

MISSING CHILD DISTINGUISHING FRAMEWORK USING DEEP LEARNING AND SVM

MINI PROJECT

Submitted by

M. SAI NEHA
20RH1A05E1

M. SRILAXMI
20RH1A05E9

M. SRIJA
20RH1A05F0

P. SNEHA
20RH1A05H3

Under the Esteemed Guidance of

Mrs.V.PRABHAVATHI

Assistant Professor

In partial fulfillment of the Academic Requirements for the Degree of

BACHELOR OF TECHNOLOGY

Computer Science and Engineering



MALLA REDDY ENGINEERING COLLEGE FOR WOMEN

(Autonomous Institution, UGC, Govt. of India)

Accredited by NBA & NAAC with 'A' Grade, UGC, Govt. of India NIRF Indian Ranking-2018, Accepted by MHRD, Govt. of India Permanently Affiliated to JNTUH, Approved by AICTE, ISO 9001:2015 Certified Institution AAAA+ Rated by Digital Learning Magazine, AAA+ Rated by Careers 360 Magazine, 6th Rank CSR, Platinum Rated by AICTE-CII Survey, Top 100 Rank band by ARIIA, MHRD, Govt. of India National Ranking-Top 100 Rank band by Outlook, National Ranking-Top 100 Rank band by Times News Magazine Maisammaguda, Dhullapally, Secunderabad, Kompally, 500100.

2023-2024



MALLA REDDY ENGINEERING COLLEGE FOR WOMEN
(Autonomous Institution-UGC, Govt. of India)

Accredited by NBA & NAAC with 'A' Grade UGC, Govt. of India
National Ranking by NIRF Innovation – Rank Band (151-300), MHRD, Govt. of India
Maisammaguda, Dhulapally, Secunderabad-500100

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING

CERTIFICATE

This is to certify that the project report entitled **“MISSING CHILD DISTINGUISHING
FRAMEWORK USING DEEP LEARNING AND SVM”** being submitted by

M. SAI NEHA – 20RH1A05E1

M. SRI LAXMI- 20RH1A05E9

M.SRIJA - 20RH1A05F0

P.SNEHA - 20RH1A05H3

in partial fulfillment for the award of the Degree of **Bachelor of Technology
in Computer Science and Engineering** to the **Malla Reddy Engineering College for
Women (Autonomous), Maisammaguda, Secunderabad** during the **academic year
2023-2024** is a record of bonafide piece of work, undertake by him/her supervision of the
undersigned.

Approved and Supervised by
Mrs.V. PRABHAVATHI
Assistant Professor

project coordinator
Dr. K. Jaya Rajan
Professor

Head of the Department
Dr. C.V.P.R. PRASAD
Professor and HOD

EXTERNAL EXAMINER



CERTIFICATION OF PROJECT COMPLETION

Date: 30-06-2023

This is to certify that the project entitled **"MISSING CHILD DISTINGUISHING FRAMEWORK USING DEEP LEARNING AND SVM"** Submitted in fulfilment of the requirements for the degree in **B. Tech (COMPUTER SCIENCE AND ENGINEERING)** From Malla Reddy Engineering College for Women, Maisammaguda, Hyderabad is a record of bonafide work carried out by

1. M. SAINHA

2. M. SRILAXMI

3. M. SRIJA

4. M.SNEHA



(20RH1A05E1)

(20RH1A05E9)

(20RH1A05F0)

(20RH1A05H3)

Under our supervision and guidance. They displayed analytical capability, had innovation approach to solve problems and produced good result.

For SLC SERVICES

Managing Director

Regards
Managing Director



reality



ACKNOWLEDGEMENT

We feel ourselves honored and privileged to place our warm salutation to our college **Malla Reddy Engineering College for Women** and Department of **Computer Science and Engineering** which gave us the opportunity to have expertise in engineering and profound technical knowledge.

We would like to deeply thank our Honorable Minister of Telangana State **Sri. Ch. MALLA REDDY Garu**, Founder Chairman MRGI, the largest cluster of institutions in the state of Telangana for providing us with all the resources in the college to make our project success.

We wish to convey gratitude to our **Principal Dr. Y. MADHAVEELATHA**, for providing us with the environment and mean to enrich our skills and motivating us in our endeavor and helping us to realize our full potential.

We would like to thank **Prof. A. Radha Rani**, Director of Computer Science and Engineering & Information Technology for encouraging us to take up a project on this subject and motivating us towards the Project Work.

We express my sincere gratitude to **Dr. C.V.P.R. Prasad**, Head of the Department of Computer Science and Engineering for inspiring us to take up a project to this subject and successfully guiding us towards its completion.

We would like to thank our Internal Guide **Mrs.V. Prabhavathi Assistant Professor** and all the Faculty members for their valuable guidance and encouragement towards the completion of our project work.

With Regards and Gratitude,
M. SRIJA (20RH1A05F0)
M. SAINHA (20RH1A05E1)
M. SRILAXMI (20RH1A05E9)
P. SNEHA (20RH1A05H3)

DECLARATION

We hereby declare that our project entitled **“MISSING CHILD DISTINGUISHING FRAMEWORK USING DEEP LEARNING AND SVM”** submitted to **Malla Reddy Engineering College for Women, Hyderabad** for the award of the Degree of Bachelor of Technology in computer science and Engineering is a result of original research work done by us.

It is declared that the project report or any part thereof has not been previously submitted to any University or Institute for the award of Degree.

M. SRIJA (20RH1A05F0)

M. SAINHA (20RH1A05E1)

M. SRILAXMI (20RH1A05E9)

P. SNEHA (20RH1A05H3)

ABSTRACT

In India a countless number of children are reported missing every year. Among the missing child cases a large percentage of children remain untraced. This paper presents a novel use of deep learning methodology for identifying the reported missing child from the photos of multitude of children available, with the help of face recognition. The public can upload photographs of suspicious child into a common portal with landmarks and remarks. The photo will be automatically compared with the registered photos of the missing child from the repository. Classification of the input child image is performed and photo with best match will be selected from the database of missing children. For this, a deep learning model is trained to correctly identify the missing child from the missing child image database provided, using the facial image uploaded by the public. The Convolutional Neural Network (CNN), a highly effective deep learning technique for image based applications is adopted here for face recognition. Face descriptors are extracted from the images using a pre-trained CNN model VGG-Face deep architecture. Compared with normal deep learning applications, our algorithm uses convolution network only as a high level feature extractor and the child recognition is done by the trained SVM classifier. Choosing the best performing CNN model for face recognition, VGG-Face and proper training of it results in a deep learning model invariant to noise, illumination, contrast, occlusion, image pose and age of the child and it outperforms earlier methods in face recognition based missing child identification. The classification performance achieved for child identification system is 99.41%. It was evaluated on 43 Child cases.

Keywords— Missing kid identification, face recognition, deep learning, CNN, VGG-Face, Multi class SVM.

CONTENT

1. INTRODUCTION	1-2
1.1 Motivation	
1.2 Objective of the project	
2. LITERATURE SURVEY	3-7
2.1 Face Recognition Using Histograms of oriented gradients	
2.2 Face Recognition Using Sift Features	
2.3 Missing Child Identification Using Face Recognition system	
2.4 Very Deep Convolutional Networks For Large Scale Image Recognition	
2.5 Software Environment	
2.6 Why Choose Python	
3.SYSTEMANALYSIS	8-11
3.1 Existing System	
3.1.1 Disadvantages of Existing System	
3.2 Proposed System	
3.3 Software Environment	
3.4 Hard Requirements	
3.5 Software Requirements	
4 SYSTEMDESIGN	12-19
4.1 System Architecture	
4.2 Data Flow Diagram	
4.3 UML Diagram	
4.3.1 Use Case Diagram	
4.3.2 Class Diagram	

4.3.3 Sequence Diagram

4.3.4 Activity Diagram

5. IMPLEMENTATION

20-25

5.1 Modules

5.2 Source Code

6. SYSTEM TESTING

26-30

6.1 Types of Tests

6.1.1 Unit Testing

6.1.2 Integration Testing

6.1.3 Functional Testing

6.1.4 System Testing

6.1.5 White Box Testing

6.1.6 Black Box Testing

7. OUTPUT

31-33

8. CONCLUSION

34

9. REFERENCES

35

LIST OF SCREENSHOTS

S No	SCREEN NAME	Page No
1	Public Can Click On “Public Upload The Suspected Child “In This.	31
2	Public Can Upload The Details In This Screen	32
3	view public upload missing child status in this screen	32
4	Official login screen	33
5	view all the details of the public uploaded photos	33

LIST OF FIGURES

S No	FIGURE NAME	Page No
1	System Architecture	12
2	Data Flow Diagram	13
3	Use Case Diagram	16
4	Class Diagram	17
5	Sequence Diagram	18
6	Activity Diagram	19

1. INTRODUCTION

1.1 MOTIVATION

India is the second populous country in the world and children represent a significant percentage of total population. But unfortunately, a large number of children go missing every year in India due to various reasons including abduction or kidnapping, run-away children, trafficked children and lost children. A deeply disturbing fact about India's missing children is that while on an average 174 children go missing every day, half of them remain untraced. Children who go missing may be exploited and abused for various purposes. As per the National Crime Records Bureau (NCRB) report which was cited by the Ministry of Home Affairs (MHA) in the Parliament (LS Q no. 3928, 20-03- 2018), more than one lakh children (1,11,569 in actual numbers) were reported to have gone missing till 2016, and 55,625 of them remained untraced till the end of the year. Many NGOs claim that estimates of missing children are much higher than reported.

The public is given provision to voluntarily take photographs of children in suspected situations and uploaded in that portal. Automatic searching of this photo among the missing child case images will be provided in the application. This supports the police officials to locate the child anywhere in India. When a child is found, the photograph at that time is matched against the images uploaded by the Police/guardian at the time of missing. Sometimes the child has been missing for a long time. This age gap reflects in the images since aging affects the shape of the face and texture of the skin. The feature discriminator invariant to aging effects has to be derived. This is the challenge in missing child identification compared to the other face recognition systems. Also facial appearance of child can vary due to changes in pose, orientation, illumination, occlusions, noise in background etc. The image taken by public may not be of good quality, as some of them may be captured from a distance without the knowledge of the child.

Introduction Related work Proposed System Algorithm References Introduction 4/24 • Many NGOs claim that estimates of missing children are much higher than reported. The missing from one region may be found in another region or another state, for various reasons. So even if a child is found, it is difficult to identify him/her from the reported missing cases. • A framework and methodology for developing an assistive tool for tracing missing child is described in this paper. An idea for maintaining a virtual space is proposed, such that the recent photographs of children given by parents at the time of reporting missing cases is saved in a repository Abstract Introduction Related work Proposed System Algorithm References Introduction 4/24 • The public is given provision to voluntarily take photographs of children in suspected situations and uploaded in that portal. Automatic searching of this photo among the missing child case images will be provided in the application. This supports the police officials to locate the child anywhere in India

1.2 OBJECTIVE OF THE PROJECT

The public is given provision to voluntarily take photographs of children in suspected situations and uploaded in that portal. Automatic searching of this photo among the missing child case images will be provided in the application. This supports the police officials to locate the child anywhere in India. When a child is found, the photograph at that time is matched against the images uploaded by the Police/guardian at the time of missing. Sometimes the child has been missing for a long time. This age gap reflects in the images since aging affects the shape of the face and texture of the skin. The feature discriminator invariant to aging effects has to be derived. This is the challenge in missing child identification compared to the other face recognition systems. Also facial appearance of child can vary due to changes in pose, orientation, illumination, occlusions, noise in background etc. The image taken by public may not be of good quality, as some of them may be captured from a distance without the knowledge of the child. A deep learning [1] architecture considering all these constraints is designed here

2. LITERATURE SURVEY

2.1 FACE RECOGNITION USING HISTOGRAMS OF ORIENTED GRADIENTS

AUTHORS: O. Deniz, G. Bueno, J. Salido, and F. D. la Torre

Abstract: Face recognition has been a long standing problem in computer vision. Recently, Histograms of Oriented Gradients (HOGs) have proven to be an effective descriptor for object recognition in general and face recognition in particular. In this paper, we investigate a simple but powerful approach to make robust use of HOG features for face recognition. The three main contributions of this work are: First, in order to compensate for errors in facial feature detection due to occlusions, pose and illumination changes, we propose to extract HOG descriptors from a regular grid. Second, fusion of HOG descriptors at different scales allows to capture important structure for face recognition. Third, we identify the necessity of performing dimensionality reduction to remove noise and make the classification process less prone to overfitting. This is particularly important if HOG features are extracted from overlapping cells. Finally, experimental results on four databases illustrate the benefits of our approach.

2.1 FACE RECOGNITION USING SIFT FEATURES

AUTHORS: C. Geng and X. Jiang

Abstract: Scale Invariant Feature Transform (SIFT) has shown to be a powerful technique for general object recognition/detection. In this paper, we propose two new approaches: Volume-SIFT (VSIFT) and Partial-Descriptor-SIFT (PDSIFT) for face recognition based on the original SIFT algorithm. We compare holistic approaches: Fisher face (FLDA), the null space approach (NLDA) and Eigenfeature Regularization and Extraction (ERE) with feature based approaches: SIFT and PDSIFT. Experiments on the ORL and AR databases show that the performance of PDSIFT is significantly better than the original SIFT approach. Moreover, PDSIFT can achieve comparable performance as the most successful holistic approach ERE and significantly outperforms FLDA and NLDA.

2.2 MISSING CHILD IDENTIFICATION USING FACE RECOGNITION SYSTEM

AUTHORS: Rohit Satle, Vishnuprasad Poojary, John Abraham and Shilpa Wakode.

Abstract: The human face plays an important role in our social interaction, conveying people's identity. Face recognition is a task that humans perform routinely and effortlessly in their daily lives. Face recognition, as one of the primary biometric technologies, became more and more important owing to rapid advances in technologies such as digital cameras, the Internet and mobile devices, and increased demands on security. A facial recognition system is a computer application capable of identifying or verifying a person from a digital image or a video frame from a video source. Face Recognition System is a computer based digital technology and is an active area of research. This paper addresses the building of face recognition system by using Principal Component Analysis (PCA) method. The PCA has been extensively employed for face recognition algorithms. It not only reduces the dimensionality of the image, but also retains some of the variations in the image data. The system functions by projecting face image onto a feature space that spans the significant variations among known face images. The significant features are known as "Eigen faces", because they are the eigenvectors (Principal Component) of the set of faces they do not necessarily correspond to the features such as eyes, ears, and noses. The projection operation characterizes an individual face by a weighted sum of the Eigen faces features and so to recognize a particular face it is necessary only to compare these weights to those individuals.

2.4 VERY DEEP CONVOLUTIONAL NETWORKS FOR LARGE-SCALE IMAGE RECOGNITION

AUTHOR: Karen Simonyan and Andrew Zisserman

Abstract: In this work we investigate the effect of the convolutional network depth on its accuracy in the large-scale image recognition setting. Our main contribution is a thorough evaluation of networks of increasing depth using an architecture with very small (3×3) convolution filters, which shows that a significant improvement on the prior-art configurations can be achieved by pushing the depth to 16–19 weight layers. These findings were the basis of our ImageNet Challenge 2014 submission, where our team secured the first and the second places in the localisation and classification tracks respectively. We also show that our representations generalise well to other datasets, where they achieve state-of-the-art results. We have made our two best-performing ConvNet models publicly available to facilitate further research on the use of deep visual representations in computer vision.

2.5 SOFTWARE ENVIRONMENT

Python is a high-level, interpreted scripting language developed in the late 1980s by Guido van Rossum at the National Research Institute for Mathematics and Computer Science in the Netherlands. The initial version was published at the alt. Sources newsgroup in 1991, and version 1.0 was released in 1994.

Python 2.0 was released in 2000, and the 2.x versions were the prevalent releases until December 2008. At that time, the development team made the decision to release version 3.0, which contained a few relatively small but significant changes that were not backward compatible with the 2.x versions. Python 2 and 3 are very similar, and some features of Python 3 have been back ported to Python 2. But in general, they remain not quite compatible.

Both Python 2 and 3 have continued to be maintained and developed, with periodic release updates for both. As of this writing, the most recent versions available are 2.7.15 and 3.6.5. However, an official End of Life date of January 1, 2020 has been established for Python 2, after which time it will no longer be maintained. If you are a newcomer to Python, it is recommended that you focus on Python 3, as this tutorial will do.

Python is still maintained by a core development team at the Institute, and Guido is still in charge, having been given the title of BDFL (Benevolent Dictator For Life) by the Python community. The name Python, by the way, derives not from the snake, but from the British comedy troupe Monty Python's Flying Circus, of which Guido was, and presumably still is, a fan. It is common to find references to Monty Python sketches and movies scattered throughout the Python documentation.

2.6 WHY CHOOSE PYTHON

If you're going to write programs, there are literally dozens of commonly used languages to choose from. Why choose Python? Here are some of the features that make Python an appealing choice.

Python is Popular

Python has been growing in popularity over the last few years. The 2018 Stack Overflow Developer Survey ranked Python as the 7th most popular and the number one most wanted technology of the year. World-class software development countries around the globe use Python every single day.

According to research by Dice Python is also one of the hottest skills to have and the most popular programming language in the world based on the Popularity of Programming Language Index Popularity of Programming Language Index.

Python is interpreted

Many languages are compiled, meaning the source code you create needs to be translated into machine code, the language of your computer's processor, before it can be run. Programs written in an interpreted language are passed straight to an interpreter that runs them directly.

This makes for a quicker development cycle because you just type in your code and run it, without the intermediate compilation step.

One potential downside to interpreted languages is execution speed. Programs that are compiled into the native language of the computer processor tend to run more quickly than interpreted programs. For some applications that are particularly computationally intensive, like graphics processing or intense number crunching, this can be limiting.

In practice, however, for most programs, the difference in execution speed is measured in milliseconds, or seconds at most, and not appreciably noticeable to a human user. The expediency of coding in an interpreted language is typically worth it for most applications.

Python is Free

The Python interpreter is developed under an OSI-approved open-source license, making it free to install, use, and distribute, even for commercial purposes. A version of the interpreter is available for virtually any platform there is, including all flavors of Unix, Windows, macOS, smart phones and tablets, and probably anything else you ever heard of. A version even exists for the half dozen people remaining who use OS/2.

Python is Portable

Because Python code is interpreted and not compiled into native machine instructions, code written for one platform will work on any other platform that has the Python interpreter installed. (This is true of any interpreted language, not just Python.)

Python is Simple

As programming languages go, Python is relatively uncluttered, developers have deliberately kept it that way.

A rough estimate of the complexity of a language can be gleaned from the number of keywords or reserved words in the language. These are words that are reserved for special meaning by the compiler or interpreter because they designate specific built-in functionality of the language. Python 3 has 33 keywords, and Python 2 has 31. By contrast, C++ has 62, Java has 53, and Visual Basic has more than 120, though these latter examples probably vary somewhat by implementation or dialect.

Some things that Python is often used for are:

- Web development
- Scientific programming
- Desktop GUIs
- Network programming
- Game programming

3. SYSTEM ANALYSIS

3.1 EXISTING SYSTEM

Mostly missing child cases are reported to the police. The child missing from one region may be found in another region or another state, for various reasons.

So even if a child is found, it is difficult to identify him/her from the reported missing cases. A framework and methodology for developing an assistive tool for tracing missing child is described in this paper.

An idea for maintaining a virtual space is proposed, such that the recent photographs of children given by parents at the time of reporting missing cases is saved in a repository. The public is given provision to voluntarily take photographs of children in suspected situations and uploaded in that portal

Automatic searching of this photo among the missing child case images will be provided in the application. This supports the police officials to locate the child anywhere in India.

3.1.1 DISADVANTAGES OF EXISTING SYSTEM

Earliest methods for face recognition commonly used computer vision features such as HOG, LBP, SIFT, or SURF. However, features extracted using a CNN network for getting facial representations gives better performance in face recognition than handcrafted features.

3.2 PROPOSED SYSTEM:

This paper presents a novel use of deep learning methodology for identifying the reported missing child from the photos of multitude of children available, with the help of face recognition. The public can upload photographs of suspicious child into a common portal with landmarks and remarks. The photo will be automatically compared with the registered photos of the missing child from the repository. Classification of the input child image is performed and photo with best match will be selected from the database of missing children. For this, a deep learning model is trained to correctly identify the missing child from the missing child image database provided, using the facial image uploaded by the public.

3.3 SOFTWARE ENVIRONMENT

The feasibility of the project is analyzed in this phase and business proposal is put forth with a very general plan for the project and some cost estimates. During system analysis the feasibility study of the proposed system is to be carried out. This is to ensure that the proposed system is not a burden to the company. For feasibility analysis, some understanding of the major requirements for the system is essential.

Three key considerations involved in the feasibility analysis are

- **ECONOMICAL FEASIBILITY**
- **TECHNICAL FEASIBILITY**
- **SOCIAL FEASIBILITY**

ECONOMICAL FEASIBILITY

This study is carried out to check the economic impact that the system will have on the organization. The amount of fund that the company can pour into the research and development of the system is limited. The expenditures must be justified. Thus the developed system as well within the budget and this was achieved because most of the technologies used are freely available. Only the customized products had to be purchased.

TECHNICAL FEASIBILITY

This study is carried out to check the technical feasibility, that is, the technical requirements of the system. Any system developed must not have a high demand on the available technical resources. This will lead to high demands on the available technical resources. This will lead to high demands being placed on the client. The developed system must have a modest requirement, as only minimal or null changes are required for implementing this system.

SOCIAL FEASIBILITY

The aspect of study is to check the level of acceptance of the system by the user. This includes the process of training the user to use the system efficiently. The user must not feel threatened by the system, instead must accept it as a necessity. The level of acceptance by the users solely depends on the methods that are employed to educate the user about the system and to make him familiar with it. His level of confidence must be raised so that he is also able to make some constructive criticism, which is welcomed, as he is the final user of the system.

3.4 HARDWARE REQUIREMENTS:

- System : Pentium Dual Core.
- Hard Disk : 120 GB.
- Monitor : 15'' LED
- Input Devices : Keyboard, Mouse
- Ram : 1 GB

3.5 SOFTWARE REQUIREMENTS:

- Operating system : Windows 10
- Coding Language : python
- Tool : PyCharm
- Database : MYSQL
- Server : Flask

FUNCTIONAL REQUIREMENTS:

In software engineering, a functional requirement defines a system or its component. It describes the functions a software must perform. A function is nothing but inputs, its behavior, and outputs. It can be a calculation, data manipulation, business process, user interaction, or any other specific functionality which defines what function a system is likely to perform.

Functional software requirements help you to capture the intended behavior of the system. This behavior may be expressed as functions, services or tasks or which system is required to perform.

1. ADMIN

2.USER

NON –FUNCTIONAL REQUIREMENTS:

NON-FUNCTIONAL REQUIREMENT (NFR) specifies the quality attribute of a software system. They judge the software system based on Responsiveness, Usability, Security, Portability and other non-functional standards that are critical to the success of the software system. Example of nonfunctional requirement, “how fast does the website load?” Failing to meet non-functional requirements can result in systems that fail to satisfy user needs. Non- functional Requirements allows you to impose constraints or restrictions on the design of the system across the various agile backlogs. Example, the site should load in 3 seconds when the number of simultaneous users are > 10000. Description of non-functional requirements is just as critical as a functional requirement.

- Usability requirement
- Serviceability requirement
- Manageability requirement
- Recoverability requirement
- Security requirement
- Data Integrity requirement
- Capacity requirement
- Availability requirement
- Scalability requirement

4. SYSTEM DESIGN

4.1 SYSTEM ARCHITECTURE:

Here we propose a methodology for missing child identification which combines facial feature extraction based on deep learning and matching based on support vector machine. The proposed system utilizes face recognition for missing child identification. This is to help authorities and parents in missing child investigation. The architecture of the proposed framework is given below.

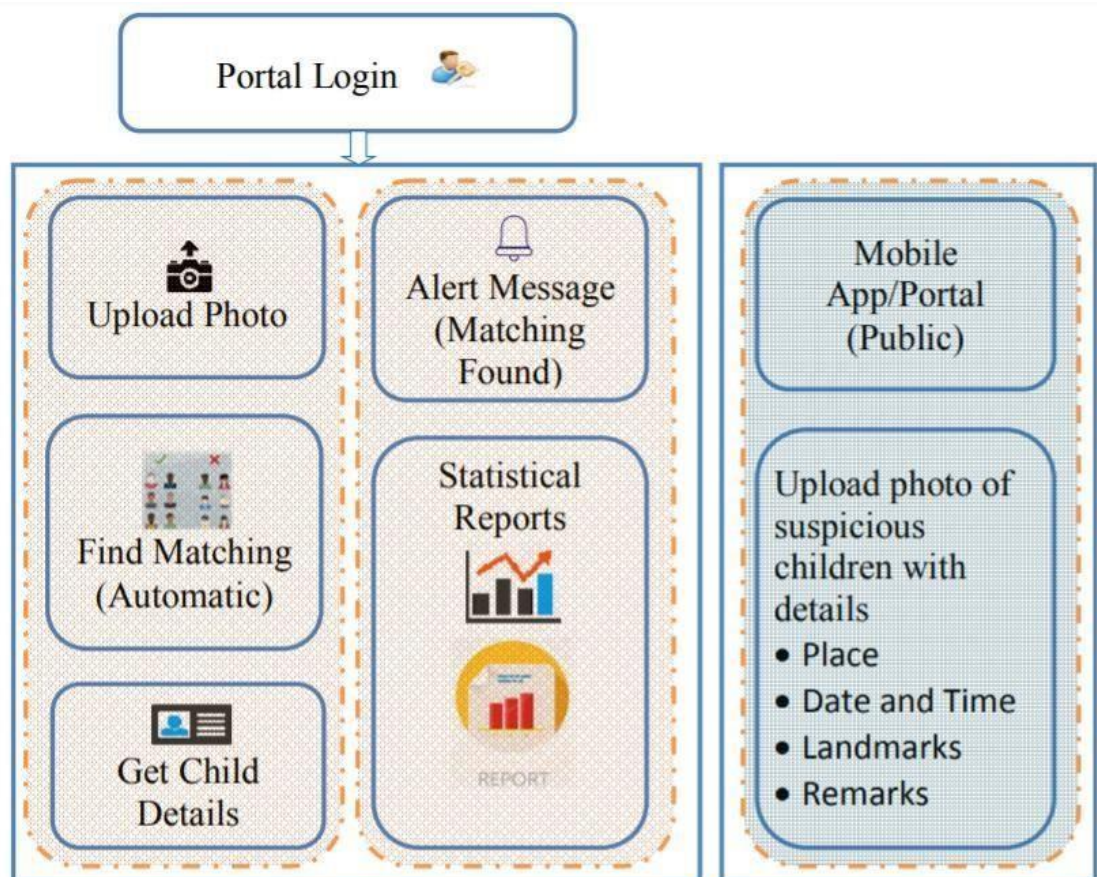


Fig. 1. Architecture of proposed child identification system

Images of reported missing children are saved in a repository and the face area is selected for cropping to obtain input face images. Learned features from a Convolutional Neural Network (CNN), a specific type of deep learning algorithm, are used for training a multi class SVM classifier. This machine learning approach is used to correctly label the child using the name indicated in the database provided by the concerned authority. In the following sections the paper details the work flow for child matching methodology. The flow chart of the automatic child face identification methodology is as shown in Fig 2.

4.2 DATA FLOW DIAGRAM:

1. The DFD is also called as bubble chart. It is a simple graphical formalism that can be used to represent a system in terms of input data to the system, various processing carried out on this data, and the output data is generated by this system.
2. The data flow diagram (DFD) is one of the most important modeling tools. It is used to model the system components. These components are the system process, the data used by the process, an external entity that interacts with the system and the information flows in the system.
3. DFD shows how the information moves through the system and how it is modified by a series of transformations. It is a graphical technique that depicts information flow and the transformations that are applied as data moves from input to output.
4. DFD is also known as bubble chart. A DFD may be used to represent a system at any level of abstraction. DFD may be partitioned into levels that represent increasing information flow and functional detail.

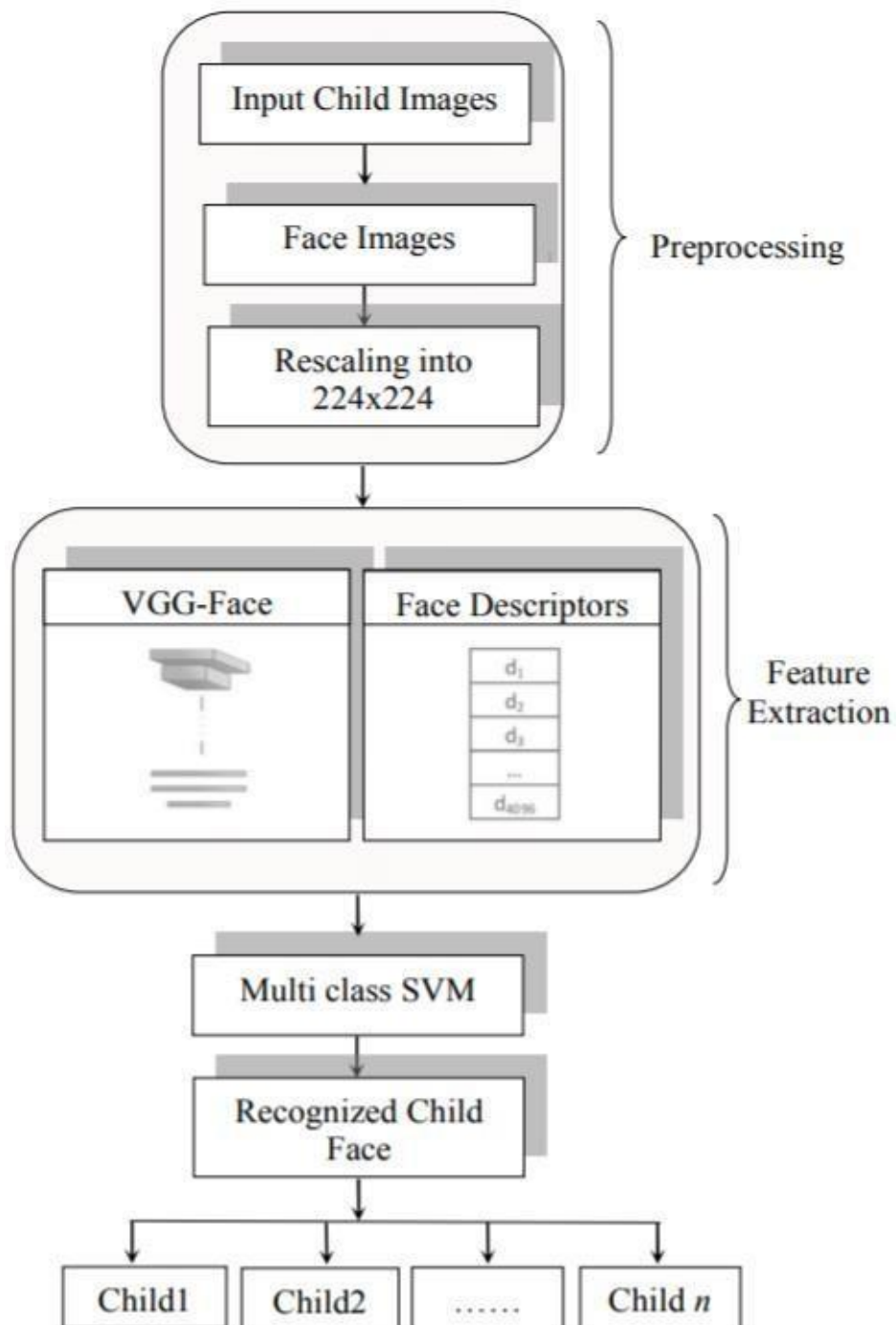


Fig. 2. Software Flow of face recognition system

4.3 UML DIAGRAMS:

UML stands for Unified Modeling Language. UML is a standardized general-purpose modeling language in the field of object-oriented software engineering. The standard is managed, and was created by, the Object Management Group.

The goal is for UML to become a common language for creating models of object oriented computer software. In its current form UML is comprised of two major components

a Meta-model and a notation. In the future, some form of method or process may also be added to; or associated with, UML.

The Unified Modeling Language is a standard language for specifying, Visualization, Constructing and documenting the artifacts of software system, as well as for business modeling and other non-software systems.

The UML represents a collection of best engineering practices that have proven successful in the modeling of large and complex systems.

The UML is a very important part of developing objects oriented software and the software development process. The UML uses mostly graphical notations to express the design of software projects.

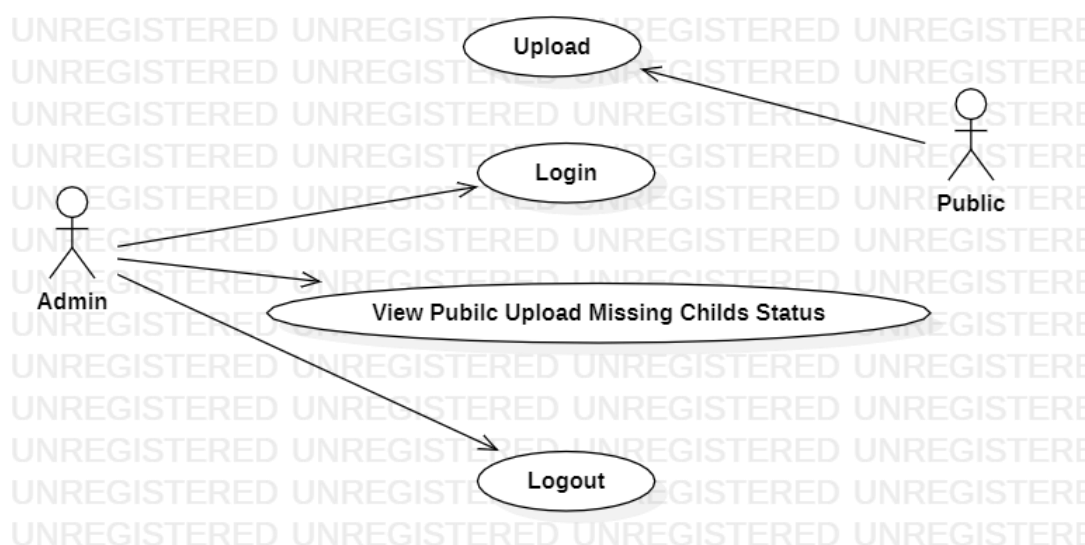
GOALS:

The Primary goals in the design of the UML are as follows:

1. Provide users a ready-to-use, expressive visual modeling Language so that they can develop and exchange meaningful models.
2. Provide extendibility and specialization mechanisms to extend the core concepts.
3. Be independent of particular programming languages and development process.
4. Provide a formal basis for understanding the modeling language.
5. Encourage the growth of OO tools market.
6. Integrate best practices.

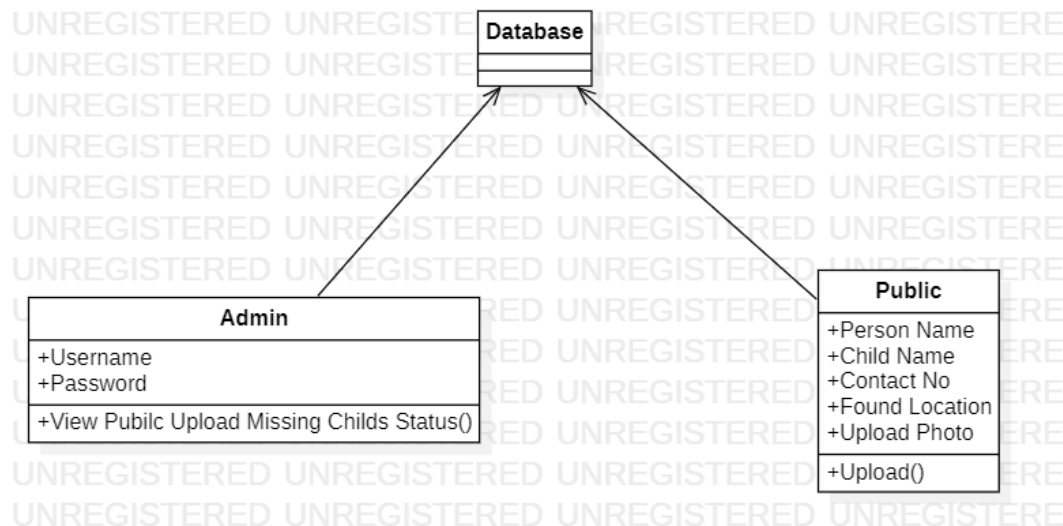
4.3.1 USE CASE DIAGRAM:

A use case diagram in the Unified Modeling Language (UML) is a type of behavioral diagram defined by and created from a Use-case analysis. Its purpose is to present a graphical overview of the functionality provided by a system in terms of actors, their goals (represented as use cases), and any dependencies between those use cases. The main purpose of a use case diagram is to show what system functions are performed for which actor. Roles of the actors in the system can be depicted.



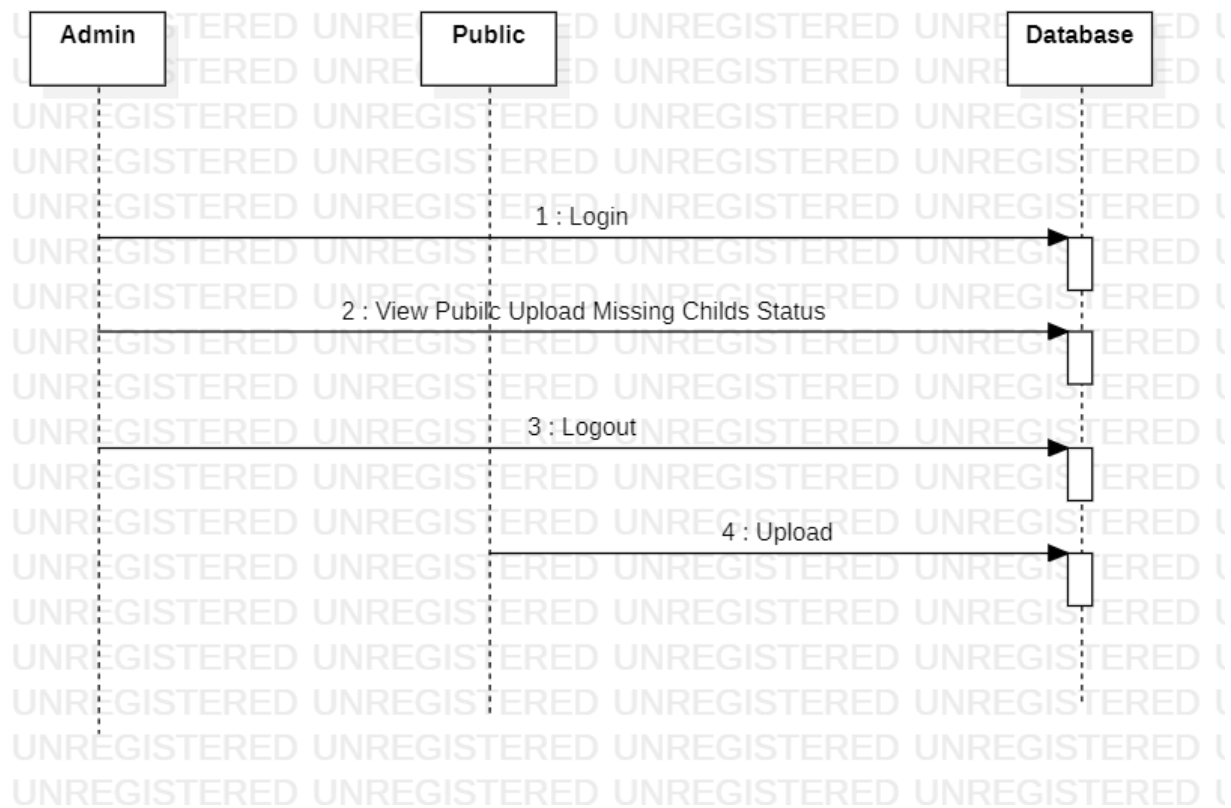
4.3.2 CLASS DIAGRAM:

In software engineering, a class diagram in the Unified Modeling Language (UML) is a type of static structure diagram that describes the structure of a system by showing the system's classes, their attributes, operations (or methods), and the relationships among the classes. It explains which class contains information.



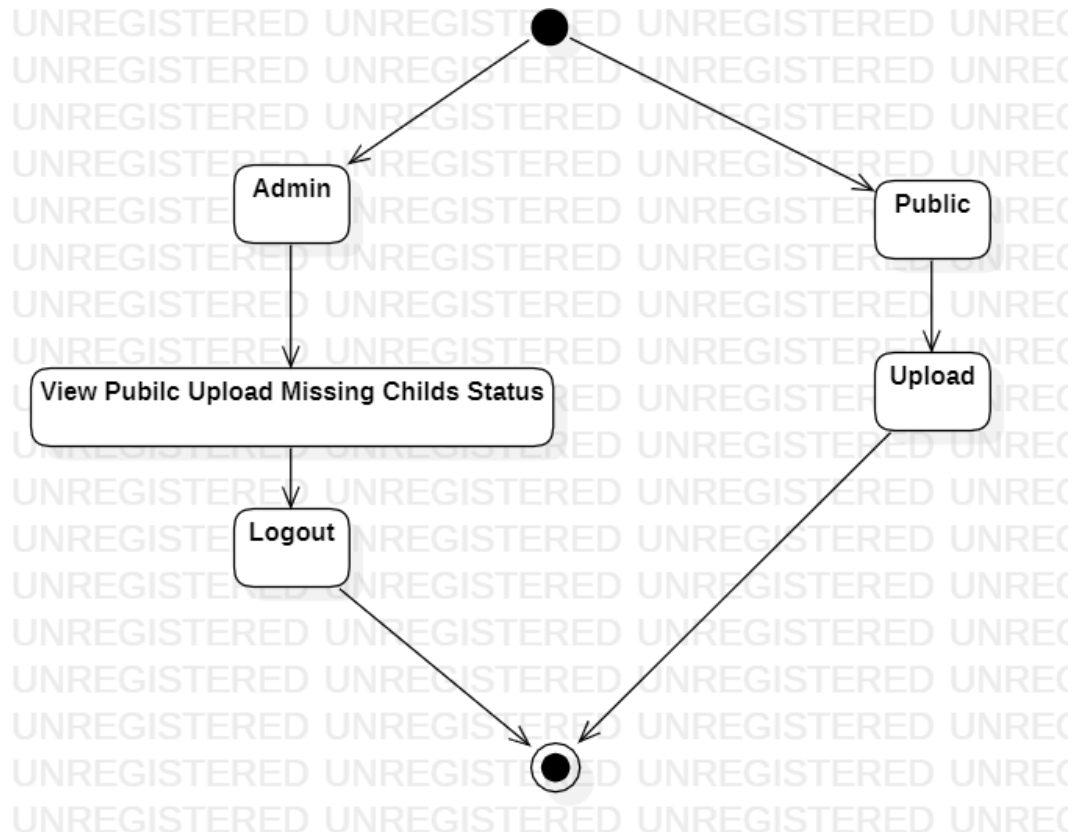
4.3.3 SEQUENCE DIAGRAM:

A sequence diagram in Unified Modeling Language (UML) is a kind of interaction diagram that shows how processes operate with one another and in what order. It is a construct of a Message Sequence Chart. Sequence diagrams are sometimes called event diagrams, event scenarios, and timing diagrams.



4.3.4ACTIVITY DIAGRAM:

Activity diagrams are graphical representations of workflows of stepwise activities and actions with support for choice, iteration and concurrency. In the Unified Modeling Language, activity diagrams can be used to describe the business and operational step-by-step workflows of components in a system. An activity diagram shows the overall flow of contr



5. IMPLEMENTATION

5.1 MODULES:

Preprocessing

Preprocessing input raw image in the context of face recognition involves acquiring the face region and standardizing images in a format compatible with the CNN architecture employed. Each CNN has a different input size requirement. The photographs of missing child acquired by a digital camera or mobile phone are taken and categorized into separate cases for creating the database of face recognition system. The face region in each image is identified and cropped for getting the input face images.

Upload Photo

It consists of a national portal for storing details of missing child along with the photo. Whenever a child missing is reported, along with the FIR, the concerned officer uploads the photo of the missing child into the portal. The public can upload photo of any suspicious child at any time into the portal with details like place, time, landmarks and remarks. The photo uploaded by the users will be automatically compared with photos of the registered missing children and if a matching photo with sufficient score is found, then an alert email will be sent to the concerned officer. The message will also be visible in the message box of the concerned officer login screen.

Search

Whenever users uploads photo of a suspected child, the system generates template vector of the facial features from the uploaded photo. If a matching is found in the repository, the system displays the most matched photo and pushes a message to the concerned Officer portal or Email the alert message of matching child. Similarly the Officer can check for any matching with the database at any time using the proposed system.

MODULES DESCRIPTION:

1. To implements this project we have used FGNET missing child dataset and by this dataset we have build and saved CNN model is available inside model folder.
2. When user upload image used trained CNN model will be applied on test to check whether image is exists in missing child are not.
3. When user found any suspected child roaming on road then user will take image and upload here and then CNN trained model will apply to get missing result. if unidentified image also upload then application say not found.

5.2 SOURCE CODE

Admin.py:

```
from django.contrib import admin  
# Register your models here.
```

Apps.py:

```
from django.apps import AppConfig  
class MissingChildappConfig(AppConfig):  
    name = 'MissingChildApp'
```

Models.py

```
from django.db import models  
# Create your models here.
```

Tests.py

```
from django.test import TestCase  
# Create your tests here.
```

Urls.py

```
from django.urls import path  
from . import views
```

```
urlpatterns = [path("index.html", views.index, name="index"),  
               path('Login.html', views.Login, name="Login"),  
               path('Upload.html', views.Upload, name="Upload"),  
               path('OfficialLogin', views.OfficialLogin, name="OfficialLogin"),  
               path('UploadAction', views.UploadAction, name="UploadAction"),  
               path('ViewUpload', views.ViewUpload, name="ViewUpload"),  
               ]
```

Views.py

```
from django.shortcuts import render  
from django.template import RequestContext  
import pymysql
```

```
from django.http import HttpResponseRedirect
from django.conf import settings
from django.core.files.storage import FileSystemStorage
import datetime
import os
import cv2
import numpy as np
from keras.utils.np_utils import to_categorical
from keras.layers import MaxPooling2D
from keras.layers import Dense, Dropout, Activation, Flatten
from keras.layers import Convolution2D
from keras.models import Sequential
from keras.models import model_from_json
```

```
global index
```

```
index = 0
```

```
global missing_child_classifier
```

```
global cascPath
```

```
global faceCascade
```

```
def index(request):
```

```
    if request.method == 'GET':
```

```
        return render(request, 'index.html', {})
```

```
def Login(request):
```

```
    if request.method == 'GET':
```

```
        return render(request, 'Login.html', {})
```

```
def Login(request):
```

```
    if request.method == 'GET':
```

```
        return render(request, 'Login.html', {})
```



```
def Upload(request):
    if request.method == 'GET':
        return render(request, 'Upload.html', { })

def OfficialLogin(request):
    if request.method == 'POST':
        username = request.POST.get('t1', False)
        password = request.POST.get('t2', False)
        if username == 'admin' and password == 'admin':
            context= {'data': 'welcome '+username}
            return render(request, 'OfficialScreen.html', context)
        else:
            context= {'data': 'login failed'}
            return render(request, 'Login.html', context)

def ViewUpload(request):
    if request.method == 'GET':
        strdata = '<table border=1 align=center width=100%><tr><th>Upload Person  
Name</th><th>Child Name</th><th>Contact No</th><th>Found Location</th><th>Child  
Image <th>Uploaded Date</th><th>Status</th></tr><tr>'
        con = pymysql.connect(host='127.0.0.1',port = 3306,user = 'root', password = 'root',
        database = 'MissingChildDB',charset='utf8')
        with con:
            cur = con.cursor()
            cur.execute("select * FROM missing")
            rows = cur.fetchall()
            for row in rows:
                strdata+='<td>'+row[0]+'</td><td>'+str(row[1])+'</td><td>'+row[2]+'</td><td>'+row[3]+'</td><td><img src=/static/photo/'+row[4]+' width=200 height=200></img></td><td>'
                strdata+=str(row[5])+'</td><td>'+str(row[6])+'</td></tr>'
            context= {'data':strdata}
        return render(request, 'ViewUpload.html', context
```

```
def UploadAction(request):
    global index
    global missing_child_classifier
    global cascPath
    global faceCascade
    if request.method == 'POST' and request.FILES['t5']:
        output = "
person_name = request.POST.get('t1', False)
child_name = request.POST.get('t2', False)
contact_no = request.POST.get('t3', False)
        location = request.POST.get('t4', False)
myfile = request.FILES['t5']
        fs = FileSystemStorage()
        filename =
fs.save('C:/Python/MissingChilds/MissingChildApp/static/photo/'+child_name+'.png',
myfile)
        #if index == 0:
cascPath = "haarcascade_frontalface_default.xml"
faceCascade = cv2.CascadeClassifier(cascPath)
        #index = 1
        option = 0;
        frame = cv2.imread(filename)
gray = cv2.cvtColor(frame, cv2.COLOR_BGR2GRAY)
        faces = faceCascade.detectMultiScale(gray,1.3,5)
print("Found {0} faces!".format(len(faces)))
img = "
        status = 'Child not found in missing database'
        if len(faces) > 0:
            for (x, y, w, h) in faces:
img = frame[y:y + h, x:x + w]
        option = 1
```

```
if option == 1:
    with open('model/model.json', "r") as json_file:
loaded_model_json = json_file.read()
missing_child_classifier = model_from_json(loaded_model_json)
missing_child_classifier.load_weights("model/model_weights.h5")
    missing_child_classifier._make_predict_function()
img = cv2.resize(img, (64,64))
    im2arr = np.array(img)
    im2arr = im2arr.reshape(1,64,64,3)
img = np.asarray(im2arr)
img = img.astype('float32')
img = img/255
    preds = missing_child_classifier.predict(img)
    if(np.amax(preds) > 0.60):
        status = 'Child found in missing database'
    now = datetime.datetime.now()
current_time = now.strftime("%Y-%m-%d %H:%M:%S")
    filename = os.path.basename(filename)
db_connection = pymysql.connect(host='127.0.0.1',port = 3308,user = 'root', password =
'root', database = 'MissingChildDB',charset='utf8')
db_cursor = db_connection.cursor()
    query = "INSERT INTO
missing(person_name,child_name,contact_no,location,image,upload_date,status)
VALUES('"+person_name+"','"+child_name+"','"+contact_no+"','"+location+"','"+filename+"
','"+str(current_time)+"','"+status+"')"
db_cursor.execute(query)
db_connection.commit()
print(db_cursor.rowcount, "Record Inserted")
    context= { 'data':'Thank you for uploading. '+status }
    return render(request, 'Upload.html', context)
```

6. SYSTEM TESTING

The purpose of testing is to discover errors. Testing is the process of trying to discover every conceivable fault or weakness in a work product. It provides a way to check the functionality of components, sub assemblies, assemblies and/or a finished product. It is the process of exercising software with the intent of ensuring that the Software system meets its requirements and user expectations and does not fail in an unacceptable manner. There are various types of test. Each test type addresses a specific testing requirement.

6.1 TYPES OF TESTS

6.1.1 Unit testing:

Unit testing involves the design of test cases that validate that the internal program logic is functioning properly, and that program inputs produce valid outputs. All decision branches and internal code flow should be validated. It is the testing of individual software units of the application. It is done after the completion of an individual unit before integration. This is a structural testing, that relies on knowledge of its construction and is invasive.

6.1.2 Integration testing:

Integration tests are designed to test integrated software components to determine if they actually run as one program. Testing is event driven and is more concerned with the basic outcome of screens or fields. Integration tests demonstrate that although the components were individually satisfactory, as shown by successful unit testing, the combination of components is correct and consistent.

6.1.3 Functional testing:

Functional tests provide systematic demonstrations that functions tested are available as specified by the business and technical requirements, system documentation, and user manuals.

Functional testing is centered on the following items:

Valid Input : identified classes of valid input must be accepted.

- Invalid Input : identified classes of invalid input must be rejected.
Functions : identified functions must be exercised.
Output : identified classes of application outputs must be exercised.
Systems/Procedures : interfacing systems or procedures must be invoked.

Organization and preparation of functional tests is focused on requirements, key functions, or special test cases. In addition, systematic coverage pertaining to identify Business process flows; data fields, predefined processes, and successive processes must be considered for testing. Before functional testing is complete, additional tests are identified and the effective value of current tests is determined.

6.1.4 System Testing:

System testing ensures that the entire integrated software system meets requirements. It tests a configuration to ensure known and predictable results. An example of system testing is the configuration oriented system integration test. System testing is based on process descriptions and flows, emphasizing pre-driven process links and integration points.

6.1.5 White Box Testing:

White Box Testing is a testing in which the software tester has knowledge of the inner workings, structure and language of the software, or at least its purpose. It is used to test areas that cannot be reached from a black box level.

6.1.6 Black Box Testing:

Black Box Testing is testing the software without any knowledge of the inner workings, structure or language of the module being tested. Black box tests, as most other kinds of tests, must be written from a definitive source document, such as specification or requirements document, such as specification or requirements document.

UNIT TESTING

Unit testing is usually conducted as part of a combined code and unit test phase of the software lifecycle, although it is not uncommon for coding and unit testing to be conducted as two distinct phases.

Test strategy and approach:

Field testing will be performed manually and functional tests will be written in detail.

Test objectives:

- All field entries must work properly.
- Pages must be activated from the identified link.
- The entry screen, messages and responses must not be delayed.

Features to be tested

- Verify that the entries are of the correct format
- No duplicate entries should be allowed
- All links should take the user to the correct page.

Integration Testing

Software integration testing is the incremental integration testing of two or more integrated software components on a single platform to produce failures caused by interface defects.

The task of the integration test is to check that components or software applications, e.g. components in a software system or – one step up – software applications at the company level – interact without error.

Test Results: All the test cases mentioned above passed successfully. No defects encountered.

ACCEPTANCE TESTING

User Acceptance Testing is a critical phase of any project and requires significant participation by the end user. It also ensures that the system meets the functional requirements.

Test Results: All the test cases mentioned above passed successfully. No defects encountered.

Input Design:

The input design is the link between the information system and the user. It comprises the developing specification and procedures for data preparation and those steps are necessary to put transaction data in to a usable form for processing can be achieved by inspecting the computer to read data from a written or printed document or it can occur by having people keying the data directly into the system. The design of input focuses on controlling the amount of input required, controlling the errors, avoiding delay, avoiding extra steps and keeping the process simple. The input is designed in such a way so that it provides security and ease of use with retaining the privacy. Input Design considered the following things:

- What data should be given as input?
- How the data should be arranged or coded?
- The dialog to guide the operating personnel in providing input.
- Methods for preparing input validations and steps to follow when error occur.

OBJECTIVES:

1. Input Design is the process of converting a user-oriented description of the input into a computer-based system. This design is important to avoid errors in the data input process and show the correct direction to the management for getting correct information from the computerized system.
2. It is achieved by creating user-friendly screens for the data entry to handle large volume of data. The goal of designing input is to make data entry easier and to be free from errors. The data entry screen is designed in such a way that all the data manipulates can be performed. It also provides record viewing facilities.
3. When the data is entered it will check for its validity. Data can be entered with the help of screens. Appropriate messages are provided as when needed so that the user will not be in maize of instant. Thus the objective of input design is to create an input layout that is easy to follow

Output Design

A quality output is one, which meets the requirements of the end user and presents the information clearly. In any system results of processing are communicated to the users and to other system through outputs. In output design it is determined how the information is to be displaced for immediate need and also the hard copy output. It is the most important and direct source information to the user. Efficient and intelligent output design improves the system's relationship to help user decision-making.

1. Designing computer output should proceed in an organized, well thought out manner; the right output must be developed while ensuring that each output element is designed so that people will find the system can use easily and effectively. When analysis design computer output, they should Identify the specific output that is needed to meet the requirements.

2. Select methods for presenting information.

3. Create document, report, or other formats that contain information produced by the system.

The output form of an information system should accomplish one or more of the following objectives.

- Convey information about past activities, current status or projections of the
- Future.
- Signal important events, opportunities, problems, or warnings.
- Trigger an action.
- Confirm an action.

7. OUTPUT

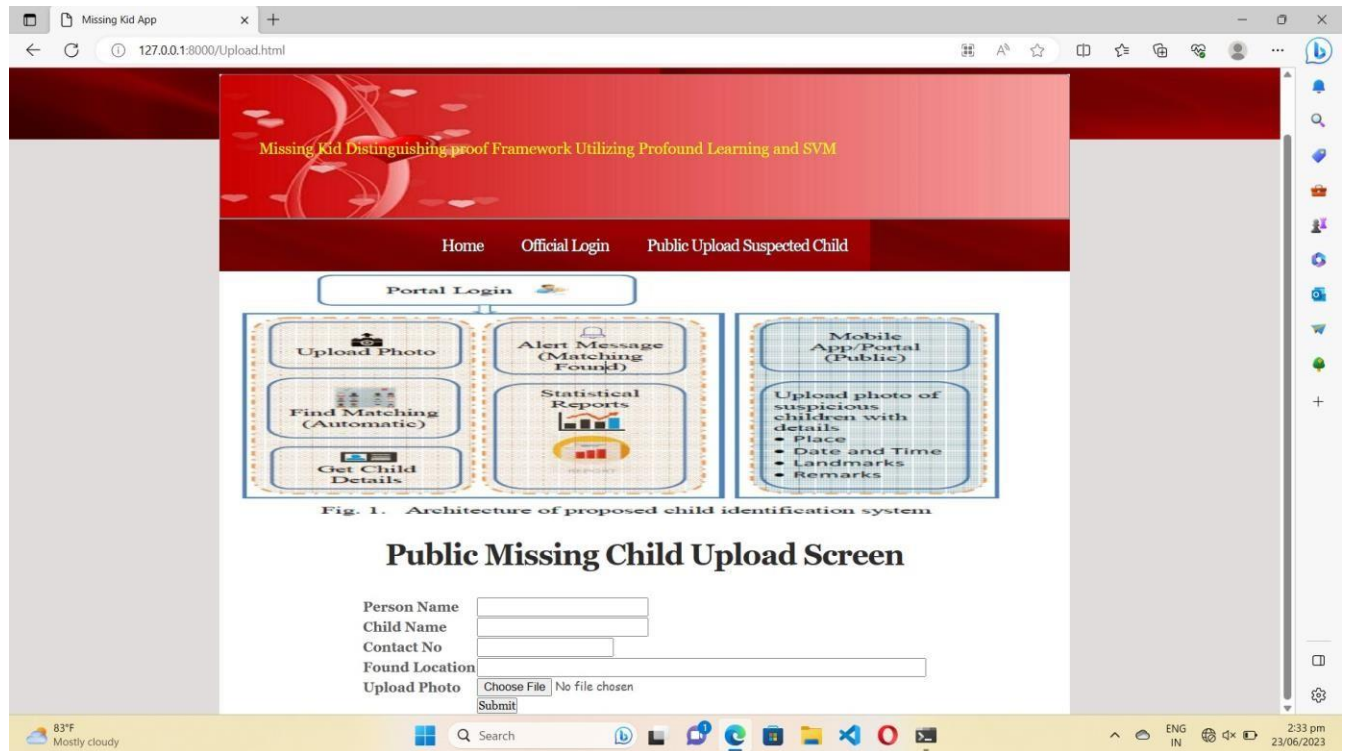


Figure:7.1 Public can click on” Public can upload the suspected child” in this screen

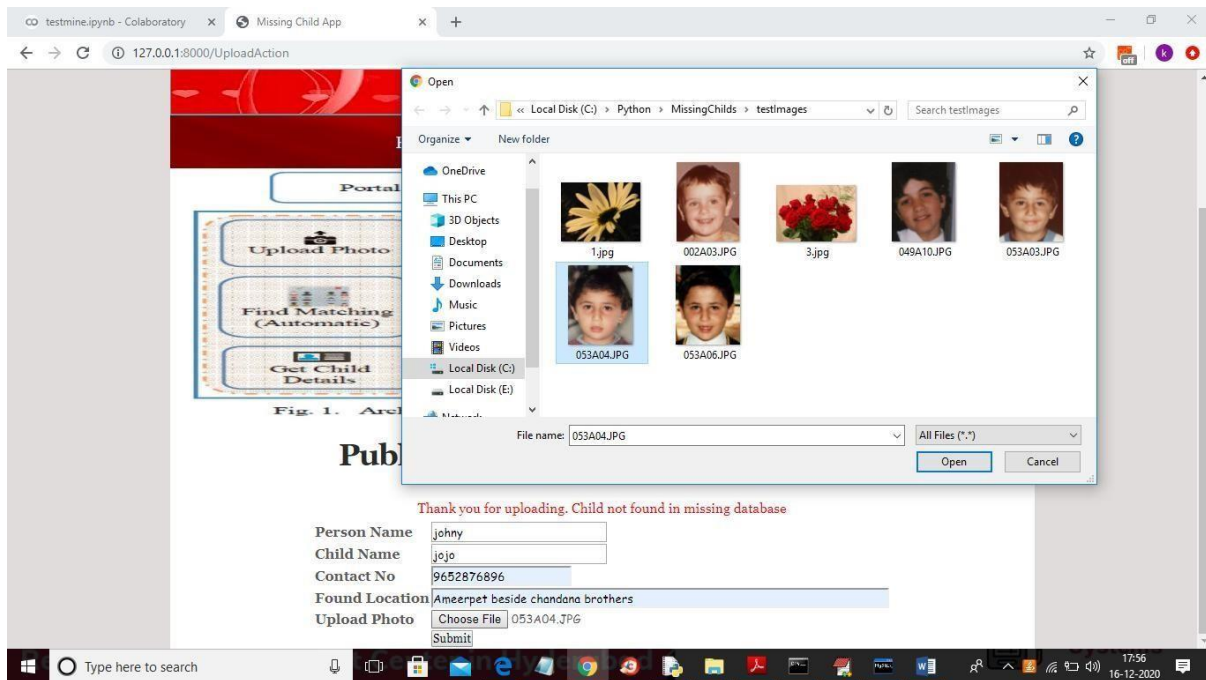


Figure7.2. public can upload the details in this screen.

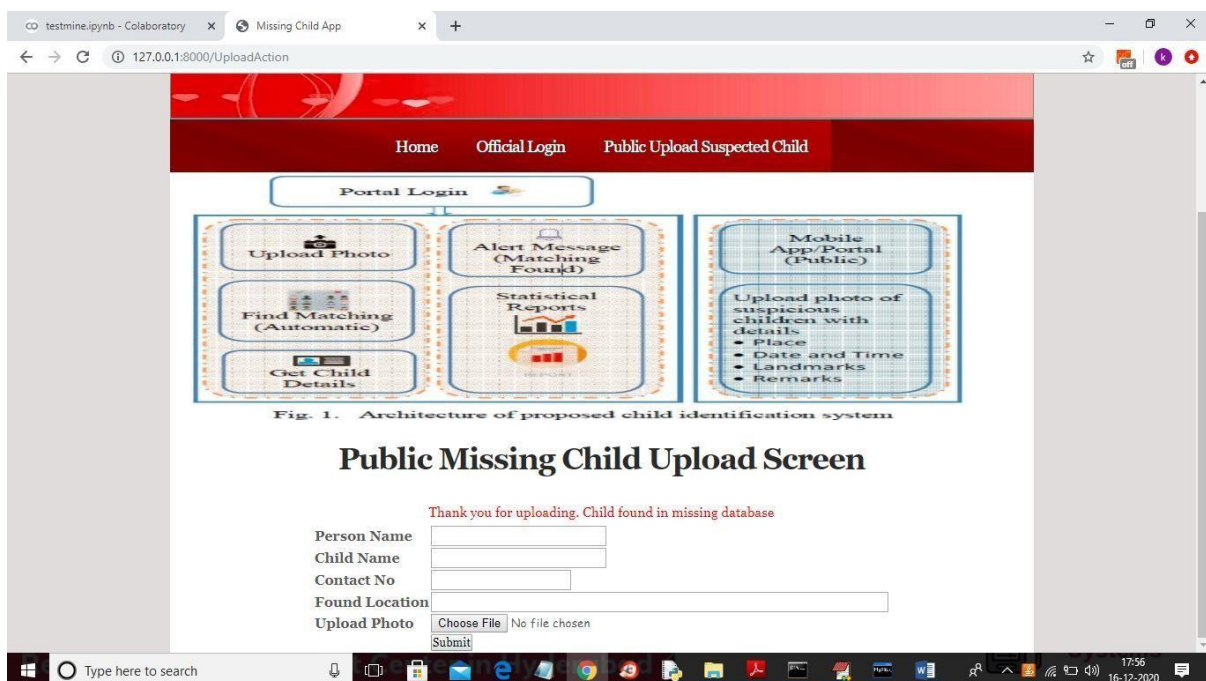


Figure7.3. view public upload missing child status in this screen

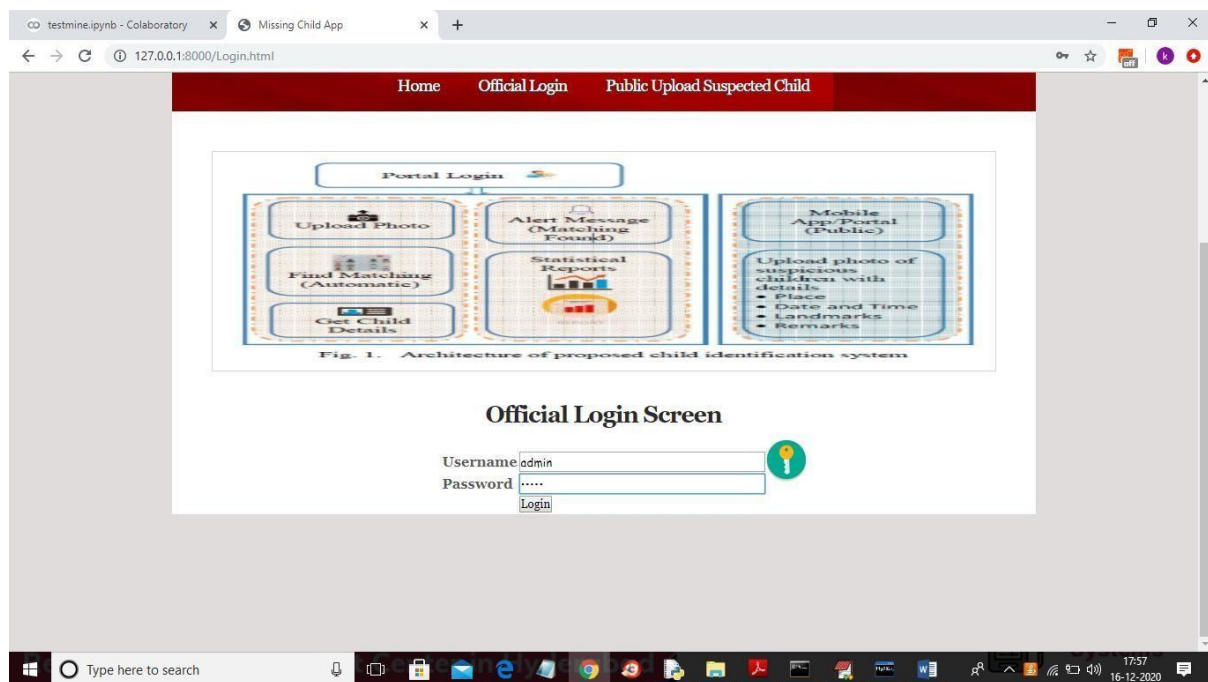





Figure:7.4. Official login screen

Upload Person Name	Child Name	Contact No	Found Location	Child Image	Uploaded Date	Status
rajesh	suresh	9652876896	Ameerpet beside chandana brothers		2020-12-16 17:54:25	Child not found in missing database
john	fredde	1234543212	Ameerpet beside chandana brothers		2020-12-16 17:55:35	Child not found in missing database
johny	jojo	9652876896	Ameerpet beside chandana brothers		2020-12-16 17:56:06	Child found in missing database

7.5. view all the details of the public uploaded photos

8. CONCLUSION

A missing child identification system is proposed, which combines the powerful CNN based deep learning approach for feature extraction and support vector machine classifier for classification of different child categories. This system is evaluated with the deep learning model which is trained with feature representations of children faces. By discarding the softmax of the VGG-Face model and extracting CNN image features to train a multi class SVM, it was possible to achieve superior performance. Performance of the proposed system is tested using the photographs of children with different lighting conditions, noises and also images at different ages of children. The classification achieved a higher accuracy of 99.41% which shows that the proposed methodology of face recognition could be used for reliable missing children identification.

9.**REFERENCES**

1. Y. LeCun, Y. Bengio, and G. Hinton, "Deep learning", *Nature*, 521(7553):436–444, 2015.
2. O. Deniz, G. Bueno, J. Salido, and F. D. la Torre, "Face recognition using histograms of oriented gradients", *Pattern Recognition Letters*, 32(12):1598–1603, 2011.
3. C. Geng and X. Jiang, "Face recognition using sift features", *IEEE International Conference on Image Processing(ICIP)*, 2009.
4. Rohit Satle, Vishnuprasad Poojary, John Abraham, Shilpa Wakode, "Missing child identification using face recognition system", *International Journal of Advanced Engineering and Innovative Technology (IJAEIT)*, Volume 3 Issue 1 July - August 2016.
5. <https://en.wikipedia.org/wiki/FindFace>
6. <https://www.reuters.com/article/us-china-trafficking-apps/mobileapp-helps-china-recover-hundreds-of-missing-childrenidUSKBN15J0GU>
7. Simonyan, Karen and Andrew Zisserman, "Very deep convolutional networks for large-scale image recognition", *International Conference on Learning Representations (ICLR)*, April 2015.
8. O. M. Parkhi, A. Vedaldi, and A. Zisserman, "Deep Face Recognition," in *British Machine Vision Conference*, vol. 1, no. 3, pp. 1-12, 2015.
9. A. Vedaldi, and K. Lenc, "MatConvNet: Convolutional Neural Networks for MATLAB", *ACM International Conference on Multimedia*, Brisbane, October 2016