FINDING MISSING PERSON USING ARTIFICIAL INTELLIGENCE

A. ARIVUNIDHI
Department of Computer Science & Engineering,
SRMIST, Vadapalani,
Chennai, India
ai7710@srmist.edu.in

R. B. BALAJI KAMAL
Department of Computer Science & Engineering,
SRMIST, Vadapalani,
Chennai, India
bk3890@srmist.edu.in

Sangeetha
Department of Computer Science & Engineering,
SRMIST, Vadapalani,
Chennai, India

Abstract--- This paper presents a proposed system to aid in the search for missing persons, including children, teenagers, the mentally challenged, and elderly individuals with Alzheimer's, who often go unaccounted for. The system employs face recognition technology, which offers numerous benefits for locating missing persons. To streamline the process of locating missing individuals, the authors intend to create a user-friendly application that can be accessed by volunteers. This application will enable quick identification and verification of missing individuals' images. Automating this process will help identify if the individual in a photo taken from a particular location matches the missing person's characteristics. If it is a match, the police can take immediate action to locate the person in that area, making their work more efficient.

1. INTRODUCTION

It is a sad reality that numerous individuals go missing each day, whether it is due to old age, mental illness, emotional disorders, Alzheimer's disease, or other causes. Unfortunately, the process of finding these missing individuals often proves difficult, and many of them remain unaccounted for. However, we have proposed a solution to address this issue. Our proposed solution involves keeping a comprehensive record of all newly filed missing person cases in a database accessible through an application. Whenever someone encounters a missing individual, they can take a photo of them and search for their information in the database. In cases where a match is not found, they have the option to upload the individual's information to the database, including their current location if known, and notify higher authorities of the situation. On the other hand, if the search results in a match for the missing individual, the relevant information from the database will be accessed, and the authorities or the family of the missing person will be notified immediately. By using this approach, we aim to streamline the search process, increasing the likelihood of locating missing individuals and ultimately reuniting them with their loved ones.

A. DOMAIN OVERVIEW

ARTIFICIAL INTELLIGENCE

Artificial Intelligence (AI) is a multidisciplinary field that focuses on developing intelligent machines that can perform tasks that typically require human intelligence. AI is a broad domain that encompasses various subfields, including machine learning, natural language processing, robotics, computer vision, and more. Machine learning is a subset of AI that focuses on developing algorithms that enable machines to learn from data and improve their performance over time without explicit programming. Deep learning is a subset of machine learning that involves the use of artificial neural networks with multiple layers to analyze and interpret large amounts of data. It enables machines to learn from unstructured data such as images, audio, and text, and has led to significant breakthroughs in fields such as image and speech recognition, natural language processing, and robotics.

Computer vision is another subfield of AI that focuses on teaching machines to interpret and analyze visual data, such as images and videos. AI is also used in various industries, including healthcare, finance, manufacturing, transportation, and more. In healthcare, AI is used for diagnosing diseases, drug discovery, and personalized treatment plans. In finance, AI is used for fraud detection, credit scoring, and investment management. In manufacturing, AI is used for predictive maintenance, quality control, and supply chain management. In transportation, AI is used for autonomous vehicles and traffic optimization. Overall, AI is a rapidly evolving domain that has the potential to revolutionize the way we live and work. As technology advances, AI is expected to play an increasingly critical role in various industries, making our lives more efficient, convenient, and safer.

B. Dataset preparation:

The dataset consists of around 1300 training images and 180 testing images that were pre-processed to extract features and then categorized into three different classes.

1) Covid19
2) Normal
3) Pneumonia

C. PROPOSED SYSTEM

The Image Similarity API measures the visual similarity between two images and generates a score that indicates their degree of similarity, with a score of '0' representing an exact match. By using this API, one can avoid the tedious task of manually searching for duplicates or visually comparable images in datasets. The API also returns a distance value that can be used to track how photographs evolve over time or detect duplicates in user data. With this information, users can group similar images together, search for duplicates, or integrate image similarity into their applications. In another application, the Sentence Similarity API can be used to search for a missing person by prompting the user to provide a photo that is then compared to images in the database using filters such as name, age, and location.

D. *EXISTING SYSTEM*

The current method of locating missing persons involves the use of facial recognition technology through a system that includes both a user dashboard and an admin dashboard. This system uses a software that employs machine learning algorithms to identify missing individuals by analyzing facial features and comparing them to a database of images. The user dashboard allows individuals to submit information about missing persons, including photographs and relevant details. The admin dashboard provides authorized personnel with access to the centralized database to search for missing persons based on various criteria such as location, age, and gender. This system represents a significant advancement in technology for locating missing persons, although it has some limitations. Nonetheless, it has proven to be a valuable tool in the search for missing individuals, and as technology continues to advance, we can expect further improvements in this field.

E. *COMPARISON OF EXISTING AND PROPOSED SYSTEM*

The existing system for finding missing persons using face matching algorithm with user and admin dashboard and the proposed system using image similarity API share some similarities in terms of the use of advanced technologies to assist in locating missing individuals. Both systems rely on machine learning algorithms to analyze images and identify individuals based on their visual features. They also provide a user interface for individuals to search for missing persons, with the existing system using a centralized database and the proposed system utilizing filters and image lookup. However, there are some significant differences between the two systems. The existing system is specifically designed for finding missing individuals using facial recognition, while the proposed system is more focused on identifying visually similar images and finding duplicates in a collection. The existing system is more comprehensive, with a user dashboard for submitting information and an admin dashboard for authorized personnel to access the centralized database. In contrast, the proposed system is more geared towards developers and incorporates image similarity into apps. Another difference between the two systems is the level of specificity in searching for missing individuals. The existing system allows for filtering by various criteria such as location, age, and gender, while the proposed system only allows for image lookup and filtering by name, age, and location. In conclusion, while both systems utilize advanced technologies for searching and identifying missing individuals, the existing system is more comprehensive and specific in its search criteria, while the proposed system is more geared towards developers and finding visually similar images.

2. LITERATURE REVIEW

Title: Pneumonia Detection using Deep Learning.
Author:Shreyas Mishra; Aniket Hazra; U.M. Prakash
Year: 2022
METHODOLOGY:
To determine the likelihood of functional connection between gene pairs based on previous information, it first employs the Nave Bayesian classifier. We used cogitation from PubMed and schematic similarity from Gene Ontology annotation in this investigation.
PERFORMANCE ANALYSIS:
The effectiveness of the BN modelling of gene expression data is much enhanced by this invention, which offers a statistical method to use the quantitative information in prior biological knowledge.
LIMITATIONS:
The reading was manually entered, and this is not an user friendly application.

Title: Reliability of Machine Learning in Eliminating Data Redundancy of Radiomics and Reflecting Pathophysiology in COVID-19 Pneumonia: Impact of CT Reconstruction Kernels on Accuracy
Author: Yauhen Statsenko; Tetiana Habuza; Tatsiana Talako; Tetiana Kurbatova; Gillian Lylian Simiyu
Year: 2021
METHODOLOGY:
1)Acquiring Hematological, Biochemical and Functional Data
2) CT Scanning
3) Image Pre-processing
PERFORMANCE ANALYSIS:
This tool allows for the assessment of lung injury in COVID-19 patients in relation to their demographics, clinical characteristics, and laboratory findings.
LIMITATIONS:
The models we developed were not trained to operate with the data usual for the middle and late periods of the illness since the radiological results differ between the disease phases. Further investigations are necessary to extend the value of the ML algorithms by applying them to the findings of the follow-up studies. The fact that we conducted extensive SARS-CoV-2 testing on patients is another drawback of the current investigation. Nonetheless, coinfections are possible and should be evaluated.

Title: On the Early Detection of COVID19 using Advanced Machine Learning Techniques: A Review
Author: Mohammed AbdulAzeem Siddiqui; Mohammed Akber Ali; Mohamed Deriche
Year: 2021
METHODOLOGY:
Pre-processing
Classification
Training
PERFORMANCE ANALYSIS:
The authors presented a DL system in which CNN, pooling, activation, and batch normalization served as the system's primary computing units [12]. In order to discover the relationships between the chest CT picture and the lung functional aberrations, our system first learnt hierarchical lung characteristics from a dataset.
LIMITATIONS:
The difficulties scientists are encountering in trying to discover COVID19, as well as the many tools and algorithms that have great potential for helping them.

Title: COVID-19 detection from Xray and CT scans using transfer learning
Author: Mohamed Berrimi; Skander Hamdi; Raoudha Yahia Cherif; Abdelouahab Moussaoui; Mourad Oussalah; Mafaza Chabane
Year: 2021
MEDHODOLOGY:
Pre-processing
Classification
AUGMENTATION
Training
PERFORMANCE ANALYSIS:
The learning was much improved when the Convolution layer was introduced to the top of the DenseNet model and the regularisation terms were added to the bottom of the model.
LIMITATIONS:
The feature maps of each layer are combined with the layer before, which is a drawback of DenseNet because it duplicates the data.

Title: Deep learning for mycoplasma pneumonia discrimination from pneumonias like COVID-19
Author: Ali Serener, Sertan Serte
METHODOLOGY:
Pre-processing
Classification
PERFORMANCE ANALYSIS:
The least effective designs include DenseNet-121, SqueezeNet, VGG, and AlexNet. The DenseNet121 architecture, with an AUC of 0.68, accuracy of 0.67, sensitivity of 1.00, and specificity of 0.55, is the least effective of the seven.
LIMITATIONS:
The feature maps of each layer are combined with the layer before, which is a drawback of DenseNet because it duplicates the data.

Title: A Cough-based deep learning framework for detecting COVID-19
Author: Truong Hoang; Lam Pham; Dat Ngo; Hoang D. Nguyen
Year: 2022
METHODOLOGY:
SVM-based SMOTE technique, a variation of SMOTE algorithm that employs SVM algorithm to find neighbor samples, is implemented on the combined features to ensure the equal number of positive and negative samples, in order to address the issue of unbalanced dataset discussed in Part II.
PERFORMANCE ANALYSIS:
The TRILL-based embedding performs better than the other single features, with average AUC scores on the Development set and Blind Test set of 73.77 and 80.57, respectively.
LIMITATIONS:
SMOTE disregards the possibility that nearby instances may belong to different classes. As a result, there may be more class overlap, which will add to the noise. With high dimensional data, SMOTE is not particularly useful.

Title: Trend Prediction of Influenza and the Associated Pneumonia in Taiwan Using
Machine Learning
Author: Ting-Chien Weng, Mei-Juan Chen
Year: 2019
METHDOLOGY:
Two deep learning hidden layers with 123 and 100 nodes each are present in the proposed MLP model. Tensorflow software implements the MLP, and the training method iteration number is 2000 to identify the best results.
PERFORMANCE ANALYSIS:
For the trend of influenza and the accompanying pneumonia volumes for the entire population, the suggested technique utilising MLP machine learning achieves 77.54% prediction accuracy. The older population is best represented by the national trend projection (81.16%).
LIMITATIONS:
Considering that MLP is completely integrated, it has much too many parameters. Redundancy and inefficiency occur from the extremely thick network of connections between each node.

Title: Dual-Sampling Attention Network for Diagnosis of COVID-19 from Community Acquired Pneumonia
Author: Xi Ouyang , Jiayu Huo , Liming Xia , Fei Shan
Year: 2020
METHODOLOGY:
All CT images are segmented into the lungs using a well-known VB-Net toolkit, which also does auto-contouring of areas that might be infected. In order to condense and merge feature map channels, the VB-Net toolkit modifies the V-Net network by adding bottleneck layers.
PERFORMANCE ANALYSIS:
Fivefold cross-validation was used to verify the learned algorithms. Five performance indicators, including accuracy, sensitivity or recall, specificity, precision (PPV), and F1 score, were used to compare the effectiveness of various networks.
LIMITATIONS:
Regarding specificity, the dual-sampling technique performs somewhat worse than ResNet34 with uniform sampling.

Title: Automatic Detection and Diagnosis of Severe Viral Pneumonia CT Images Based on LDAS
Author: Gengfei Ling1, Congcong Cao2
Year: 2019
METHODOLOGY:
Kaao helps to identify different lung types using the support vector machine technique to get side effects.Futher more, Adaboost algorithm the influence of strong classifier by merging weak classifiers. An iterative method is used.
PERFORMANCE ANALYSIS:
The models created for the three types of features using the imbalanced GWCMCx dataset are provided along with their performance results1. Models created utilising KNN and a histon-based superpixel characterization produced the greatest results.
LIMITATIONS:
Large data sets are not a good fit for the SVM algorithm. When the target classes are overlapping and the data set includes more noise, SVM does not perform very well.

Title: AI help in screening Viral and COVID-19 pneumonia
Author: Muhammad E. H. Chowdhury1 , Tawsifur Rahman
Year: 2021
METHODOLOGY:
The pre-trained networks' Softmax and classification layer are modified using the trained model from a sizable dataset like ImageNet under the transfer learning idea. The network is then trained more quickly using the previously learned weights for an application with a relatively small dataset.
PERFORMANCE ANALYSIS:
The developed algorithms were verified using 5-fold cross-validation. Five performance indicators, including accuracy, sensitivity or recall, specificity, precision (PPV), and F1 score, were used to compare the effectiveness of various networks.
LIMITATIONS:
The primary clinical approach for diagnosing COVID-19 is currently the Reverse transcription polymerase chain reaction (RT-PCR), which is expensive, less accurate, and requires specialized medical personnel. X-ray imaging could be a useful alternative for COVID-19 diagnosis, as it is widely available.

3. SYSTEM ARCHITECTURE AND DESIGN

This graphic provides a concise and understandable description of all the entities currently integrated into the system. The diagram shows how the many actions and choices are linked together. You might say that the whole process and how it was carried out is a picture. The figure below shows the functional connections between various entities. To illustrate the movement of information throughout a procedure or system, one might use a Data-Flow Diagram (DFD). A data-flow diagram does not include any decision rules or loops, as the flow of information is entirely one-way. A flowchart can be used to illustrate the steps used to accomplish a certain data-driven task. Several different notations exist for representing data-flow graphs. Each data flow must have a process that acts as either the source or the target of the information exchange. Rather than utilizing a data-flow diagram, users of UML often substitute an activity diagram. In the realm of data-flow plans, site-oriented data-flow plans are a subset. Identical nodes in a data-flow diagram and a Petri net can be thought of as inverted counterparts since the semantics of data memory are represented by the locations in the network. Structured data modelling (DFM) includes processes, flows, storage, and terminators.

A. FEASIBILITY STUDY

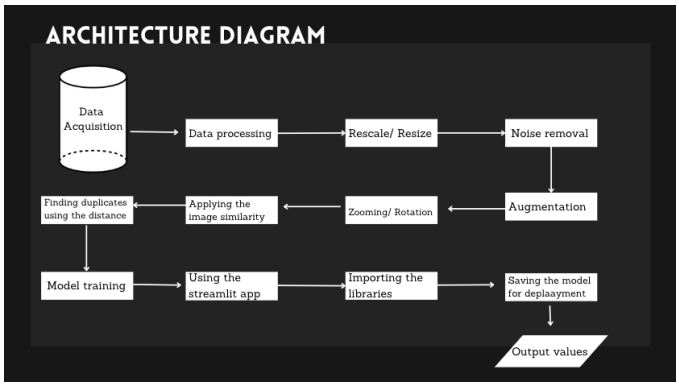
A business proposal detailing the project's main objectives and providing some rough cost estimates is provided here with the aim of evaluating the project's viability and enhancing server performance. After a thorough investigation has been completed, the viability of your suggested system may be evaluated. Before starting the feasibility study, it is crucial to have a firm grasp of the fundamental requirements of the system in question. The four main lines of thought in the feasibility investigation are as follows:

- Economical feasibility
- Technical feasibility
- Operational feasibility
- Social feasibility

ECONOMICAL FEASIBILITY: Upper management might use the study's findings to evaluate the possible cost savings from utilising this technology. Before running out of money, the company can only invest so much in building and studying the system. Every dollar that is spent needs to be justified. The cost of the updated infrastructure was far lower than expected because the majority of the employed technologies are open-source and free. It was extremely important to only purchase customisable items.

TECHNICAL FEASIBILITY: In order to assure the system's smooth evolution, this research tries to determine its technological viability. The IT team shouldn't be under too much stress as a result of adding more systems. As a result, the buyer will feel needless tension. It is essential that the system be as straightforward in its design as feasible because it is unlikely that any alterations will need to be made during installation.

OPERATIONAL FEASIBILITY: Hearing from those who have actually utilised this technology is a crucial component of our research. The procedure comprises guiding the user towards making the best possible use of the available resource. The system shouldn't make the user feel threatened; rather, they should view it as a necessary evil. How quickly new users embrace a system is directly impacted by the training and orientation they receive. Users must

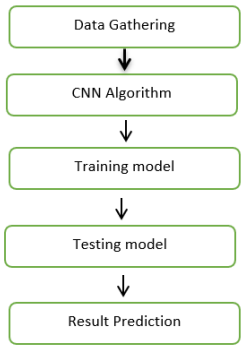


have more confidence in the system before they can offer helpful criticism.

SOCIAL FEASIBILITY: In the social feasibility analysis, we consider how the project might alter the neighbourhood. This is done to determine how much the general public is interested in the project. Due to institutional structures and ingrained cultural norms, it's conceivable that some workers will be hard to find or even nonexistent.

B. CONSTRUCTION OF A DETECTING MODEL

To achieve accurate predictions, deep learning requires a significant amount of historical image data to be collected for training and testing the model.



STEPS OF DATAFLOW DIAGRAM

C. PREPROCESSING

To avoid inconsistency from missing values from the collected data, data are preprocessed for efficiency improvement over the algorithm and get the best result. Removal of outliers and conversions of variable are also required to be completed.

D. PREDICTIVE MODEL CONSTRUCTION

Historical and raw data are gathered first when it comes to the process of machine learning. Then the data should be pre-processed as the raw data cannot be used directly without pre processing as input values. Thirdly the suitable algorithm for the particular problem in machine learning is figured out and applied to the model made. After splitting the model into testing and training the output is predicted and compared with the actual output and errors are also minimized along the process. Tuned model is made and brought into application along with tuned time in order to make sure the accuracy is better and high. Thus, upon following all these steps the prediction is made by the machine**.**

E. ENVIRONMENTAL REQUIREMENTS

HARDWARE REQUIREMENTS

Processor	: Pentium Dual Core 2.00GHZ
Hard disk	: 120 GB
RAM	: 2GB (minimum)
Keyboard	: 110 keys enhanced

SOFTWARE REQUIREMENTS

Operating system	: Windows7 (with service pack 1), 8, 8.1 and 10
Language	: Python

F. SYSTEM ARCHITECTURE

Python packages used:

1. Pycharm- With PyCharm, you can improve your coding speed and efficiency thanks to its intelligent and customizable editor which will support code folding, code completion, snippets, and split windows. The tools in PyCharm are tightly integrated to provide an efficient environment for Python development.
2. NumPy- Numpy is a module in Python that focuses on mathematical functions and numerical calculations. It provides the ability to read and manipulate data in arrays.
3. Pandas- Pandas is a module that allows for reading and writing of files. It also provides data frames to effectively manipulate data.
4. Matplotlib- Matplotlib can be used in combination with data frames to visualize and identify patterns in the data being used.

6. MODULE DESCRIPTION

MODULE 1: DATA ACQUISITION AND PRE-PROCESSING

Data Acquisition

To train our programme, we will use a deep learning dataset with sets of comparable photos. Many photos in the collection may or may not be comparable to one another. By the creation of an image similarity API, we will use it to determine whether two images are similar.

Data Cleaning - Preprocessing of images

We pre-processed the dataset of photos we had gathered after the photo collection. The following a variety of pre-processing procedures were applied to the dataset's photographs.

Image

A two-dimensional array that is precisely set up in rows and columns can be used to define a picture. A digital image is made up of an infinite number of discrete elements, each of which has a unique value at a unique place. These parts are referred to as pixels.

Types of Image:

- *Binary Image*
- *Black and White Image*
- *8-bit color format*
- *16-bit color format*

Zooming

Image zooming means converting the image into a magnified image. The zoomed
version of thermal images can be used for augmentation as they represent the case
when images are taken from a close distance.

Rotation

Rotation of an image means changing the position of an object about a pivot point at some angle. Image would represent the case as if the picture was taken from a different angle.

Salt and pepper noise

Salt and pepper noise is added to an image by randomly changing intensities of some of the pixels of an image to 1 and some other to 0. Salt and pepper noise can represent a case where images were collected on a dusty day, or when the camera has dust on it.
A few pixels are very noisy. This is known as salt and pepper noise, and it applies to a wide range of mechanisms that cause this basic

image deterioration. The result is akin to scattering white and black dots across the picture.
Cleaning unwanted images. The dataset had a lot of unwanted data so we need to remove those images from our dataset. Now that we′ve chosen our parameters for image tagging, we can iterate through each of the folders, finding artificial images and removing them using Path.unlink().
This enables us to threshold these photographs at a different level than those without text depending on the colour proportion. It makes sense that photographs with incorrectly recognised text will have a much more varied tonal distribution than those with text added (which typically have a singular font color).

Resizing the images

Initially we had 1920x1080 size of images but for reducing computation we have to reduce our images to a size of 128x128 pixels.

Color Channel changing

We shall modify the colour channel after scaling in order to extract the features.
Enhancing the contrast in an image allows it to have more contrast and more intensities overall than it would without. Since the same photograph can have stronger contrast on a sunny or bright day, this was used to enhance images. Our training set was chosen to include a wide variety of potential outcomes.

Labeling out images

Now we need to label the images as per real or fake in training dataset. The first and most important step is labelling the image. Although labelling takes a lot of time, the more effort we put into labelling the photographs, the more accurate our model will be.

5.2 MODULE 2: MODEL IMPROVISATION

Deep ai's image similarity model will be applied. When two photographs are compared, Image Similarity gives us a result that indicates how visually similar the two images are.
With a score of '0' meaning that the two photos are identical, the lower the value, the more contextually similar the two images are. Letting machine vision do it for you using this API will save you from having to sift through datasets looking for duplicates or identifying a visually comparable set of images.
The image similarity API analyses two photos and produces a distance between the two images. The distance value tells us how visually similar the two photographs are, with a distance value of 0 representing an exact match. With the help of the distance value, we can determine how two photographs evolve over time or find duplicates in your user data.
An indicator of how visually similar two photographs are is returned by the API. With this, you can group similar images together, search for duplicates in a collection, or incorporate image similarity into your apps.
We can use the sentence similarity API to:
Lookup using an image In this scenario, the user is prompted to provide a picture of the missing person so that the database can be searched. This has two applications. The user will first see information on the missing individual, such as name, age, contact information, and location, if the record in the database matches.
Search by filter: Users can quickly search for records by using the following filters in
addition to the two options listed above.
Filter by name: When a user enters a name, the appropriate information is taken from the database.
Filter by age: If the stranger uploading the case does not know the exact age, a slider is provided to select a range of ages. The details of those belonging to the chosen age group will be shown.
Filter by location: User will be prompted to enter the state to receive the relevant
information when using the location filter.

Model Training

Next is to train the model using GPU. We will now use batches of data to train our model. It is preferable that images derived from the same movie are included in the same data set because images from the same film may be somewhat comparable. The likelihood that the model would accurately predict more images from a given images will increase if it was trained with images. The goal is to find similar elements that are present in numerous images, though. As a result, the image ID was used to divide datasets rather than the picture ID. Images ranged in duration as well. As a result, there were differing numbers of photos in the training data set and test data set.

5.3 MODULE 3: CREATING ADMIN UI

We will now use Streamlit to develop the admin ui. Without using HTML, CSS, or JS, we can create UIs for our models with the aid of the Python module Streamlit. The majority of models are unattractive and die inside a Jupyter notebook. But using Streamlit, we can design a simple user interface to display your model to others.
Steps:
1. Install the necessary libraries, including Matplotlib, Scikit-Learn, Pandas, and Streamlit.
2. import the required libraries, including confusion matrix, train test split, and logistic regression.
3. Data frame loading
4. data manipulation
5. split between the two
6. Create the model.
7. Start the streamlit app by typing streamlit run app.py.

Getting MongoDB database

Since mongoDB will be our database, connecting to the MongoDB cluster will be the initial step. A document database called MongoDB is used to create scalable and highly accessible web applications. When your data is document-centric and doesn't fit well into the schema of a relational database, when you need to support vast size, when you need to quickly prototype, and for a few additional use cases, NoSQL databases like MongoDB are a solid option.
Due to its flexible schema approach and popularity among agile development teams, we have implemented it.
We need the PyMongo driver to connect. Then follow the steps:
1. Click connect.
a. Click Databases in the top-left corner of Atlas.
b. In the Database Deployments view, click Connect for the database deployment to which you want to connect.
2. Select a connection method by clicking.
3. Connect your application by clicking.
4. Choose your driver version and Python.
5. The connection string is copied.
6. Set up the connection string that is provided.
7. import PyMongo's MongoClient.
8. To your cluster, connect.

Creating Missing Person Registration Function

Now we will create a missing registration function to register about the missing personby providing his/her name, contact and image. This function will add details about a missing person, by adding name, image_path, and number. When we add the details, the function will print “Report added.” The function is also capable of displaying in case the report is not added due to error. It will print “Error happened while uploading missing person’s data to cloud.”

Processing Input

Now we will create a function that will take that input data and process it and save it to the mongoDB database.
1. AWS will by default be the chosen Cloud Provider & Region. You can now
construct and use an online database by clicking on Create Cluster.
2. Now, in the dialogue box shown in the bottom-left corner, select Create your first database user.
3. The dialogue box will close once you enter the new database user's name and
password and click the Add User button.
4. In the Data Storage section, click the Network Access hyperlink. You'll view the page after that.
5. Simply select Allow Access From Anywhere, and then select Confirm. Now click the Clusters hyperlink, and the following page will appear.

6. When you click on the Connect your application section, the following dialogue box will appear.
7. Finally, select the correct driver and version, copy the link, and then paste it into your code. password> should be replaced with the password you used to create a new database user, and dbname> should be changed to the name of the
database (Remember Database, not Database user).
8. Using MongoDB Atlas, here is how we developed our own storage area for our
programme.

Providing data to check the similarity

Now if we get the person info from the phone number we will provide an image of the person we got from the user panel and we will get whether the 2 images are of same person or not.

5.3 MODULE 3: CREATING USER UI

Now we will create an user UI using Streamlit. We will now use Streamlit to develop the admin ui. Without using HTML, CSS, or JS, we can create UIs for our models with the aid of the Python module Streamlit. The majority of models are unattractive and die inside a Jupyter notebook. But using Streamlit, we can design a simple user interface to display your model to others.

Steps:

1. Install the necessary libraries, including Matplotlib, Scikit-Learn, Pandas, and
Streamlit.
2. import the required libraries, including confusion matrix, train test split, and
logistic regression.
3. Data frame loading
4. data manipulation
5. split between the two
6. Create the model.
7. Start the streamlit app by typing streamlit run app.py.

Connecting to mongoDB database

At first users have to be connected to the mongoDB database.
We need the PyMongo driver to connect. Then follow the steps:
1. Click connect.
a. Click Databases in the top-left corner of Atlas.
b. In the Database Deployments view, click Connect for the database deployment to which you want to connect.
2. Select a connection method by clicking.
3. Connect your application by clicking.
4. Choose your driver version and Python.
5. The connection string is copied.
6. Set up the connection string that is provided.
7. import PyMongo's MongoClient.
8. To your cluster, connect.

Querying for person from the Database

We create a function that will check if the missing person is already in the database using phone number. The section is same the section mentioned in module 3, but the only difference is that we request for the data from the user page not the admin page.

Providing data to check the similarity

Now if we get the person info from the phone number we will provide an image of the person we got from the user panel and we will get whether the 2 images are of same person or not.

7. DISCUSSIONS AND CONCLUSION

The "Finding the Missing Person Using Face Match Making Algorithm with User and Admin Dashboard" initiative is a highly innovative and practical solution to locate missing individuals and improve public safety. This project uses advanced technologies such as facial recognition and machine learning algorithms to enhance the accuracy and speed of the search process. By continuously updating the database of known individuals, the system can compare images of missing persons with potential matches, thereby increasing the likelihood of locating them. One of the primary advantages of this project is its user-friendly interface and admin dashboard, which allows users to submit images of missing persons with ease and enables administrators to manage the system efficiently. Additionally, the project's success will be measured by its ability to locate missing persons and reunite them with their loved ones, which will serve as a benchmark for assessing its effectiveness. Moreover, the "Finding the Missing Person Using Face Match Making Algorithm with User and Admin Dashboard" project has the potential to revolutionize the way missing persons are located, which represents a significant step forward in the use of technology for public safety. This project is a valuable contribution to society as it has the potential to save lives and bring peace of mind to families affected by missing persons. In conclusion, the "Finding the Missing Person Using Face Match Making Algorithm with User and Admin Dashboard" project is a crucial tool for locating missing individuals and enhancing public safety. This project's advanced technology and user-friendly interface make it a valuable contribution to society, and its potential to revolutionize the way missing persons are located is remarkable. Therefore, the project represents an essential step forward in the use of technology for public safety and has the potential to save lives and reunite families.

8. REFERENCES

[1] M. Pajewski, C. Kulkarni, N. Daga and R. Rijhwani, "Predicting Survivability in Lost Person Cases," 2021 Systems and Information Engineering Design Symposium (SIEDS), 2021

[2] P. D. N. H. Sai, V. V. S. Kiran, K. Rohith and D. RajeswaraRao, "Identification of Missing Person Using Convolutional Neural Networks," 2022 International Conference on Sustainable Computing and Data Communication Systems (ICSCDS), 2020

[3] H. Yanagimoto, K. Hashimoto and T. Matsuo, "Visualization Based on Person Move Similarity for Person Behavior Analysis," 2019 8th International Congress on Advanced Applied Informatics (IIAI-AAI), 2019

[4] A. Nadeem, K. Rizwan, A. Mehmood, N. Qadeer, F. Noor and A. AlZahrani, "A Smart City Application Design for Efficiently Tracking Missing Person in Large Gatherings in Madinah Using Emerging IoT Technologies," 2021 Mohammad Ali Jinnah University International Conference on Computing (MAJICC), 2021

[5] K. M. A. Solaiman, T. Sun, A. Nesen, B. Bhargava and M. Stonebraker, "Applying Machine Learning and Data Fusion to the ″Missing Person" Problem," in Computer, vol. 55, no. 6, pp. 40-55, June 2022

[6] M. K. Singh, P. Verma, A. S. Singh and A. K, "Implementation of Machine Learning and KNN Algorithm for Finding Missing Person," 2022 2nd International Conference on Advance Computing and Innovative Technologies in Engineering (ICACITE), 2022

[7] Ž. Marušić, D. Zelenika, T. Marušić and S. Gotovac, "Visual Search on Aerial Imagery as Support for Finding Lost Persons," 2019 8th Mediterranean Conference on Embedded Computing (MECO), 2019

[8] D. Hanna and A. Ferworn, "A UAV-based Algorithm to Assist Ground SAR Teams in Finding Lost Persons Living with Dementia," 2020 IEEE/ION Position, Location and Navigation Symposium (PLANS), 2020

[9] K. M. A. Solaiman, T. Sun, A. Nesen, B. Bhargava and M. Stonebraker, "Applying Machine Learning and Data Fusion to the ″Missing Person" Problem," in Computer, vol. 55, no. 6, pp. 40-55, June 2022

[10] S. Alagarsamy, K. V. Sudheer Kumar, P. Vamsi, D. Bhargava and B. D. Hemanth, "Identifying the Missing People using Deep Learning Method," 2022 7th International Conference on