FACE DETECTION SYSTEM WITH FACE RECOGNITION

Mohd Ismail [1], Mohammed Mutaseem Hussain [2], Mohd Salman Qureshi [3] Dr. Veeramalaya [4]

[1]BE Student, [2] BE Student,[3] BE Student [4] Professor

CSE Dept, ISL ENGINEERING COLLEGE, Hyd.

ABSTRACT: The face is one of the easiest ways to distinguish the individual identity of each other. Face recognition is a personal identification system that uses personal characteristics of a person to identify the person's identity. Human face recognition procedure basically consists of two phases, namely face detection, where this process takes place very rapidly in humans, except under conditions where the object is located at a short distance away, the next is the introduction, which recognize a face as individuals. Stage is then replicated and developed as a model for facial image recognition (face recognition) is one of the much-studied biometrics technology and developed by experts. There are two kinds of methods that are currently popular in developed face recognition pattern namely, Eigenface method and Fisherface method. Facial image recognition Eigenface method is based on the reduction of face dimensional space using Principal Component Analysis (PCA) for facial features. The main purpose of the use of PCA on face recognition using Eigen faces was formed (face space) by finding the eigenvector corresponding to the largest eigenvalue of the face image. The area of this project face detection system with face recognition is Image processing. The software requirements for this project is numpy software.

Keywords: face detection, Eigen face, Cv2,Numpy

Extension: There are vast number of applications from this face detection project, this project can be extended that the various parts in the face can be detect which are in various directions and shapes.

1. INTRODUCTION

The face is one of the easiest ways to distinguish the individual identity of each other. Face recognition is a personal identification system that uses personal characteristics of a person to identify the person's identity. Human face recognition procedure basically consists of two phases, namely face detection, where this process takes place very rapidly in humans, except under conditions where the object is located at a short distance away, the next is the introduction, which recognize a face as individuals. Stage is then replicated and developed as a model for facial image recognition (face recognition) is one of the much-studied biometrics technology and developed by experts. There are two kinds of methods that are currently popular in developed face recognition pattern namely, Eigenface method and Fisherface method. Facial image recognition Eigenface method is based on the reduction of face dimensional space using Principal Component Analysis (PCA) for facial features. The main purpose of the use of PCA on face recognition using Eigen faces was formed (face space) by finding the eigenvector corresponding to the largest eigenvalue of the face image. The area of this project face detection system with face recognition is Image processing. The software requirements for this project is numpy software.

1.1EXISTING SYSTEM

The System is on using the LBP cascade classifiers to where we try to predict t the object rectangular region in specific front_object rectangular region. Basically in LBP, For LBP, a binary pattern is extracted inside a given rectangular region. In this paper, we simplify the computational complexity of LBP features for fast feature extraction time. To achieve this, we quantize the gradient angle into 2 orientations(horizontal and vertical axes).The LBP front_object xml files have many training data in order to locate the rectangular portion of the object in particular. Here the Algorithms are predefined positive and negative images which will define all the objects in the

Disadvantages:

- LBP is less accurate when compared to the Haar Cascade Classifier which is implemented in the project.
- In any window inside an image, a huge amount of MB-LBP features can be found. So, during the training age, it is necessary to focus on a small set of critical features, discarding most of the non-critical ones in order to increase classification speed significantly without affecting accuracy

1.2 PROPOSED SYSTEM The proposed system consists of both hardware units and software.To implement such a project, the main and most important step was finding the hardware to use for the device. We have chosen a Raspberry Pi model B3 to use in our device. We have done a lot of research, and compared elements in different microcontrollers, like, cost, processing, and user friendliness. The main reasons why we have chosen this specific element are the high processing capacity, relatively low price, and its ability to adapt in different programming modes.

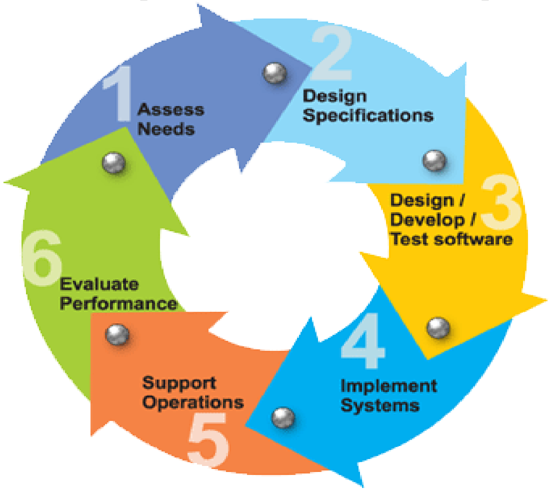
we intend to implement the Haar-Classifier for Object detection and tracking based on the Haar-Features on System on Chip(SoC) for use in a human machine inter object and action interpretation. Haar-like features can be computed at any scale or location in constant time using the integral image representation for image-like rectangle features have become a popular choice as image features in the context of recognition. We compare our rectangular features with Haarlikefeatures Haar-like features are attributes extracted from images used in pattern recognition. Their name comes from their similarity to Haar wavelets. The utilization of these features instead of handling gray or color level of the pixels directly was proposed in . First, the pixel values inside the black area are added together; then the values in the white area are summed, Then the total value of the white area is subtracted from the total value of the black area. This result is used to categorize image sub-regions.
The image that is passed to detect the object using the haar cascade classifier plots the rectangular region in most accurate manner.

Advantages:

- Object detection and tracking is being challenging for many researchers with real time Image sensor.
- With the advancement the real time object detection in remote monitoring is help for building much efficient application. Moreover such technology can be useful in tracking the lost object under dynamic environment.

Software Development Life Cycle:-

There is various software development approaches defined and designed which are used/employed during development process of software, these approaches are also referred as "Software Development Process Models". Each process model follows a particular life cycle in order to ensure success in process of software development.



Requirements

Business requirements are gathered in this phase. This phase is the main focus of the project managers and stake holders. Meetings with managers, stake holders and users are held in order to determine the requirements. Who is going to use the system? How will they use the system? What data should be input into the system? What data should be output by the system? These are general questions that get answered during a requirements gathering phase. This produces a nice big list of functionality that the system should provide, which describes functions the system should perform, business logic that processes data, what data is stored and used by the system, and how the user interface should work. The overall result is the system as a whole and how it performs, not how it is actually going to do it.

Design

The software system design is produced from the results of the requirements phase. Architects have the ball in their court during this phase and this is the phase in which their focus lies. This is where the details on how the system will work is produced. Architecture, including hardware and software, communication, software design (UML is produced here) are all part of the deliverables of a design phase.

Implementation

Code is produced from the deliverables of the design phase during implementation, and this is the longest phase of the software development life cycle. For a developer, this is the main focus of the life cycle because this is where the code is produced. Implementation my overlap with both the design and testing phases. Many tools exists (CASE tools) to actually automate the production of code using information gathered and produced during the design phase.

Testing

During testing, the implementation is tested against the requirements to make sure that the product is actually solving the needs addressed and gathered during the requirements phase. Unit tests and system/acceptance tests are done during this phase. Unit tests act on a specific component of the system, while system tests act on the system as a whole.

So in a nutshell, that is a very basic overview of the general software development life cycle model. Now let's delve into some of the traditional and widely used variations.

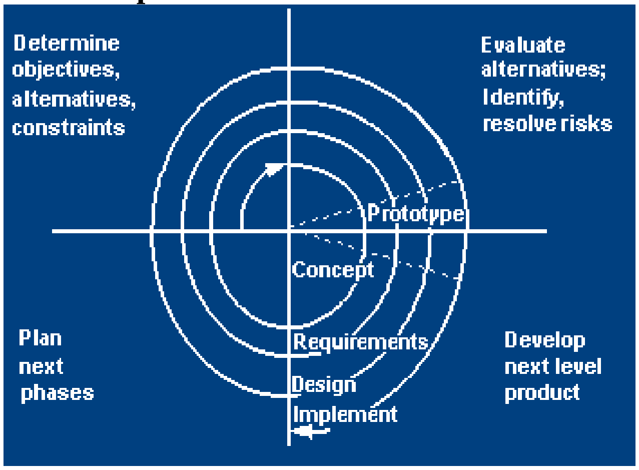
<u>SDLC METHDOLOGIES</u>

This document play a vital role in the development of life cycle (SDLC) as it describes the complete requirement of the system. It means for use by developers and will be the basic during testing phase. Any changes made to the requirements in the future will have to go through formal change approval process.

SPIRAL MODEL was defined by Barry Boehm in his 1988 article, "A spiral Model of Software Development and Enhancement. This model was not the first model to discuss iterative development, but it was the first model to explain why the iteration models.

As originally envisioned, the iterations were typically 6 months to 2 years long. Each phase starts with a design goal and ends with a client reviewing the progress thus far. Analysis and engineering efforts are applied at each phase of the project, with an eye toward the end goal of the project.

The following diagram shows how a spiral model acts like:


Determine objectives, alternatives, constraints
Evaluate alternatives; Identify, resolve risks
Prototype
Concept
Requirements
Design
Implement
Plan next phases
Develop next level product

The steps for Spiral Model can be generalized as follows:

- The new system requirements are defined in as much details as possible. This usually involves interviewing a number of usersrepresenting all the external or internal users and other aspects of the existing system.
- A preliminary design is created for the new system.
- A first prototype of the new system is constructed from the preliminary design. This is usually a scaled-down system, and represents an approximation of the characteristics of the final product.
- A second prototype is evolved by a fourfold procedure:
    1. Evaluating the first prototype in terms of its strengths, weakness, and risks.
    2. Defining the requirements of the second prototype.

3. Planning an designing the second prototype.
4. Constructing and testing the second prototype.

- At the customer option, the entire project can be aborted if the risk is deemed too great. Risk factors might involved development cost overruns, operating-cost miscalculation, or any other factor that could, in the customer's judgment, result in a less-than-satisfactory final product.
- The existing prototype is evaluated in the same manner as was the previous prototype, and if necessary, another prototype is developed from it according to the fourfold procedure outlined above.
- The preceding steps are iterated until the customer is satisfied that the refined prototype represents the final product desired.
- The final system is constructed, based on the refined prototype.
- The final system is thoroughly evaluated and tested. Routine maintenance is carried on a continuing basis to prevent large scale failures and to minimize down time.

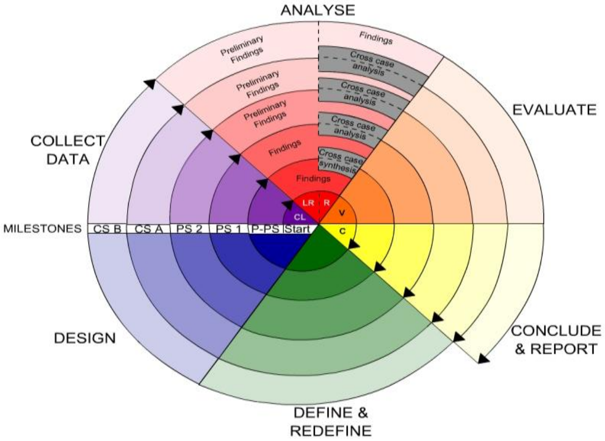
2 STUDY OF THE SYSTEM

In the flexibility of uses the interface has been developed a graphics concepts in mind, associated through a browser interface. The GUI's at the top level has been categorized as follows

1. Administrative User Interface Design
2. The Operational and Generic User Interface Design

The administrative user interface concentrates on the consistent information that is practically, part of the organizational activities and which needs proper authentication for the data collection. The Interface helps the administration with all the transactional states like data insertion, data deletion, and data updating along with executive data search capabilities.

The operational and generic user interface helps the users upon the system in transactions through the existing data and required services. The operational user interface also helps the ordinary users in managing their own information helps the ordinary users in managing their own information in a customized manner as per the assisted flexibilities.



3. Fundamental Concepts on (Domain)

What is Cloudera?

Cloudera is revolutionizing enterprise data management by offering the first unified Platform for Big Data: The Enterprise Data Hub. Cloudera offers enterprises one place to store, process, and analyze all their data, empowering them to extend the value of existing investments while enabling fundamental new ways to derive value from their data.

Why do customers choose Cloudera?
Cloudera was the first commercial provider of python-related software and services and has the most customers with enterprise requirements, and the most experience supporting them, in the industry. Cloudera's combined offering of differentiated software (open and closed source), support, training, professional services, and indemnity brings customers the greatest business value, in the shortest amount of time, at the lowest TCO.

SOFTWARE MODEL OR ARCHITECTURE ANALYSIS:

Structured project management techniques (such as an SDLC) enhance management's control over projects by dividing complex tasks into manageable sections. A software life cycle model is either a descriptive or prescriptive characterization of how software is or should be developed. But none of the SDLC models discuss the key issues like Change management, Incident management and Release management processes within the SDLC process, but, it is addressed in the overall project management. In the proposed hypothetical model, the concept of user-developer interaction in the conventional SDLC model has been converted into a three dimensional model which comprises of the user, owner and the developer. In the proposed hypothetical model, the concept of user-developer interaction in the conventional SDLC model has been converted into a three dimensional model which comprises of the user, owner and the developer. The ―one size fits all‖ approach to applying SDLC methodologies is no longer appropriate. We have made an attempt to address the above mentioned defects by using a new hypothetical model for SDLC described elsewhere. The drawback of addressing these management processes under the overall project management is missing of key technical issues pertaining to software development process that is, these issues are talked in the project management at the surface level but not at the ground level.

Proposed System:

Our goal is to implement machine learning model in order to classify, to the highest possible degree of accuracy, credit card fraud from a dataset gathered from Kaggle. After initial data exploration, we knew we would implement a logistic regression model for best accuracy reports.

Logistic regression, as it was a good candidate for binary classification. Python sklearn library was used to implement the project, We used Kaggle datasets for Credit card fraud detection, using pandas to data frame for class ==0 forno fraud and class==1 for fraud, matplotlib for plotting the fraud and non fraud data, train_test_split for data extraction (Split arrays or matrices into random train and test subsets) and used Logistic Regression machine learning algorithm for fraud detection and print predicting score according to the algorithm. Finally Confusion matrix was plotted on true and predicted.

Functional requirements

Outputs from computer systems are required primarily to communicate the results of processing to users. They are also used to provide a permanent copy of the results for later consultation. The various types of outputs in general are:

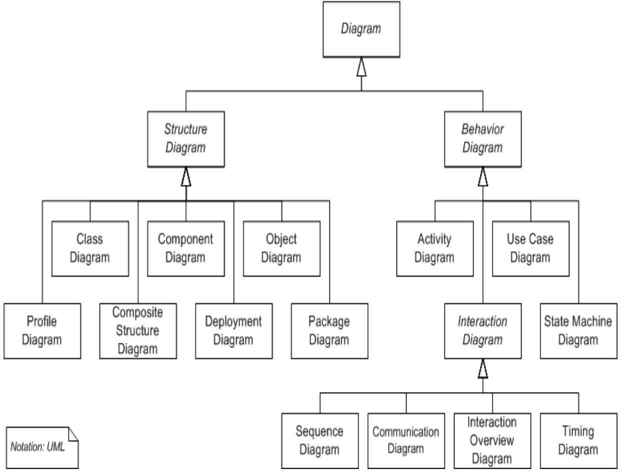
- External Outputs, whose destination is outside the organization,.
- Internal Outputs whose destination is within organization and they are the
- User's main interface with the computer.
- Operational outputs whose use is purely within the computer department.
- Interface outputs, which involve the user in communicating directly.
- Understanding user's preferences, expertise level and his business requirements through a friendly questionnaire.
- Input data can be in four different forms - Relational DB, text files, .xls and xml files. For testing and demo you can choose data from any domain. User-B can provide business data as input.

Non-Functional Requirements

1. Secure access of confidential data (user's details). SSL can be used.
2. 24 X 7 availability.
3. Better component design to get better performance at peak time
4. Flexible service based architecture will be highly desirable for future extension

modeling. It can be used with all processes, throughout the software development life cycle, and across different implementation technologies· UML has synthesized the notations of the Booch method, the Object-modeling technique (OMT) and Object-oriented software engineering (OOSE) by fusing them into a single, common and widely usable modeling language. UML aims to be a standard modeling language which can model concurrent and distributed systems.

UML Diagrams Overview

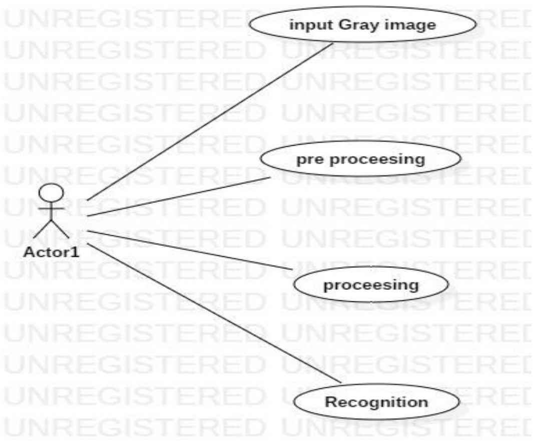


UML combines best techniques from data modeling (entity relationship diagrams), business modeling (work flows), object modeling, and component modeling. It can be used with all processes, throughout the software development life cycle, and across different implementation technologies· UML has synthesized the notations of the Booch method, the Object-modeling technique (OMT) and Object-oriented software engineering (OOSE) by fusing them into a single, common and widely usable modeling language. UML aims to be a standard modeling language which can model concurrent and distributed systems.

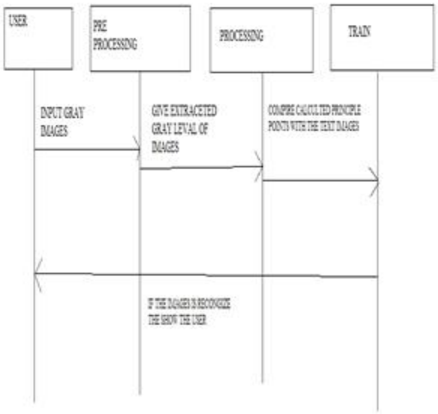
UMLS ON FACE RECOGNITION

UML DIAGRAMS:

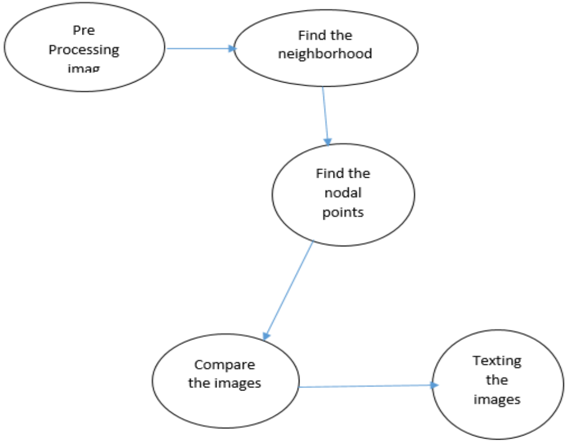
USE CASE DIAGRAM:



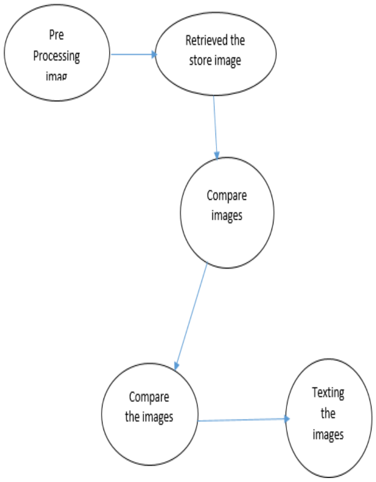
SEQUENCE DIAGRAM:


USER
PRE PROCESSING
PROCESSING
TRAIN
INPUT GRAY IMAGES
GIVE EXTRACETED GRAY LEVAL OF IMAGES
COMPRE CALCULTED PRINCIPLE POINTS WITH THE TEXT IMAGES
IF THE IMAGES IS RECONGIZE THE SHOW THE USER

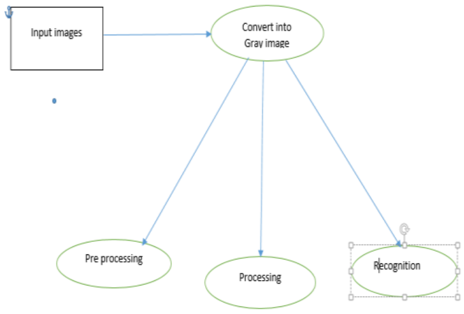
Processing:



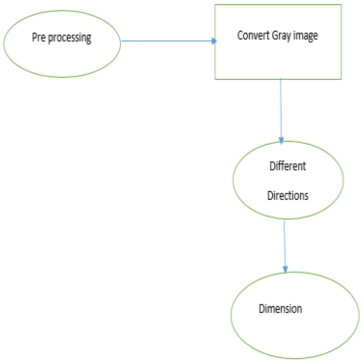
Recognition



Zero level:



Zero level 1:



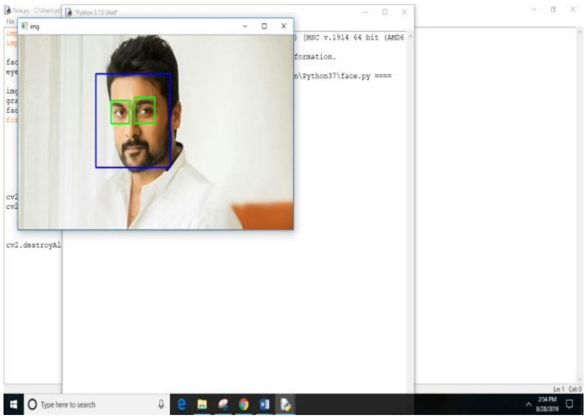
Application of predictive modeling inface recognition

The face is one of the easiest ways to distinguish the individual identity of each other. Face recognition is a personal identification system that uses personal characteristics of a person to identify the person's identity. Human face recognition procedure basically consists of two phases, namely face detection, where this process takes place very rapidly in humans, except under conditions where the object is located at a short distance away, the next is the introduction, which recognize a face as individuals. Stage is then replicated and developed as a model for facial image recognition (face recognition) is one of the much-studied biometrics technology and developed by experts. There are two kinds of methods that are currently popular in developed face recognition pattern namely, Eigenface method and Fisherface method. Facial image recognition Eigenface method is based on the reduction of facedimensional space using Principal Component Analysis (PCA) for facial features. The main purpose of the use of PCA on face recognition using Eigen faces was formed (face space) by finding the eigenvector corresponding to the largest eigenvalue of the face image. The area of this project face detection system with face recognition is Image processing. The software requirements for this project is numpy moduls

Keywords: face detection, Eigen face, Cv2,Numpy

Extension: There are vast number of applications from this face detection project, this project can be extended that the various parts in the face can be detect which are in various directions and shapes.

OUTPUT SCREENS



```
imp
imp

fac
eye

img
gra
fac
for

cv2
cv2

cv2.destroyAl
```

```
] [MSC v.1914 64 bit (AMD6
formation.
n\Python37\face.py ====
```

6. CONCLUSTION

Face recognition technology has come a long way in the last twenty years. Today, machines are able to automatically verify identity information for secure transactions, for surveillance and security tasks, and for access control to buildings.

7. REFERENCES

[1] Zhen Hu, Jibe Zhu, and Ken Tse “Stocks Market Prediction Using Support Vector Machine”,6TH Conference on Information Management,Innovation Management and Industrial Engineering, 2013.M.

[2] Wei Huang, Yoshiteru Nakamori, Shou-Yang Wang, “Forecasting stock market movement direction with support vector machine”, Computers & Operations Research, Volume 32, Issue 10, October 2005, Pages 2513–2522.

[3] N. Ancona, Classification Properties of Support Vector Machines for Regression, Technical Report, RIIESI/CNR-Nr. 02/99.

[4 K. jae Kim, “Financial time series forecasting using support vector machines,” Neurocomputing, vol. 55, 2003.

[5] Debashish Das and Mohammad shorif uddin data mining and neural network techniques in stock market prediction: a methodological review, international journal of artificial intelligence & applications, vol.4, no.1, January 2013