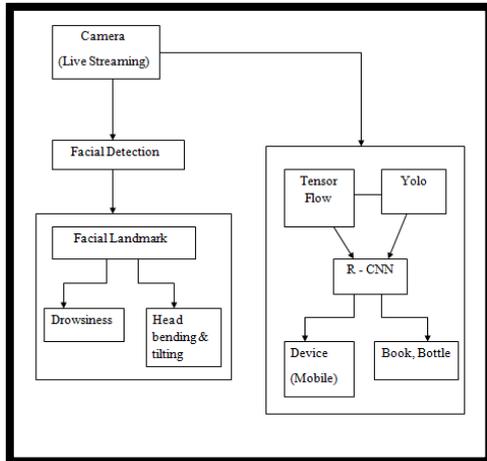


Once OpenCV is installed, the OPENCV_BUILD\install directory will be populated with three types of files:

- **Header files:** These are located in the OPENCV_BUILD\install\include subdirectory and are used to develop new projects with OpenCV.
- **Library binaries:** These are static or dynamic libraries (depending on the option selected with CMake) with the functionality of each of the OpenCV modules. They are located in the bin

subdirectory (for example, x64\mingw\bin when the GNU compiler is used).

- **Sample binaries:** These are executables with examples that use the libraries. The sources for these samples can be found in the source package.



1 Recognition Experiments Acceptance Testing

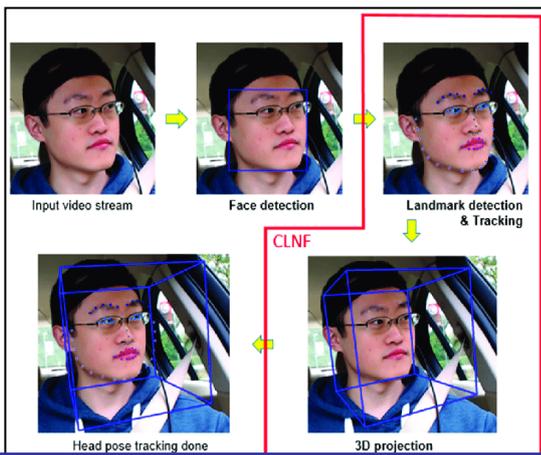
Acceptance testing is actually a series of different tests whose primary purpose is to fully exercise the computer-based system. Include recovery testing crashes, security testing for unauthorized user, etc. Acceptance testing is sometimes performed with realistic data of the client to demonstrate that the software is working satisfactorily. This testing in FDAC focuses on the external behavior of the system.

Validation Testing

At the culmination of integration testing, software is completely assembled as a packages; interfacing errors have been covered and corrected, and final series of software tests-validating testing may begin. Validation can be defined in many ways, but a simple definition is that validation succeeds when software functions in a manner that can be reasonably expected by customers. Reasonable expectation is defined in the software requirement specification- a document that describes all users' visible attributes of the software. The specification contains a section title "validation criteria". Information contained in that section forms the basis for validation testing approach.



2. RESULTS



3. FURTHER ISSUES AND CONCLUSION

The field of computer vision is widely utilized in several disciplines of science worldwide, and day after day its applications that touch our daily lives are growing. The students' activities in the examination rooms are one of the most important fields that affect many dimensions. Most conventional approaches rely upon the utilization of human beings as the main power for monitoring the students' behaviors. In this study a monitoring system is proposed that is capable of continuously monitoring the behavior of the students using a fixed camera. The proposed monitoring system consists of three layers which are, face detection, suspicious state detection (using neural network) and anomaly detection (using Gaussian-based method). The results achieved prove the validity of our proposed prototype to monitor students successfully by detecting the students in the examination room, and segmenting them successfully from the input camera feed. As well as, the ability to detect and track the faces of each segmented

image and classify them as being in suspicious or non-suspicious states using a one layer neural net. Finally, a simplification of detecting anomalous behavior is done by measuring the rate of anomalous states in a fixed window of a sequence of n-frames based on the Gaussian distribution method. This opens the door for further investigation along the direction of the presented/discussed two-layer monitoring system, as it is valid for accurately handling the investigated problem. Future endeavors are to consider depending on the hand gestures and other clues that student provide while cheating, applying the optical flow method to detect any fast movements could also be a good idea to try to implement.

4. REFERENCES

- [1] H. Kopetz, "Internet of things," in Real-time systems. Springer, 2011, pp. 307–323.
- [2] R. H. Weber and R. Weber, Internet of things. Springer, 2010, vol. 12.
- [3] R. Brunelli and T. Poggio, "Face recognition: Features versus templates," IEEE transactions on pattern analysis and machine intelligence, vol. 15, no. 10, pp. 1042–1052, 1993.
- [4] K.-C. Lee, J. Ho, and D. J. Kriegman, "Acquiring linear subspaces for face recognition under variable lighting," IEEE Transactions on pattern analysis and machine intelligence, vol. 27, no. 5, pp. 684–698, 2005.
- [5] M. A. Turk and A. P. Pentland, "Face recognition using eigenfaces," in Computer Vision and Pattern Recognition 1991. Proceedings CVPR'91., IEEE Computer Society Conference on. IEEE, 1991, pp. 586–591.
P. N. Belhumeur, J. P. Hespanha, and D. J.