The Design Of Auto-Motive Burglar-Proof Based On Opencv And Arm9

Mohammed Shahbaz Uddin
M.Tech Student,
VIF College of Engineering and Technology

B. Kotesh, M.Tech
Asst professor,
VIF College of Engineering and Technology

Imthiazunnisa Begum, M.Tech
HOD, ECE,
VIF College of Engineering and Technology

Korani Ravinder, M.Tech(Ph.D)
Assistant professor,
VIF College of Engineering and Technology

Abstract:

In recently, the end of economic growing has been come since the series of passive reasons in economic area the vehicle industry was influenced by the economic recession, the structure of vehicle industry technology are changing largely. However, the number of vehicle theft crime has been rising when more and more vehicles are brought into our daily life. The vehicle owners' concern about the issues that how to prevent the vehicle theft crimes.

However, there are many bugs on the present vehicle burglarproof. In the traditional vehicle burglarproof areas, the common measures could be classified by several categories that they either rely on simple wireless control device to achieve the targets, or use the GSM network technology to achieve. Nevertheless, the functions of these devices are too simple to prevent the vehicle theft crimes from happening, furthermore, their burglarproof methods are not only character.

Nowadays, with the development of science, the biological recognition technology which has much unique advantage, such as uniqueness, changeless, and so on, has been extraordinary improving. One of biological recognitions is human-face recognition. The methods and algorithms of human-face characters' extraction are developed.

They provide a new idea to the vehicle burglarproof area. This system takes advantage of only character of human face feature recognition algorithm in which efficiency of recognition rate is considerable. We proposed that the whole system is built on the platform of embedded system which took advantage of the algorithm of human face recognition.

The ARM9-based system combined with the traditional merits of vehicle burglarproof. At the same time, GSM network could be also used in the system.

When the crime happened or was about to doing, the messages of alarm would be sent to the vehicle owner as soon as possible. The performance of vehicle burglarproof system has been improved by the system, and the traditional vehicle burglarproof disadvantage could also be solved.

In this project, The ARM9 chip is used as the core of this embedded system which is combined with the technologies of human-face recognition and GSM wireless communication. The new vehicle burglarproof system contained the function of human-face recognition and take advantage of the advantage of present ones.

INTRODUCTION:

The Automobile industry in India is one of the largest in the world and one of the fastest growing globally. India's passenger car and commercial vehicle manufacturing industry is the seventh largest in the world, with an annual production of more than 3.7 million units in 2010. According to recent reports, India is set to overtake Brazil to become the sixth largest passenger vehicle producer in the world, growing 16-18 per cent to sell around three million units in the course of 2011-12. In 2009, India emerged as Asia's fourth largest exporter of passenger cars, behind Japan, South Korea, and Thailand

It's almost impossible to steal a modern car without first obtaining the keys – Thatcham, the Motor Insurance Repair Research Centre recently reported that 70% of cars stolen using the key. There are many different ways of obtaining car keys but Thatcham figures suggest that at least one in five cases involve domestic burglary – snatch the keys from the house and take the car from the drive. Over 19,400 cars were stolen in this way – more than 1,600 per month – last year. So here, we are proposing a system which can add its own importance to the existing security present in the automobile industry.

The main functions are shown as follows:

(1)Human-face recognition: The owners' face information is used as the standards of recognition. It must verify the feature of the human face before using vehicle.

(2)Message alarming: When someone tries to thieve the vehicle, the message can be send to the owners' mobile phone as soon as possible without any noise.

(3)GSM network: The system can call the police or send the message via the GSM network wherever the vehicle was.

(4)Two methods of recognition: Besides the human face recognition, the method of password recognition can be also used to the system.

Architecture:

The ARM processor is a Reduced Instruction Set Computer (RISC). The ARM was originally developed at Acorn Computers Limited of Cambridge, England, between 1983 and 1985.

ARM9 ORGANIZATION AND IMPLEMENTATION:

Since 1995 several new ARM cores have been introduced which deliver significantly higher performance through the use of 5-stage pipelines and separate instruction and data memories (usually in the form of separate caches which are connected to a shared instruction and data main memory system). ARM920T implements 5-stage pipeline architecture.

5stage pipeline ARM organization:

All processors have to develop to meet the demand for higher performance. The 3-stage pipeline used in the ARM cores up to the ARM is very cost-effective, but higher performance requires the processor organization to be rethought.

With a given compiler using a given set of optimizations, and so on) there are only two ways to increase performance:

- Increase the clock rate, Fclk. This requires the logic in each pipeline stage to be simplified and, therefore, the number of pipeline stages to be increased.

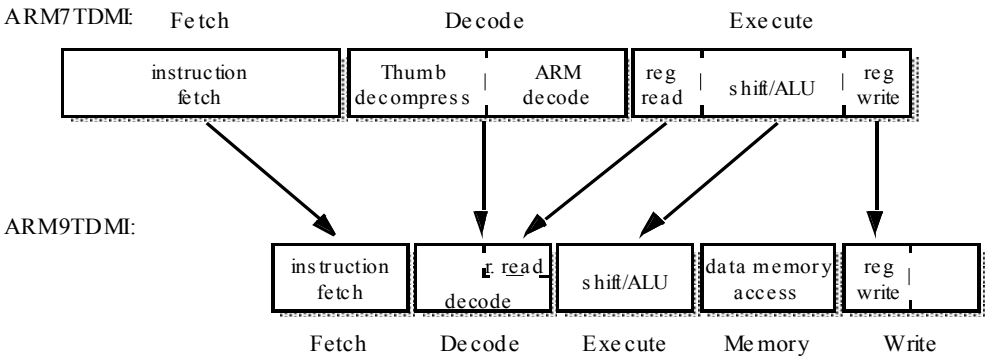
- Reduce the average number of clock cycles per instruction, CPI. This requires either that instructions which occupy more than one pipeline slot in a 3-stage pipeline ARM are re-implemented to occupy fewer slots, or that pipeline stalls caused by dependencies between instructions are reduced, or a combination of both.

The fundamental problem with reducing the CPI relative to a 3-stage core is related to the von Neumann bottleneck - and stored-program computer with a single instruction and data memory will have its performance limited by the available memory bandwidth. A 3-stage ARM core accesses memory on (almost) every clock cycle either to fetch an instruction or to transfer data.

Simply tightening up on the few cycles where the memory is not used will yield only a small performance gain. To get a significantly better CPI the memory system must deliver more than one value in each clock cycle either by delivering more than 32 bits per cycle from a single memory or by having separate memories for instruction and data accesses.

and data memories (which may be separate caches connected to a unified instruction and data main memory) allow a significant reduction in the core's CPI.

A typical 5-stage ARM pipeline is that employed in the ARM9TDMI. The organization of the ARM9TDMI is illustrated in Figure. The ARM processors which use a 5-stage pipeline have the following pipeline stages:



BLOCK DIAGRAM

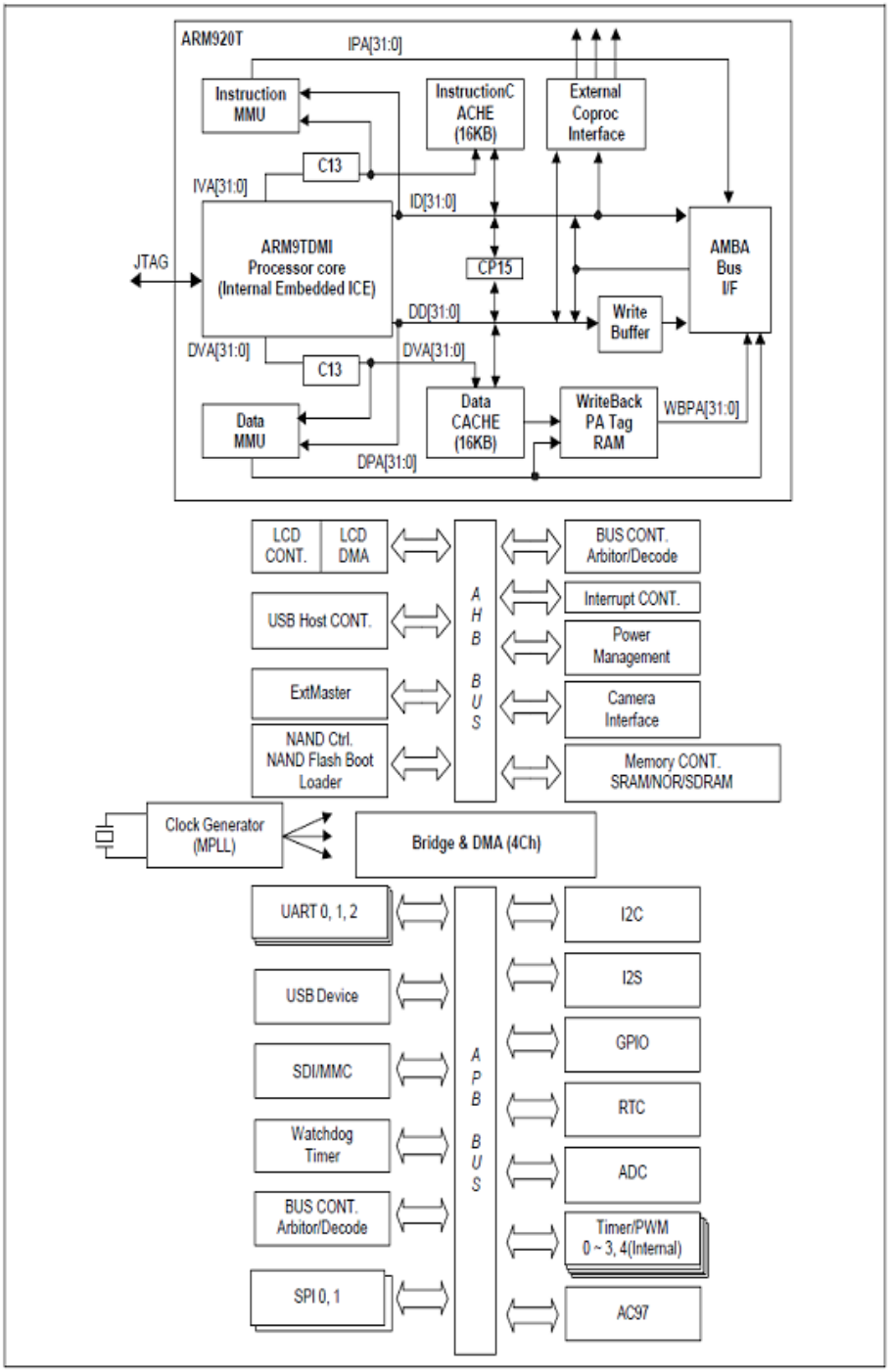


Figure 10: S3C2440A Block Diagram

Design and implementation:

Face Detection Overview:

Face detection is a computer technology that determines the locations and sizes of human faces in arbitrary (digital) images. It detects facial features and ignores anything else, such as buildings, trees and bodies.

Techniques:

Many algorithms implement the face-detection task as a binary pattern-classification task.

That is, the content of a given part of an image is transformed into features, after which a classifier trained on example faces decides whether that particular region of the image is a face, or not.

Often, a window-sliding technique is employed. That is, the classifier is used to classify the (usually square or rectangular) portions of an image, at all locations and scales, as either faces or non-faces (background pattern). Images with a plain or a static background are easy to process. Remove the background and only the faces will be left, assuming the image only contains a frontal face.

Using skin color to find face segments is a vulnerable technique. The database may not contain all the skin colors possible. Lighting can also affect the results. Non-animate objects with the same color as skin can be picked up since the technique uses color segmentation. The advantages are the lack of restriction to orientation or size of faces and a good algorithm can handle complex backgrounds.

. Once the face is detected, the model is laid over the face and the system is able to track face movements.

Haar Cascades:

A recognition process can be much more efficient if it is based on the detection of features that encode some information about the class to be detected. This is the case of Haar-like features that encode the existence of oriented contrasts between regions in the image. A set of these features can be used to encode the contrasts exhibited by a human face and their spacial relationships. Haar-like features are so called because they are computed similar to the coefficients in Haar wavelet transforms.

The object detector of OpenCV has been initially proposed by Paul Viola and improved by Rainer Lienhart. First, a classifier (namely a cascade of boosted classifiers working with haar-like features) is trained with a few hundreds of sample views of a particular object (i.e., a face or a car), called positive examples, that are scaled to the same size (say, 20x20), and negative examples - arbitrary images of the same size.

After a classifier is trained, it can be applied to a region of interest (of the same size as used during the training) in an input image. The classifier outputs a "1" if the region is likely to show the object (i.e., face/car), and "0" otherwise. To search for the object in the whole image one can move the search window across the image and check every location using the classifier. The classifier is designed so that it can be easily "resized" in order to be able to find the objects of interest at different sizes, which is more efficient than resizing the image itself. So, to find an object of an unknown size in the image the scan procedure should be done several times at different scales.

How Face Detection Works:

OpenCV's face detector uses a method that Paul Viola and Michael Jones published in 2001. Usually called simply the Viola-Jones method, or even just Viola-Jones, this approach to detecting objects in images combines four key concepts:

- Simple rectangular features, called Haar features.
- An Integral Image for rapid feature detection.
- The AdaBoost machine-learning method.
- A cascaded classifier to combine many features efficiently.

The features that Viola and Jones used are based on Haar wavelets. Haar wavelets are single wavelength square waves (one high interval and one low interval). In two dimensions, a square wave is a pair of adjacent rectangles - one light and one dark.

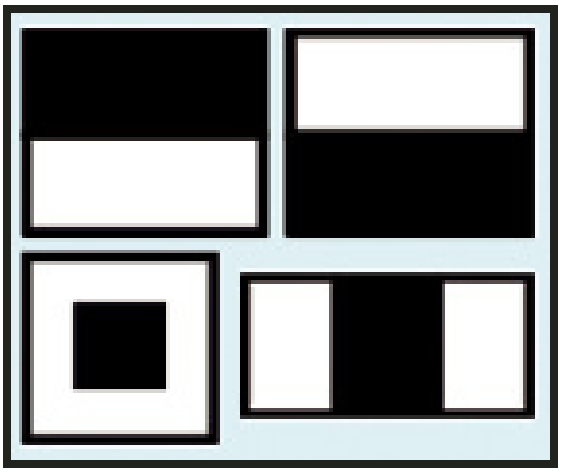


Fig 28: Examples of the Haar features used in OpenCV.

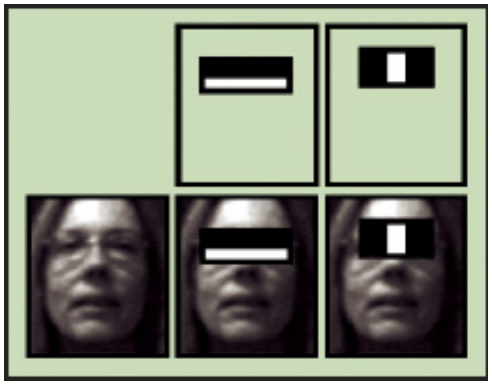


Fig 29: The first two Haar features in the original Viola-Jones cascade.

The actual rectangle combinations used for visual object detection are not true Haar wavlets. Instead, they contain rectangle combinations better suited to visual recognition tasks.

Because of that difference, these features are called Haar features, or Haarlike features, rather than Haar wavelets. Figure 28 shows the features that OpenCV uses.

The presence of a Haar feature is determined by subtracting the average dark-region pixel value from the average light-region pixel value. If the difference is above a threshold (set during learning), that feature is said to be present.

To determine the presence or absence of hundreds of Haar features at every image location and at several scales efficiently, Viola and Jones used a technique called an Integral Image. In general, "integrating" means adding small units together. In this case, the small units are pixel values. The integral value for each pixel is the sum of all the pixels above it and to its left. Starting at the top left and traversing to the right and down, the entire image can be integrated with a few integer operations per pixel.

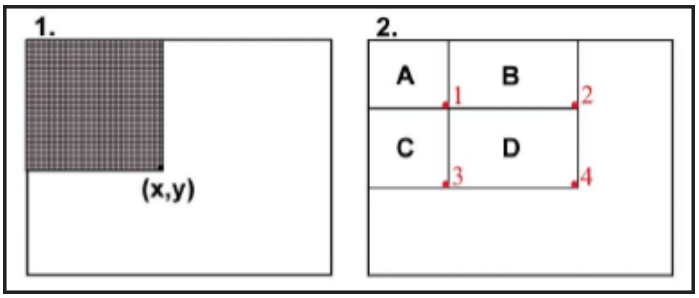


Fig 30: Cascade movement on a Image.

As Figure 30 shows, after integration, the value at each pixel location, (x,y), contains the sum of all pixel values within a rectangular region that has one corner at the top left of the image and the other at location (x,y). To find the average pixel value in this rectangle, you'd only need to divide the value at (x,y) by the rectangle's area.

But what if you want to know the summed values for some other rectangle, one that doesn't have one corner at the upper left of the image? Figure 31 shows the solution to that problem. Suppose you want the summed values in D. You can think of that as being the sum of pixel values in the combined rectangle, A+B+C+D, minus the sums in rectangles A+B and A+C, plus the sum of pixel values in A. In other words,

D = A+B+C+D - (A+B) - (A+C) + A.

Conveniently, A+B+C+D is the Integral Image's value at location 4, A+B is the value at location 2, A+C is the value at location 3, and A is the value at location 1. So, with an Integral Image, you can find the sum of pixel values for any rectangle in the original image with just three integer operations:

(x4, y4) - (x2, y2) - (x3, y3) + (x1, y1). a

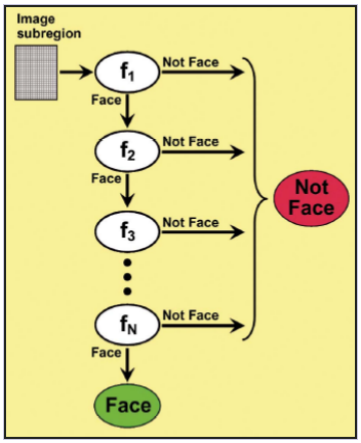


FIG 31: Flowchart of cascade movement on the image.

The classifier cascade is a chain of filters. Image subregions that make it through the entire cascade are

Face recognition Overview:

The task of facial recognition is discriminating input signals (image data) into several classes (persons). The input signals are highly noisy (e.g. the noise is caused by differing lighting conditions, pose etc.), yet the input images are not completely random and in spite of their differences there are patterns which occur in any input signal.

Such patterns, which can be observed in all signals, could be – in the domain of facial recognition – the presence of some objects (eyes, nose, mouth) in any face as well as relative distances between these objects. These characteristic features are called Eigen faces in the facial recognition domain (or principal components generally). They can be ex-tracked out of original image data by means of a mathematical tool called Principal Component Analysis (PCA).

By means of PCA one can transform each original image of the training set into a corresponding Eigen face. An important feature of PCA is that one can reconstruct recon-strut any original image from the training set by combining the Eigen faces. Remember that Eigen faces are nothing less than characteristic features of the faces. Therefore one could say that the original face image can be reconstructed from Eigen faces if one adds up all the Eigen faces (features) in the right proportion. Each Eigen face represents only certain features of the face, which may or may not be present in the original image.

If the feature is present in the original image to a higher degree, the share of the corresponding Eigen face in the "sum" of the Eigen faces should be greater.

If, contrary, the particular feature is not (or almost not) present in the original image, then the corresponding Eigen face should contribute a smaller (or not at all) part to the sum of Eigen faces. So, in order to reconstruct the original image from the Eigen faces, one has to build a kind of weighted sum of all Eigen faces. That is, the reconstructed original image is equal to a sum of all Eigen faces, with each Eigen face having a certain weight. This weight specifies, to what degree the specific feature (Eigen face) is present in the original image.

If one uses all the Eigen faces extracted from original images, one can reconstruct the original images from the Eigen faces exactly. But one can also use only a part of the Eigen faces. Then the reconstructed image is an approximation of the original image. However, one can ensure that losses due to omitting some of the Eigen faces can be minimized. This happens by choosing only the most important features (Eigen faces).

Omission of Eigen faces is necessary due to scarcity of computational resources. How does this relate to facial recognition? The clue is that it is possible not only to extract the face from Eigen faces given a set of weights, but also to go the opposite way. This opposite way would be to extract the weights from Eigen faces and the face to be recognized. These weights tell nothing less, as the amount by which the face in question differs from"typical" faces represented by the Eigen faces.

Conclusion:

The recognition rate of the complete system will be around 75% under testing conditions.So this system is applicable to the middle level security/authentication systems.

Future Scope:

We can increase the recognition rate by changing the no of pixels in an individual image but we require high end controller to process in which FARM fails.

We can use image edging and image smoothing processes before going for detection which will increase the face detection correctness and as a result recognition rate will also increase.

References:

1.http://opencv.willowgarage.com/wiki/
2.http://sourceforge.net/projects/opencv/
3.http://www.cmake.org/
4.http://www.vtk.org/Wiki/CMake/
5.http://www.friendlyarm.net
6.http://www.unistring.com
7.www.embeddedworld.com

results

1.1.Face Detection

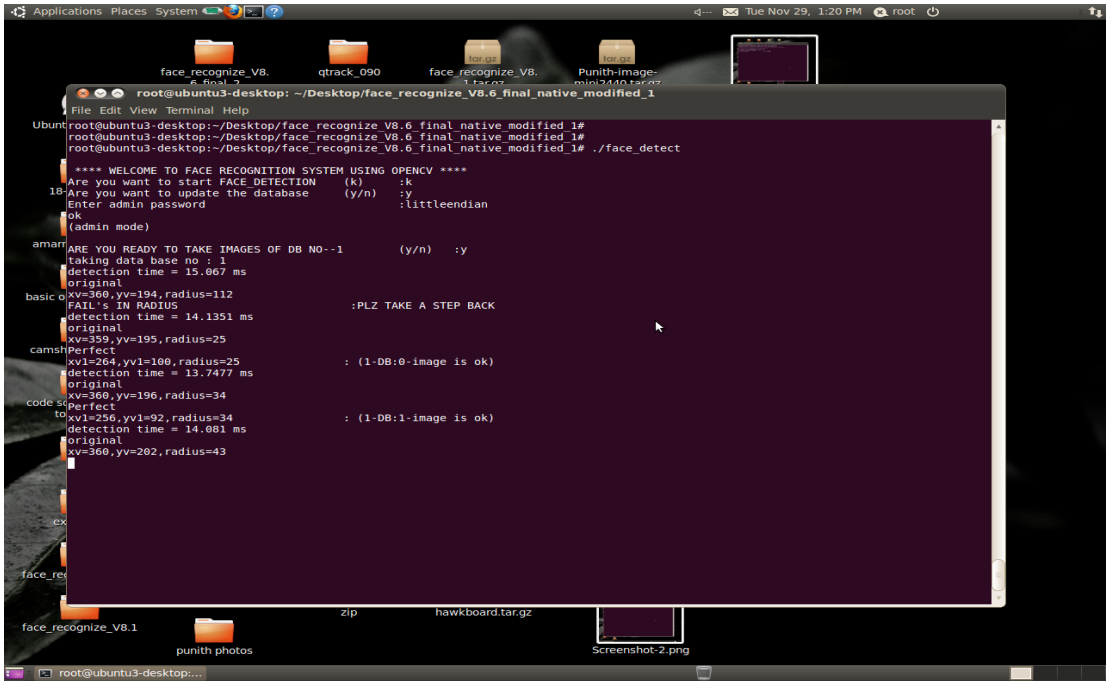


FIG 37: Output window for data base updating

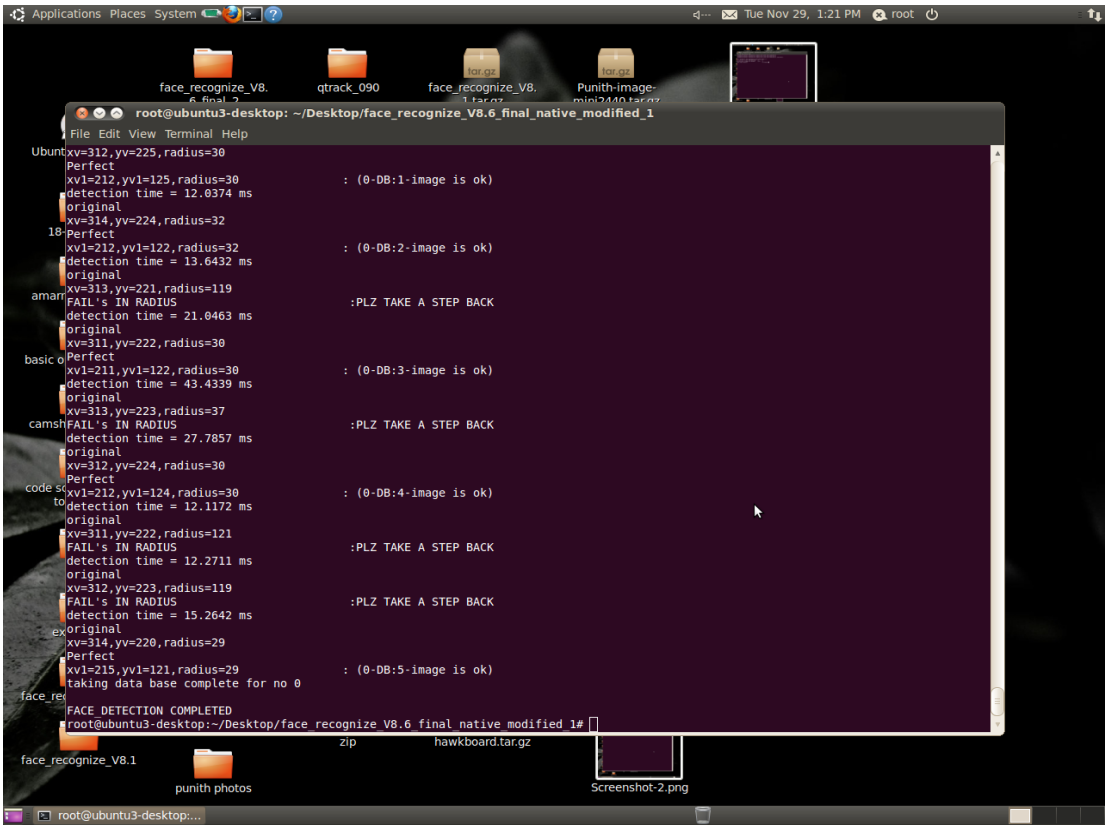


FIG 38: Output window of FACE DETECTION.

1.2 Face Recognition

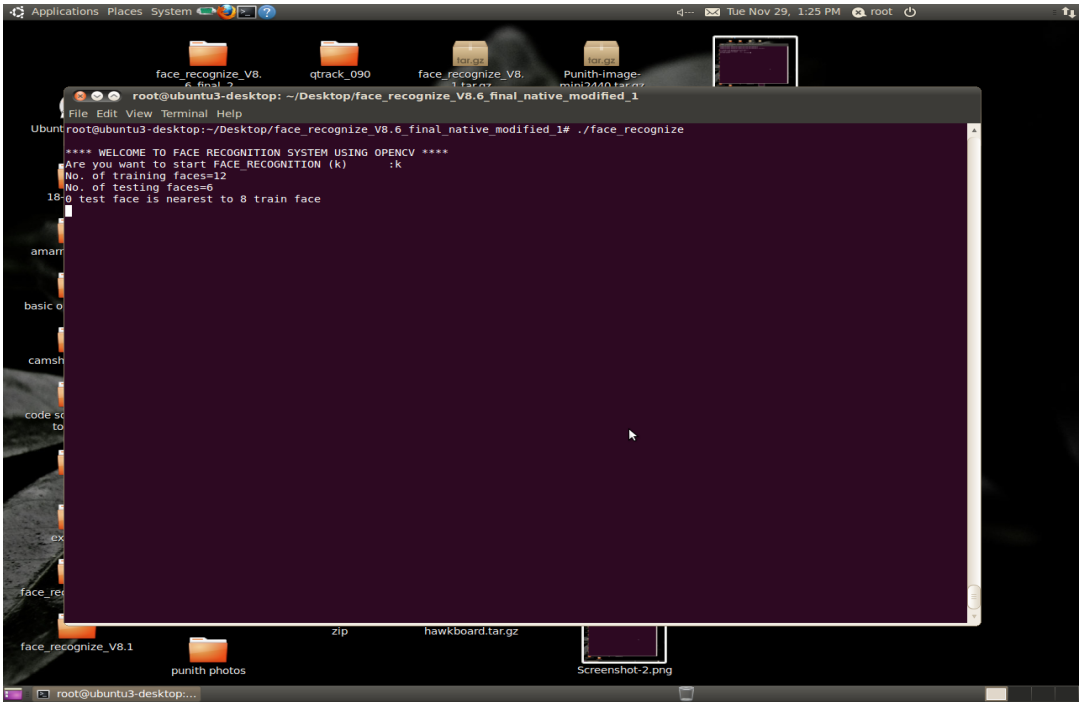


FIG 39: Output window of FACE_RECOGNITION

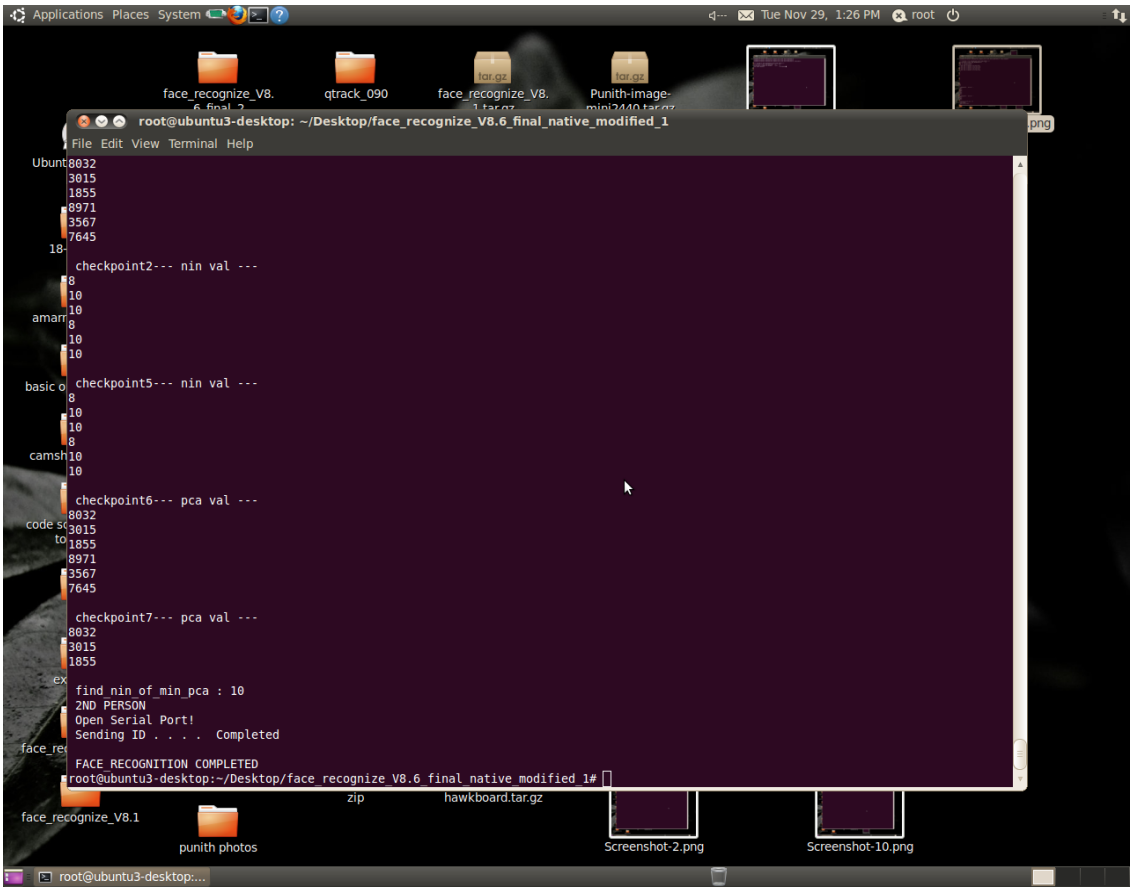


FIG 40: Output window of ANALYSIS PHASE

1.3 Hardware images

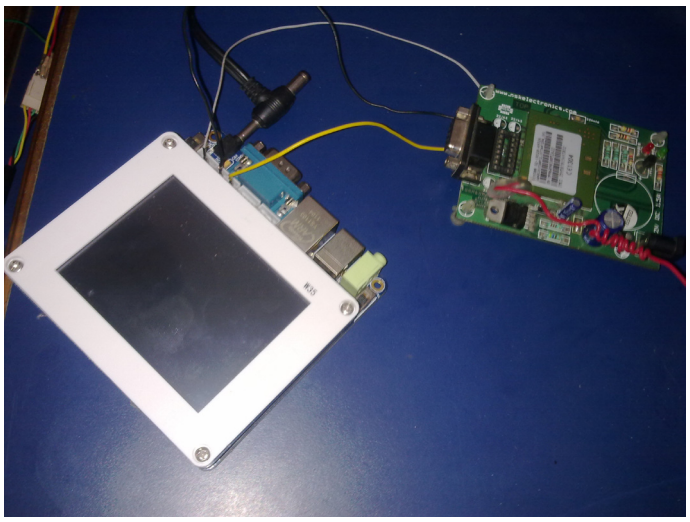


FIG 41: HARDWARE SETUP

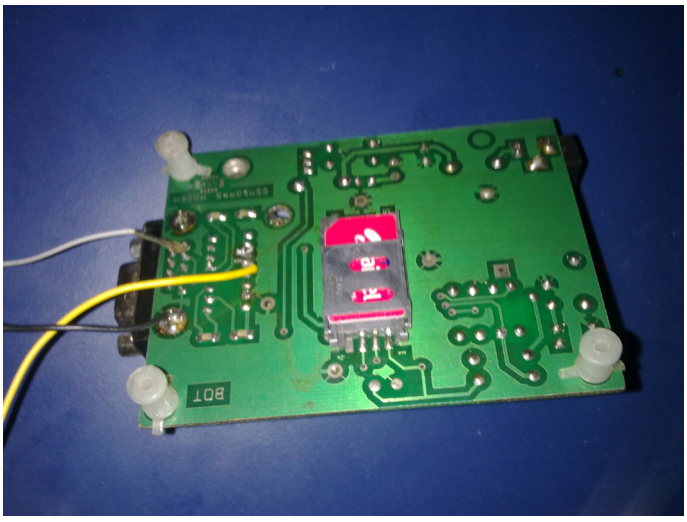


FIG 42: GSM MODEM