REAL TIME SURVEILLANCE USING DEEPLEARNING

Mr. DHANUSH MUNAGALA[1], Miss. A NARMADA[2]

UG Scholar[1], Assistant professor[2]

Branch of Electronics and Communication Engineering,Matrusri engineering college, Hyderabad, Telangana, India.E-Mail: munagaladhanush4036@gmail.com[1],narmada_mecs@matrusri.edu.in[2].

ABSTRACT

Basic capabilities like video recording and archiving are all that traditional video surveillance offers. It cannot distinguish between approved and unauthorised faces. A vital component of contemporary cities' security and defence systems is video surveillance, thanks to the rapid growth of information technology. Particularly prevalent and important in the current world are surveillance cameras, which are almost everywhere. But as the monitoring network continues to grow, more and more people are using surveillance cameras, which not only make life easier but also produce a huge quantity of data that is difficult to store, analyse, and retrieve. Both authorised and illegal people may be seen by an intelligent surveillance system using intelligent video analysis technologies. In this research, we can distinguish between approved and unauthorised persons by applying deep learning algorithm approaches to identify people filmed using camera module. The individual in the video is recognised and recorded using convolutional neural networks (CNN), and the security team will instantly get an email notice with a picture of the unauthorised person if they enter prohibited places.

Keywords: Surveillance, DeepLearning,FaceRecognition,CNN,Monitoring.

1. INTRODUCTION

1.1 Introduction

The technique of keeping an eye on what's going on in any interior or outdoor place is known as video surveillance. Many businesses have put in place video surveillance systems for a variety of reasons, including staff monitoring and security management. The administrator can keep an eye on everything that happens on the premises and everyone's actions by designing video surveillance systems. The market for video surveillance is expanding quickly. The field of video surveillance, sometimes referred to as CCTV (Closed Circuit Television), has been around for more than 30 years and has seen a number of technical advancements. In the current world, video surveillance has become an essential component of guaranteeing public safety. With the aid of technology, everyday living is becoming more effective and convenient. A vital component of contemporary cities' security and defence systems is video surveillance, thanks to the rapid growth of information technology. But as the monitoring network continues to grow, more and more people are using surveillance cameras, which not only make life easier but also produce a huge quantity of data that is difficult to store, analyse, and retrieve.

Image, video, and multimedia material have significantly increased recently and continue to rise daily. This is a nearly hard assignment since cops continuously watch security recordings that have been recorded and kept for a while to determine whether the person entering the workplace is who they think they are. Personnel and their ongoing focus. Manual monitoring takes a lot of effort and time. Video surveillance is considered to be the greatest alternative for observation and surveillance as a result. We can quickly determine if the individual is permitted or not by utilising this monitoring system. The quantity of cameras used in the surveillance procedure determines how difficult the task is. We can automate this identification process with the use of deep learning-based video analytics. A potent picture categorization technique is deep learning. We don't have enough time in our hectic lives to keep track of everything. Every family has a lot of responsibilities, and hospitals and shopping centres can't listen in on every discussion. We aim to design more intelligently in the twenty-first century to make our lives safer and simpler.

1.2 Problem Statement

This is so because basic features like video recording and storage are all that conventional video surveillance offers. It cannot distinguish between approved and unauthorised faces. To get around this, we provide a deep learning system to track unlawful entrance and report the unauthorised person's picture to the appropriate owner.

1.3 Objectives

Our project's primary goal is to create a surveillance system that utilises deep learning algorithms to detect whether a person is permitted or not, and send a picture of the unauthorised person to the owner of the area if access to that area is limited, along with a warning letter.

2. LITERATURE SURVEY

To build this project, several studies have been undertaken by numerous researchers. They use distinct technology and serve a different purpose, however. The technology and applications of a few of these papers are highlighted here.

[1] M A Imran, M S U Miah, H Rahman, "Face Recognition using Eigenfaces" International Journal of Computer Applications (0975 – 8887) 2015.

This study investigates a "performance-based" real-time face recognition and identification system. The Viola-Jones method is used to identify objects. To find faces, the PCA-based method Eigen Faces is used. They used training data to instantly identify the face. Each person was represented by five photos for the data training set, with the eigenvalues adjusted to match the known individual. The most challenging part of face identification using eigenface is obtaining 80% accuracy and changes in features like backdrop, lighting, facial expression, stance, scale sensitivity, etc.

[2] SenthamizhSelvi.R, D.Sivakumar, Sandhya.J.S, "Face Recognition Using Haar - Cascade Classifier for Criminal Identification", 2019 International Journal of Recent Technology and Engineering (IJRTE)

Face recognition may be done in two phases in real time, for example, B. Face Recognition and Face Recognition. And in this article, "Local Binary Pattern Algorithm" is used to detect faces structured in Open CV using Python language as well as "Haar Cascade Algorithm" to recognise human faces. Even with varied expressions, this classifier, when combined with other current algorithms, generates a high recognition rate, effective feature selection, and a minimal selection of false positive features. Only 200 of the 6000 elements in Haar's feature-based cascade classification algorithm are used to obtain an 85% identification rate.

[3] Ni Kadek Ayu Wirdiani, PrabaHridayami, Ni Putu Ayu Widiari, "Face Identification Based on K-

Nearest Neighbor", 2019 Scientific Journal of Informatics.

This paper's major goal is to identify faces using facial traits for security reasons. K-Nearest Neighbor is the classification approach used in this study (KNN). To choose an appropriate example value, the KNearest Neighbor method employs a neighbour classifier. An instance-based learning cluster is part of K-NN. In this study, principal component analysis is used to construct facial recognition (PCA). In this study, contrast stretching, grayscale segmentation, and hair cascade segmentation were utilised as pre-processing techniques. 30 persons participated in this study, with 3 pictures used for training and 2 photos for assessment. The greatest accuracy of 81% at k=1 is found in the results of multiple testing of the k-value.

[4] Cong Geng, & Jiang, X. (2009). "Face recognition using sift features". 2009 16th IEEE International Conference on Image Processing (ICIP).

A successful broad object detection/recognition method is the Scale Invariant Feature Transform (SIFT). In this article, we provide two fresh methods for face identification based on the original SIFT algorithm: Volume SIFT (VSIFT) and Partial Description SIFT (PDSIFT). We contrast feature-based techniques SIFT and PDSIFT with absolute approaches Fisherface(FLDA), null space approach (NLDA), and eigenfeature regularisation and extraction (ERE). PDSIFT outperforms the original SIFT method, according to experiments with the ORL and AR datasets. PDSIFT also outperforms FLDA and NLDA greatly and achieves performance equivalent to ERE, making it the most effective holistic technique.

[5] S.M Amir Khan , Md. Nazmul Islam Shuzan, Moajjem Hossain Chowdhury "Smart Entrance System Using Computer Vision at Corporate Environment", 2019 International Journal of Recent Technology and Engineering (IJRTE)

The corporate office is suggested a method in this paper for registering its staff. The Viola-Jones algorithm is used for face recognition in this system. An employee noticed a neck twist and used the Kazemi algorithm to fix the face alignment. To assess if male workers may have distinct facial moustaches and beards on various days, we apply a screening technique. In order to visualise how he might seem if he had a moustache or beard, filtering was applied to the picture. Due to the decreased chance of erroneous detection, all face traits are taken into account. The database now contains these pictures. Ignefaces is used for face recognition, which makes counting an employee in an intelligent access system simpler and quicker. We are aware that older systems allow for card punching, which is sometimes seen as dishonest, by firm personnel to get access. However, these onerous chores may be reduced with the development and implementation of intelligent entry systems that use face recognition.

3 PROBLEM SPECIFICATION

We must first produce our own face recognition dataset in order to begin this procedure. A created data set consists of a number of images of the same person. The resulting dataset is then split into training and validation halves. We train images using the CNN method and the VGG16 model to address the inadequacies of the previous models. We enter the validation stage after a certain number of training iterations. We test our model in a real-time setting that detects individuals from the dataset once the model execution is finished. If the individual's face doesn't match, an email is sent to the owner.

4. System Architecture

4.1 proposed system

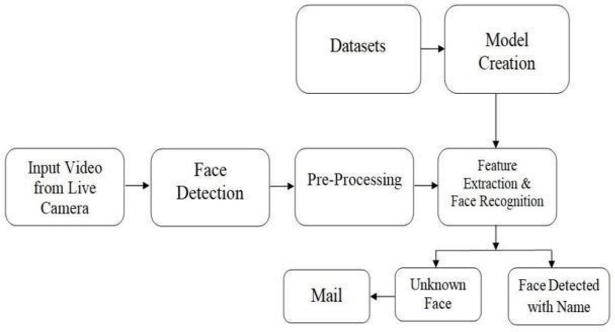

Fig1.Block Diagram of our proposed system

This study uses deep learning algorithm approaches to look for illegal access to a monitored website. The block diagram in Fig. 5 illustrates how we first generate a unique data collection for each individual who will be processed. Basically, each unique human picture that the camera records is used to construct a dataset by extracting face areas and traits. The training and testing portions of the dataset are separated. Following dataset segmentation, the VGG16 model with CNN layers is used to build the actual model. A.h5 file will be used to save the model. The model is then put into practice and examined in a genuine setting.

4.1.1 Creation of Datasets

A picture must first be captured before a data record can be created. There are several ways to take images, including using a camera, a laptop camera, a smartphone, etc. When building a face dataset, the HaarCascaded frontal face default.xml file, used to recognize and record faces, is imported.

4.1.2 Input Video from Live Camera

A picture of the individual is taken from the input video and compared to the model's data set. Input films are captured using a variety of image capture devices, including standalone cameras, laptop cameras, cellphones, and other devices. A digital picture is captured during the process of image capture.

4.1.3 Face Detection

A multitude of applications employ facial recognition technologies to identify people's faces in digital photographs. The mental process through which people notice and pay attention to faces in a visual context is often referred to as face recognition.

4.1.4 Pre-processing

Preprocessing is the process of converting unprepared input into data that is suitable for deep learning. The most crucial and first stage in creating a deep learning model is this. It is a word for operations on pictures that use intensity images as both the input and the output at the most fundamental level of abstraction.

4.1.5 Feature Extraction

It is possible to extract features manually or automatically. A technique for manually extracting features must be developed after identifying and defining the characteristics that are pertinent to a given issue.

4.1.6 Face Recognition

One of the most popular applications of face recognition is for facial recognition. A biometric approach for identifying a person's face after exposure is face recognition.

4.1.7 Model Generation

The model is constructed using a convolutional neural network (CNN), and it is trained and validated using a VGG16 model. A Python programming language Jupyter notebook is used to carry out this task. We must first use the Keras library in Jupyter Notebook to import the vgg16 model into our system before we can do this procedure.

5. THEORETICAL CONSIDERATIONS

This section basically talks about the Deep Learning, CCTV systems and the convolutional neural networks. Theories behind these two concepts are combined to build an intelligent computer vision system.

5.1 Deep Learning

A subset of artificial intelligence based on artificial neural networks employing representational learning, deep learning (also known as deep structured learning) is a method of machine learning that is a member of a broader family of methods. Unsupervised, semi-supervised, and supervised learning are the three different styles of education. To extract high-level characteristics from raw data, it employs numerous layers. Each level of deep learning gains the ability to represent the input data in a more abstract and composite manner. Figure 3.1 depicts a fundamental deep learning architecture that consists of input, hidden, and output layers. The input and output layers of a deep neural network are the layers that are visible. The final prediction or classification is made by a deep learning model in the output layer after the data has been processed in the input layer. Deep learning techniques traverse numerous levels of neural networks or hidden layers.

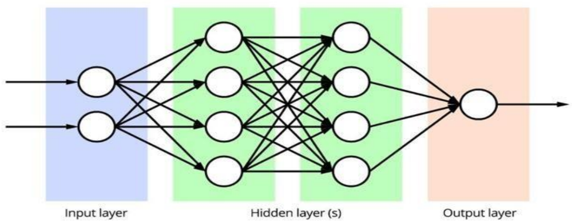

Fig 2: Deep Learning Architecture

5.2 Types of Deep Learning

Three categories are used to categorise deep learning

5.2.1 Supervised Learning

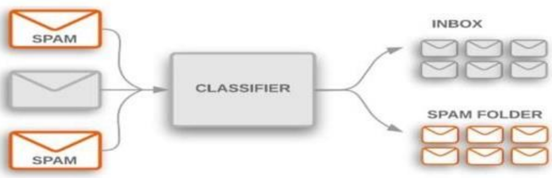

Fig 3: Supervised Learning

5.2.2 Unsupervised Learning

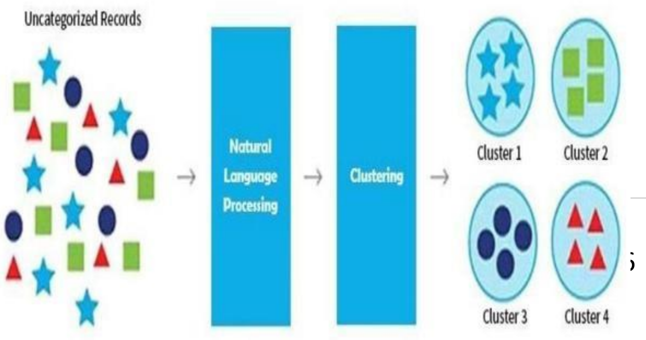

Fig 4: Unsupervised Learning

5.2.3 Reinforcement Learning

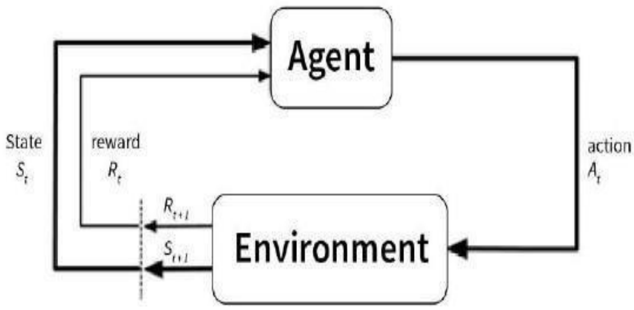

Fig 5. Reinforcement Learning

5.3 Convolutional Neural Network

Convolutional neural networks are multi-stage architectures that can be trained to learn invariant features. It is composed of filter banks, some non-linear and feature pooling layers . In general, there are three layers that make up the convolutional neural network, namely convolutional layer, pooling layer and the fully - connected layer. A typical architecture of CNN is shown in Figure 1. The convolutional layer is the core part of the network because it contains the local connections and weights of the shared characteristics obtained say, from an image. The convolutional layer is where features of the input/s are learned. There are feature maps composed of neurons which extract the local spatial characteristics in the former layer . For learning a new feature, the convolution of the input feature maps with a learned kernel is performed and then the results are passed into a non-linear activation function like the sigmoid, tanh and Relu . With this, many different features are obtained.

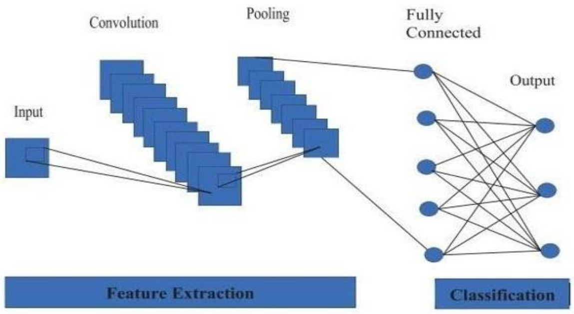

Fig 6: Basic Architecture of Convolutional Neural Network

5.4 VGG16

The Visual Geometry Group 16 is known as VGG16. Convolutional neural network VGG16 was trained using data from a portion of the ImageNet dataset, which consists of 14 million images organised into 22,000 categories. This model was proposed in the 2015 article Very Deep Convolutional Networks for Large-Scale Image Recognition by K. Simonyan and A. Zisserman. The structure of the VGG16 model's pre-trained model is shown in Fig. 3.8. The "top" (the fully connected layer) of the model is left out while the convolutional and pooling layers are imported. We apply VGG16's convolutional layers to our input images, and the result is a feature stack of recognised visual features. The three-dimensional feature stack can now be quickly flattened into a NumPy array, making it ready for any modelling you desire.

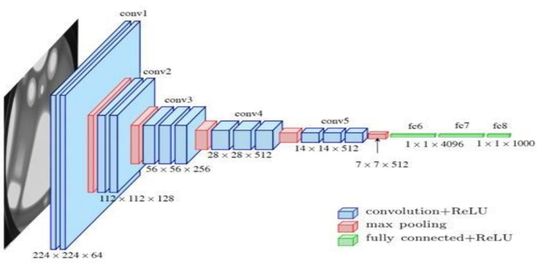

Fig 7: VGG16 Architecture

6. SOFTWARE IMPLEMENTATION

6.1 Python

Python is a straightforward, widely-used programming language. The most recent version of the Python programming language, Python 3, is utilised in applications for deep learning, machine learning, and web development. Both novice programmers and seasoned users of other languages like C++ and Java may benefit from learning the Python programming language. Python supports procedural and object-oriented programming. Python is an interpreted, object-oriented, high-level, dynamically semantic programming language.

The biggest strength of Python is huge collection of standard library which can be used for the following:

- Machine Learning
- Deep Learning
- GUI Applications (like Tkinter, PyQt etc.)
- Web frameworks like Django (used by YouTube, Instagram, Dropbox)

Image processing (like openCV, Pillow).

6.2 Jupyter Notebook

You can create and share documents with live code, equations, visualisations, and explanatory text using the open-source online application Jupiter Notebook. Data transformation and cleansing, numerical simulation, statistical modelling, data visualisation, machine learning, and other processes are some of its uses.

Software Packages

Table 1: Software Packages

Software Packages	Version
Python	3.6 and Above
Anaconda Distribution	3
Jupyter Notebook	Preinstalled with Anaconda

Libraries and their versions

Table 2: Libraries and their versions

Library	Version
OpenCV	4.5.5.64
TensorFlow	2.1.0 and Above
Keras	2.3.0 and Above
Numpy	1.20.3
Matplotlib	3.4.3
SMTP	Inbuilt Version

7. TESTING

7.1 Test Cases

Table 3: Unit Testing

Test Id	Testing Type	Description	Input	Predicted Output	Actual Output	Result
1	Unit Testing	Packages	Package Installation	Done Successful	All Packages Installed	Test Case Passed
2	Unit Testing	Dataset Creation	Database	Dataset Created Successfully	Dataset Created Successfully	Test Case Passed
3	Unit Testing	Live Video from Camera	Laptop Camera	Camera capturing images	Camera Capturing Images Successfully	Test Case Passed
4	Unit Testing	Model Generation	Using CNN Layers	Classifier Created	Classifier Created Successfully	Test Case Passed
5	Unit Testing	Face Recognition	Laptop Camera	Recognition Done	Recognition Done	Test Case Passed
6	Unit Testing	Sending Mail in case Unknown Person	Laptop Camera & SMTP	Send Mail using SMTP	Mail Sent using SMTP	Test Case Passed

Database Creation

We must first establish a folder where our code will execute in a certain path depending on the names of the people on the desktop in order to construct each person's database. Sample data sets for each person are shown in Fig. 8.

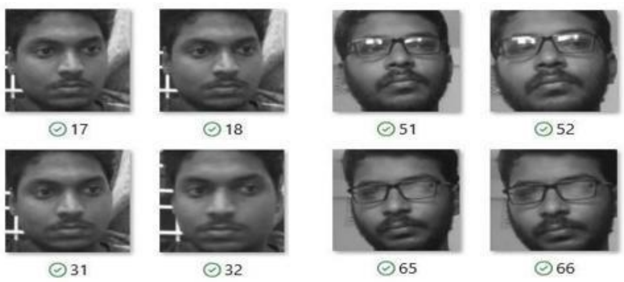

Fig 8. Datasets of two authorized persons

Camera

If you want to use the laptop camera to use an external camera, pass argument as one (1). If you want to check the camera's functionality, pass argument as zero (0) for video capture.

Model Generation

After generating the datasets for the model to be trained and validated, create a new Model creation.ipynb file. Before training and validating the model, import the CNN layers from the VGG16 model.

Save the file with a.h5 extension after the training and validation portions have been finished. Large volumes of data in the form of multidimensional arrays are often stored in.h5 files.

Face Recognition

After the model has been created, it needs to be validated using real-time images of authorised and unauthorised people using the model. Using this model, we can determine a person's level of authority.

Sending mail in case of unauthorized person

When the camera module spots an unauthorised person, it uses the SMTP module to send an alert message to the concerned owner or security team.

8. RESULTS

The various observations made and findings from this work are covered in this chapter.

- Following dataset creation, the generated data is trained using the CNN model, and the CNN layers are implemented using the VGG16 model (see Fig. 9 for an example of this).

Model: "model_4"

Layer (type)	Output Shape	Param #
input_5 (InputLayer)	[(None, 224, 224, 3)]	0
block1_conv1 (Conv2D)	(None, 224, 224, 64)	1792
block1_conv2 (Conv2D)	(None, 224, 224, 64)	36928
block1_pool (MaxPooling2D)	(None, 112, 112, 64)	0
block2_conv1 (Conv2D)	(None, 112, 112, 128)	73856
block2_conv2 (Conv2D)	(None, 112, 112, 128)	147584
block2_pool (MaxPooling2D)	(None, 56, 56, 128)	0
block3_conv1 (Conv2D)	(None, 56, 56, 256)	295168
block3_conv2 (Conv2D)	(None, 56, 56, 256)	590080
block3_conv3 (Conv2D)	(None, 56, 56, 256)	590080
block3_pool (MaxPooling2D)	(None, 28, 28, 256)	0
block4_conv1 (Conv2D)	(None, 28, 28, 512)	1180160
block4_conv2 (Conv2D)	(None, 28, 28, 512)	2359808
block4_conv3 (Conv2D)	(None, 28, 28, 512)	2359808
block4_pool (MaxPooling2D)	(None, 14, 14, 512)	0
block5_conv1 (Conv2D)	(None, 14, 14, 512)	2359808
block5_conv2 (Conv2D)	(None, 14, 14, 512)	2359808
block5_conv3 (Conv2D)	(None, 14, 14, 512)	2359808
block5_pool (MaxPooling2D)	(None, 7, 7, 512)	0
flatten_4 (Flatten)	(None, 25088)	0
dense_4 (Dense)	(None, 2)	50178

Total params: 14,764,866
Trainable params: 50,178
Non-trainable params: 14,714,688

Found 600 images belonging to 2 classes.
Found 600 images belonging to 2 classes.

Fig 9: Implementation of CNN Layers

• Following the construction of the layers, the training and validation process for already provided datasets starts. Model accuracy is shown in Fig. 10, and model loss is depicted in Fig. 11 following training and validation for 5 epochs.

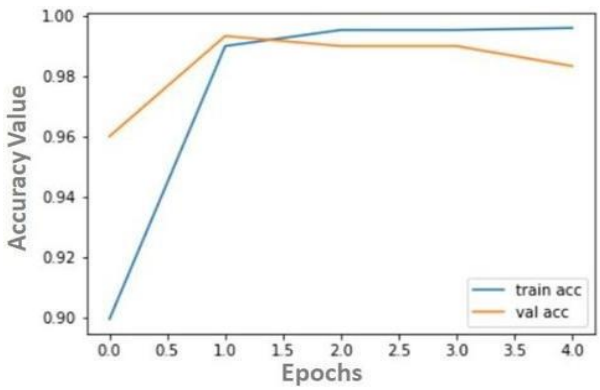

Fig 10: Accuracy of the model

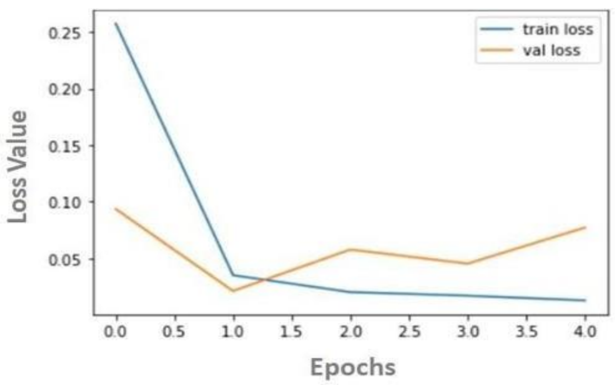

Fig 11: Loss of the model

The created datasets are used to train the model, and real-time faces captured by a laptop camera are used to validate it. Fig. 12 depicts the model detecting authorised person 1 and Fig. 13 depicts the model detecting authorised person 2, respectively. If the person is approved, their name will be displayed.

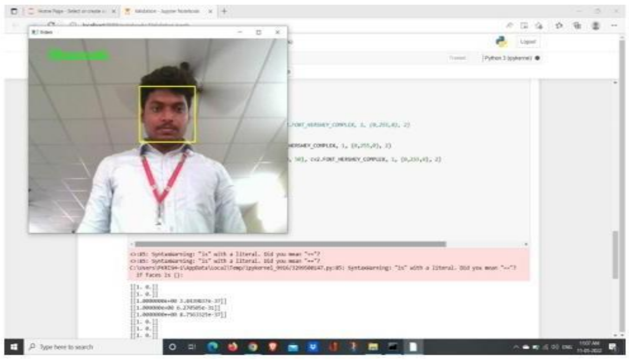

Fig 12: Authorized Person 1

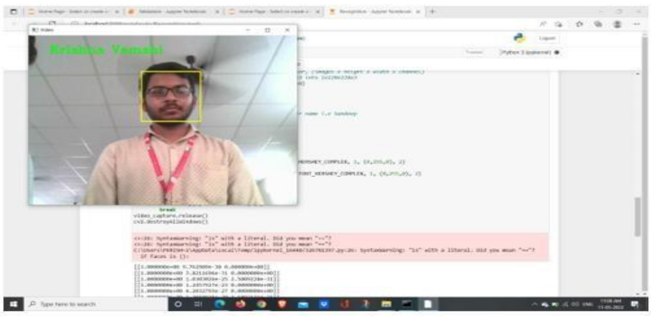

Fig 13: Authorized person 2

By leveraging the security cameras (system cameras), this model identifies illicit entrance into restricted sites in real time by automatically comparing the pictures with previously produced datasets. Figure 14 below demonstrates that the model failed to identify the subject and displayed the result as an unidentified face.

Fig 14: Unauthorized/ Unknown Person 1

If the surveillance system discovers an unauthorised entry, it will automatically take the person's picture and send it to the relevant security team via SMTP server with a message alerting them to the discovery/entry of an unknown person. Figure 15 illustrates how the security team received an alert mail informing them that an unauthorised person had entered the site.

This model will take a person's photo and store it in an unknown database in our file location path if it discovers an additional unauthorised face through a surveillance camera after comparing the dataset. Figure 16 below demonstrates that the model is not able to identify the person and displays the output as unknown face found.

Fig 16: Unauthorized/ Unknown Person 2

The person's photo will be sent to the business owner using the SMTP protocol after being stored in an unidentified database along with an alert message and the picture of the unidentified person, as shown in Fig. 17. The security team received an alert mail stating that an unauthorised person had entered the site.

The table below summarises the outcomes of all the algorithms along with the research's findings. As we can see, the accuracy levels have increased as a result of this work when compared to the existing models when using three sets of images.

Table 4: Performance Comparison between other models

Model	Accuracy(%)
Eigenfaces [1]	80
K Nearest Neighbor [3]	81
Haar Cascaded [2]	85
LDA (Linear Discriminant Analysis)	86
PCA (Principle Component Analysis)	88
SVM (Support Vector Machine)	91.2
Fisherfaces	93
Viola Jones Algorithm	95
Convolutional Neural Network	98

9. CONCLUSION FUTURE SCOPE

Conclusion

Deep learning is being used for real-time monitoring in this project. This model identifies and recognises faces of individuals using convolutional neural networks by fitting the pre-built VGG16 model. It utilises a camera to identify those entering a regulated place who are permitted and who are not. An alert email is automatically sent to the impacted team when an unauthorised individual visits a controlled site, accompanied with a picture of the illegal person accessing the site. According to test findings, utilising CNN classifier may provide accuracy levels between 95% and 98%. This feature aids in identifying unlawful entry into forbidden places. By extending the data set, system performance is further evaluated.

Future Scope

The system will be trained on a bigger dataset in the future, and experiments with other age groups and persons wearing masks will be conducted on additional datasets. To speed up execution, a graphics processing unit (GPU) might be employed. Future research on deep learning algorithms for real-time monitoring will build on the methodology of this study. This method has been improved so that it can now distinguish between authorised and unauthorised people of all ages, colours, haircuts, beards, masks, backlight, and other characteristics. It is also possible to employ several neural network types to recognise humans using security cameras. Once the system has perfect identification accuracy, this application may be utilised to enhance monitoring in limited locations.

REFERENCES

[1] M A Imran, M S U Miah, H Rahman, “Face Recognition using Eigenfaces” International Journal of Computer Applications (0975 – 8887) 2015.

[2] SenthamizhSelvi.R ,D.Sivakumar, Sandhya.J.S, “Face Recognition Using Haar - Cascade Classifier for Criminal Identification”, 2019 International Journal of Recent Technology and Engineering (IJRTE).

[3] Ni Kadek Ayu Wirdiani, PrabaHridayami, Ni Putu Ayu Widiari, Face Identification Based on K-Nearest Neighbor, 2019 Scientific Journal of Informatics.

[4] Cong Geng, & Jiang, X. (2009). Face recognition using sift features. 2009 16th IEEE International Conference on Image Processing (ICIP).

[5] S.M Amir Khan , Md. Nazmul Islam Shuzan, Moajjem Hossain Chowdhury “Smart Entrance System Using Computer Vision at Corporate Environment”, 2019 International Journal of Recent Technology and Engineering (IJRTE)

[6] P. Viola and M. Jones, "Robust Real-Time Face Detection", International Journal of Computer Vision, vol. 57, no. 2, pp. 137-154, 2004.

[7] M. Chillaron, L. Dunai, G. P. Fajarnes, and I. L. Lengua, “Face detection and recognition application for Android,” IECON 2015 - 41st Annual Conference of the IEEE Industrial Electronics Society, 2015.

[8] M.-Y. Lee, Y.-S. Baek, S.-M. Kim, H. Kim, B.-T. Koo, and J.-H. Lee, “HW/SW co-design of face detection & recognition on virtual platform,” 2016 IEEE International Conference on Consumer Electronics-Asia (ICCE-Asia)

[9] Matija Burić, Marina Ivašić-Kos, MiranPobar, “Player Tracking in Sports Videos”, IEEE International conference on cloud computing technology and science.

[10] Hanmandlu, M., Gupta, D., &Vasikarla, S. (2013). Face recognition using Elastic bunch graph matching. 2013 IEEE Applied Imagery Pattern Recognition Workshop (AIPR).

[11] R. Girshick, J. Donahue, T. Farrell and J. Malik, “Rich feature hierarchies for accurate object detection and semantic segmentation”, In CVPR, 2014.

[12] Uijlings J R R, Sande K E A V D, Gevers T, et al, “Selective search for object recognition”, International journal of computer vision, 2013,, 104(2):154-171.

eISSN 1303-5150