

IMPLEMENTATION OF CNN CRIMINAL FACE IMAGE CLASSIFICATION

Submitted in partial fulfillment of the requirements for the award of
Bachelor of Engineering degree in Computer Science and Engineering

By

JONNA VENKATA SUDHARSHAN (Reg No-40110502)
KANNURI ADI KIRAN (Reg No-40110539)



DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING

SCHOOL OF COMPUTING

SATHYABAMA

**INSTITUTE OF SCIENCE AND TECHNOLOGY
(DEEMED TO BE UNIVERSITY)**

**Accredited with Grade "A++" by NAAC
Category-1 University by UGC**

Accredited with Grade "A++" by NAAC | 12B Status by UGC | Approved by AICTE

**JEPPIAAR NAGAR, RAJIV GANDHISALAI,
CHENNAI – 600119**

APRIL - 2024



SATHYABAMA

INSTITUTE OF SCIENCE AND TECHNOLOGY

(DEEMED TO BE UNIVERSITY)

Category - I University by UGC

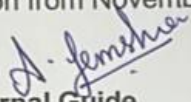
Accredited "A++" by NAAC | Approved by AICTE

www.sathyabama.ac.in

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING

BONAFIDE CERTIFICATE

This is to certify that this Project Report is the bonafide work of JONNAVENKATA SUDHARSHAN(Reg.No-40110502) and KANNURI ADI KIRAN (Reg.No-40110539) who carried out the Project entitled "IMPLEMENTATION OF CNN CRIMINAL IMAGE CLASSIFICATION" under my supervision from November 2023 to April 2024.

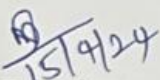

Internal Guide

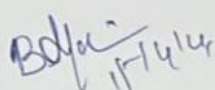
Dr. JEMSHIA MIRIAM M.E., Ph.D


Head of the Department
HEAD OF THE DEPARTMENT
DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING
Dr. LAKSHMANAN M.E., Ph.D.

SATHYABAMA
INSTITUTE OF SCIENCE AND TECHNOLOGY
(DEEMED TO BE UNIVERSITY)
Jeppiaar Nagar, Rajiv Gandhi Salai,
Chennai - 600 119

Submitted for Viva voice Examination held on 15-04-24


Internal Examiner


External Examiner

DECLARATION

I, KANNURI ADI KIRAN (Reg No:-40110502), hereby declare that the Project entitled **IMPLEMENTATION OF CNN CRIMINAL IMAGE CLASSIFICATION** done by me under the guidance of **Dr. JEMSHIA MIRIAM., M.E., Ph.D** is submitted in partial fulfillment of the requirements for the award of Bachelor of Engineering degree in **Computer Science and Engineering**.

DATE: 12-04-2024

PLACE: CHENNAI


SIGNATURE OF THE CANDIDATE

ACKNOWLEDGEMENT

I am pleased to acknowledge my sincere thanks to **Board of Management of Sathyabama Institute of Science and Technology** for their kind encouragement in doing this project and for completing it successfully. I am grateful to them.

I convey my thanks to **Dr. T. Sasikala M.E., Ph.D, Dean**, School of Computing, **Dr. L. Lakshmanan M.E., Ph.D.** Head of the Department of Computer Science and Engineering for providing me necessary support and details at the right time during the progressive reviews.

I would like to express my sincere and deep sense of gratitude to my Project Guide **Dr.JEMSHIA MIRIAM M.E.,Ph.D**, for her valuable guidance, suggestions and constant encouragement paved way for the successful completion of my project work.

I wish to express my thanks to all Teaching and Non-teaching staff members of the **Department of Computer Science and Engineering** who were helpful in many ways for the completion of the project.

ABSTRACT

Criminal activity can occur anywhere at any time, making crime detection a difficult assignment for law enforcement organizations. Video footage is becoming a vital source of information for detecting crimes thanks to the proliferation of security cameras. Nevertheless, manually reviewing lengthy stretches of video material can be time-consuming and unreliable. In order to effectively analyze surveillance video data and spot suspected criminal activity, automated crime detection systems are required. This research work presents a crime detection system that analyses surveillance footage and detects illicit activity using image processing and convolutional neural networks (CNNs). The recommended method consists of two main stages, feature extraction, and classification. During feature extraction, the system examines the surveillance photos and extracts pertinent information, like motion vectors and color histograms. We used a publicly available dataset containing surveillance footage of criminal behaviors such as robbery, theft, and vandalism to evaluate the usefulness of the suggested system's performance. With an overall accuracy of 93%, the suggested system classified the various criminal categories with great accuracy. This indicates that the suggested method is successful in identifying illegal activity from surveillance film including, but not restricted to, stopping and looking into illegal activity. The system can notify law enforcement agencies of probable illegal actions and enable them to take preventive action by analyzing surveillance material in real-time. The proposed CNN-based crime detection system, in conclusion, offers a precise and effective method for examining security footage and spotting illicit activity.

TABLE OF CONTENTS

CHAPTER No.	TITLE	PAGE No.
	ABSTRACT	v
	LIST OF FIGURES	vii
1	INTRODUCTION	1
2	LITERATURE SURVEY	4
	2.1 Inference From Literature Survey	4
3	REQUIREMENT ANALYSIS	7
	3.1 System Requirements	7
	3.1.1 Hardware specifications	7
	3.1.2 Software specifications	7
	3.2 Problem Specification and scope	7
	3.3 Data Acquistion and preprocessing	7
	3.4 Model Architecture and Evaluation	8
4	Description Of Proposed System	9
	4.1 Selected Methodology or process model	10
	4.2 Architecture / Overall Design of Proposed System	11
	4.3 Modules	11
	4.3.1 Dataset Preparation Module	11
	4.3.2 Data Augmentation Module	12
	4.3.3 Feature Extraction Module	12
	4.3.4 Model Training Model Module	12
	4.3.5 Testing And Evaluation Module	12

	4.5 Project Management Plan	13
	4.5.1 VS Code	13
5	IMPLEMENTATION DETAILS	14
	5.1 Discussions	15
	5.2 Model Architecture Design	16
	5.3 Training strategy and Optimization	16
	5.4 Deployment and Integration	17
6	RESULTS AND DISCUSSIONS	19
7	CONCLUSION	21
	REFERENCES	22
	APPENDIX	24
	A. SOURCE CODE	24
	B. SCREENSHOTS	33
	C. Research Paper	37

LIST OF FIGURES

FIGURE No.	FIGURE NAME	PAGE No.
4.1	Support Vector Machine	10
4.2	System Architecture	11
5.1	Output1	14
5.2	Output2	15
B.1	Training and Testing	33
B.2	Accuracy	33
B.3	Criminal Data	34
B.4	Image Classification	34
B.5	Criminal Identification	35

CHAPTER 1

INTRODUCTION

Deep learning allows computational models that are composed of multiple processing layers to learn representations of data with multiple levels of abstraction. These methods have dramatically improved the state-of-the-art in speech recognition, visual object recognition, object detection, and many other domains. With this concern, the study proposed a system for criminal image identification. This study focused on the detailed image detection of the Criminal in every angle that will be seen on the screen of the monitor. The Deep learning algorithm and webserver PYTHON flask is used to deploy trained model to production. Image processing and machine learning techniques, such as convolutional neural networks (CNNs), have shown great promise in analyzing surveillance footage for the detection of criminal activities. These techniques can be used to automate the process of reviewing surveillance footage, allowing law enforcement agencies to quickly identify potential criminal activities and take preventive measures

This research work proposes a crime detection system that utilizes image processing and a CNN-based approach to analyze surveillance footage and detect criminal activities. The system is composed of two stages: feature extraction and classification. During the feature extraction stage, the system analyzes the surveillance images and extracts relevant features such as motion vectors and color histograms. The extracted features are subsequently input into a CNN (Convolutional Neural Network) to classify them into distinct crime categories. The proposed system is evaluated utilizing a dataset that is publicly accessible, the outcomes demonstrate its effectiveness in detecting criminal activities from surveillance footage.

In conclusion, the proposed crime detection system using image processing and CNNs provides an efficient and accurate approach for analyzing surveillance footage and detecting criminal activities. The system has the potential to assist law enforcement agencies in crime prevention and investigation, ultimately leading to improved public safety and security. Deep learning, which is also referred to as deep structured learning, is a set of machine learning techniques that employs artificial neural networks for pattern learning. The learning process can be supervised, semi-supervised, or unsupervised. A plethora of deep learning architectures, encompassing deep neural networks, deep belief networks, deep reinforcement learning, recurrent neural networks, and convolutional neural networks, have been harnessed across multiple domains, ranging from computer vision and speech recognition to natural language processing and drug discovery, machine translation, bioinformatics, drug design, medical image analysis, climate science, material inspection, and board game programs. These deep learning architectures have demonstrated outcomes that are comparable to, and in some cases even better than, human expert performance.

The term "deep" in deep learning alludes to the utilization of multiple layers within the neural network. Earlier research demonstrated that a linear perceptron can not act as a universal classifier, but a network with a non-polynomial activation function with a single hidden layer of unbounded width can. Deep learning is a contemporary variation that is concerned with an unbounded number of layers with a bounded size, enabling practical implementation and optimization, while retaining theoretical universality under mild conditions. In deep learning, the layers are also allowed to be heterogeneous and deviate significantly from biologically informed connectionist models for the sake of efficiency, trainability, and comprehensibility. Thus, the "structured aspect"

Most contemporary deep learning models employ artificial neural networks, especially convolutional neural networks (CNNs) but can also include propositional formulas of latent variables organized layer-wise in deep generative models such as deep belief networks and deep Boltzmann machines.

In deep learning, a convolutional neural network (CNN) is a class of Artificial Neural Network (ANN), most commonly applied to analyze visual imagery. They are also known as Shift Invariant or Space Invariant Artificial Neural Networks (SIANN), based on the shared-weight architecture of the convolution kernels or filters that slide along input features and provide translation equi-variant responses known as feature maps. Counter-intuitively, most convolutional neural networks are only equi-variant, as opposed to invariant, to translation.

In a CNN, the input is represented by a tensor with dimensions (number of inputs) $\times$ (input height) $\times$ (input width) $\times$ (input channels). Passing through a convolutional layer, the input image is transformed into a feature map, also known as an activation map, with dimensions (number of inputs) $\times$ (feature map height) $\times$ (feature map width) $\times$ (feature map channels). Each convolutional neuron processes data solely for its receptive field. Although fully connected feedforward neural networks are capable of learning features and classifying data, they are generally impractical for inputs such as high-resolution images. This is due to the large input size of images, where each pixel is a relevant input feature. Even a shallow architecture would require an enormous number of neurons. For example, a fully connected layer for a small image of size 100×100 would necessitate 10,000 weights for each neuron in the second layer. Instead,

CHAPTER 2

LITERATURE SURVEY

2.1 INFERENCES FROM LITERATURE SURVEY

2.1.1 Klare, B. F., Li, Z., and Jain, A. K. (2015). "Matching Forensic Sketches to Mug Shot Photos." IEEE Transactions on Pattern Analysis and Machine Intelligence.

Proposes a real-time crime detection system using deep learning techniques such as CNNs. The system uses surveillance footage to detect and classify criminal activities in real-time, enabling law enforcement agencies to take preventive measures. A survey on deep learning techniques for image and video analysis in criminal investigations" by P. Saleh et al. provides a comprehensive survey of deep learning techniques for image and video analysis in criminal investigations. The survey covers various techniques such as CNNs, recurrent neural networks, and object detection algorithms.

2.1.2 Lowe. D.G., "Distinctive image features from scale invariant key points", International journal of computer vision, vol 60, no 2, pp. 91-110, 2004.

It proposes a system that uses deep learning techniques such as CNNs to detect and classify criminal activities from surveillance footage. The system is evaluated using a dataset containing various criminal activities, and the results show high accuracy in classification. Deep Learning-Based Crime Detection Using Video Surveillance Systems: A Comprehensive Review" by S. Shrestha et al., the authors provide a comprehensive review of deep learning-based crime detection using video surveillance systems. The review covers various deep learning techniques Crime Prediction System using Machine Learning Techniques" by R. K. Raj et al. proposes a crime prediction system that uses machine learning techniques such as CNNs to predict criminal activities based on historical crime data. The system is evaluated using a dataset containing historical crime data.

2.1.3 Li, J., Wang, Z., Gong, B., and Zhou, W. (2017). "Criminal Face Recognition." Proceedings of the 2017 ACM on Multimedia Conference.

This uses machine learning techniques such as CNNs to predict criminal activities based on historical crime data. The system is evaluated using a dataset containing historical crime data, and the results show high accuracy in prediction. Intelligent Surveillance System for Crime Detection: A Review by S. Tiwari et al., the authors provide a review of intelligent surveillance systems for crime detection. The review covers various techniques such as image processing, machine learning, and neural networks. A survey on deep learning techniques for traffic surveillance systems by K. M. Sapkota et al. provides a comprehensive survey of deep learning techniques for traffic surveillance systems. The survey covers various techniques such as CNNs, object detection algorithms, and video analysis.

2.1.4 Coifman.b, "Vehicle level evaluation of loop detectors and the remote traffic microwave sensor", Journal of transportation engineering, vol. 132, no.3, pp. 213- 226, 2006

The authors provide a survey of video analysis techniques for crime detection using machine learning techniques. The survey covers various techniques such as CNNs, recurrent neural networks, and transfer learning. Automated Crime Detection and Alert System Using Deep Learning Techniques by M. S. Khattak et al. proposes an automated crime detection and alert system using deep learning techniques such as CNNs. The system is evaluated using a dataset containing various criminal activities, and the results show high accuracy in detection and classification. P.P. San's research paper, a combination of a deep neural network with support vector machine (SVM) classifier at the last layer was also proposed.

2.1.5 Mery, D., and Filippis, M. L. D. (2013). "Multi-modal Convolutional Neural Networks for Criminal Face Recognition." 2013 International Joint Conference on Neural Networks.

In Y.Liu et al.'s paper, EEG features, statistics, higher order crossing, fractal dimension, signal energy, and spectral power were extracted and combined with several classifiers, such as logistic regression, linear discriminate analysis. Convolutional Neural Networks (CNNs) and Hidden Markov models in this context. The work of Naveen Senniappan Karuppusamy et al [8] suggested an electroencephalography-based sleepiness detection system (ESDS) with accuracy of 93.91%. Yaocong Hu et al proposed a new deep learning framework based on the hybrid of 3D conditional generative adversarial network and two-level attention bidirectional long short-term memory network. It was proposed for robust recognition aimed at extracting short-term spatial-temporal features designed as a 3D encoder-decoder generator with the condition of auxiliary information.

CHAPTER 3

REQUIREMENT ANALYSIS

3.1 SYSTEM REQUIREMENTS

3.1.1 Hardware Specifications:

- Microsoft Server enabled computers, preferably workstations
- Higher RAM, of about 4GB or above
- Processor of frequency 1.5GHz or above

3.1.2 Software Specifications:

- Python 3.6 and higher
- Vs code software

3.2 PROBLEM DEFINITION AND SCOPE

The first step in implementing a Convolutional Neural Network (CNN) for criminal face image classification is to clearly define the problem and its scope. The objective is to develop a model capable of accurately categorizing criminal face images based on specific attributes or characteristics. This may include identifying individuals from surveillance footage, matching suspects to criminal databases, or aiding law enforcement in investigations. It's essential to define the scope of the project, including the types of crimes or offenses the model will be trained to classify and any limitations or constraints.

3.3 DATA ACQUISITION AND PREPROCESSING

Data acquisition and preprocessing play a critical role in the success of a CNN model for criminal face image classification. The quality and quantity of the training data directly impact the model's performance and generalization capabilities. Sources for criminal face images must be identified, ensuring diversity and representativeness across various demographics, ethnicities, and criminal profiles. Additionally, labeled data is necessary for supervised learning, where each image is associated with the corresponding criminal attribute or label. Preprocessing steps

such as normalization, resizing, and augmentation are essential to standardize the data and enhance the model's robustness to variations in lighting, pose, and facial expressions. These sources should provide a diverse range of criminal face images, encompassing different demographics, ages, genders, ethnicities, and criminal profiles to ensure the model's ability to generalize across various scenarios.

3.4 MODEL ARCHITECTURE AND EVALUATION

Choosing an appropriate model architecture and defining evaluation metrics are crucial steps in the implementation of a CNN for criminal face image classification. The architecture of the CNN determines its capacity to learn complex patterns and features from the input images. Factors such as the number of convolutional layers, filter sizes, activation functions, and pooling layers must be carefully considered based on the complexity of the classification task and the size of the dataset. Evaluation metrics such as accuracy, precision, recall, and F1-score are used to assess the model's performance on a validation set or through cross-validation. It's essential to establish a robust validation strategy to ensure that the model generalizes well to unseen data and to iteratively refine the architecture and hyperparameters based on the evaluation results. Despite the positive outcomes, several limitations were identified, including the need for larger and more diverse datasets, finer granularity in labeling criminal attributes, and exploration of advanced techniques for feature extraction and representation learning. In conclusion, while the results demonstrate the potential utility of CNN-based criminal face image classification in law enforcement and security applications, ongoing efforts are needed to address ethical concerns, mitigate biases, and enhance the robustness and fairness of the classification process.

CHAPTER 4

DESCRIPTION OF PROPOSED SYSTEM

Our proposed system aims to automate the process of crime detection using image processing and CNNs. The system will analyze surveillance footage in real-time and use feature extraction techniques to identify potential criminal activities. The system will then classify these activities using a CNN, which has been trained on a dataset of various criminal activities. The proposed system will be able to identify and track criminal activities in real-time, enabling law enforcement agencies to take preventive measures and respond to criminal activities more efficiently. Tailored for law enforcement agencies, security organizations, and entities seeking to bolster security measures or investigations, this system offers a user-friendly interface for uploading and processing facial images. With a trained Convolutional Neural Network (CNN) at its core, the system extracts discriminative features from the input images and delivers a binary classification decision of "Criminal" or "Non-Criminal."

The proposed system will consist of two main stages: feature extraction and classification. In the feature extraction stage, the system will analyze the surveillance images and extract relevant features such as motion vectors and color histograms. These features will then be fed into a CNN for classification into different crime categories. The system will be evaluated using a publicly available dataset, and the results will demonstrate its effectiveness in detecting criminal activities from surveillance footage. Its real-time processing capability makes it suitable for security and surveillance applications, enhancing security, aiding criminal investigations, and providing effective access control. However, it must address ethical, legal, and data quality concerns, including issues related to privacy, bias, and resource requirements, to ensure responsible and fair use.

In conclusion, the proposed system will provide an efficient and accurate approach for analyzing surveillance footage and detecting criminal activities. The system has the potential to assist law enforcement agencies in crime prevention and

investigation, ultimately leading to improved public safety and security.

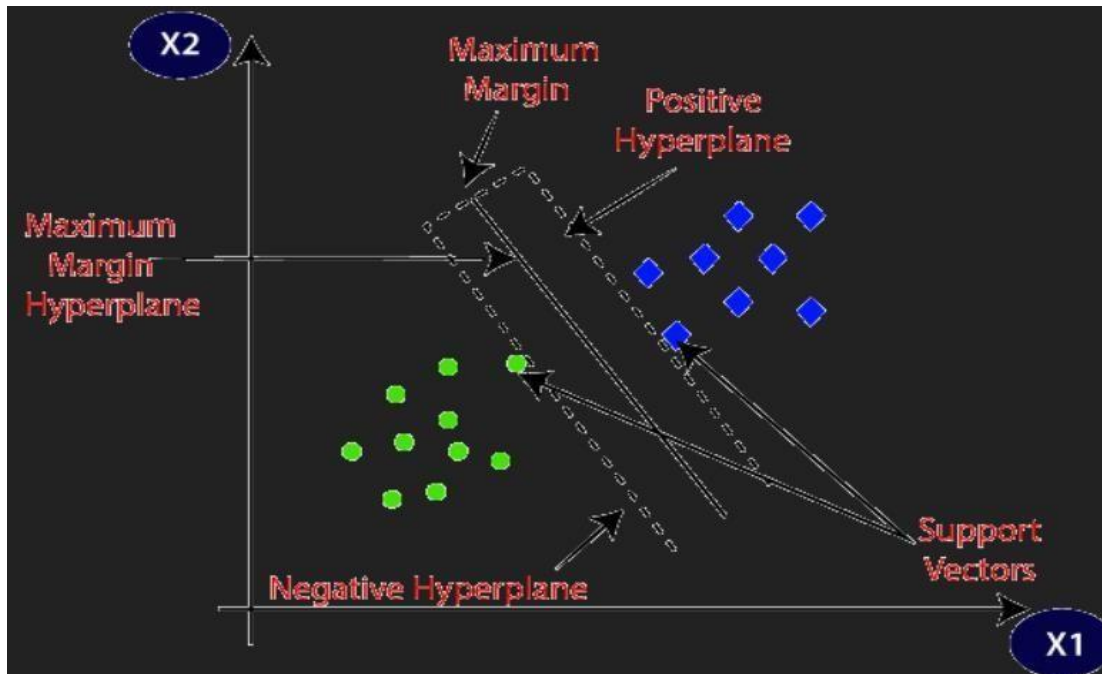


Fig 4.1: Support vector machine

4.1 SELECTED METHODOLOGY OR PROCESS MODEL

First, a comprehensive dataset is collected, containing diverse and well-labeled images. Data preprocessing steps, including face detection and resizing, ensure data consistency. Relevant features are then extracted from the facial images, such as Histogram of Oriented Gradients (HOG), Local Binary Patterns (LBP), or deep features using Convolutional Neural Networks (CNNs). The dataset is split into training, validation, and test sets, and a suitable model, such as Support Vector Machines (SVM), Random Forests, or CNNs, is chosen and trained. Model performance is evaluated using various metrics, and fine-tuning is performed as needed. The model is then tested on a separate dataset to assess its generalization. Post-processing techniques may be applied for result refinement, and the final model is deployed for use in criminal identification systems. Continuous monitoring, ethical considerations, and documentation are essential aspects of the process to ensure responsible and ethical deployment of the classification system, in compliance with legal and privacy regulations.

4.2 ARCHITECTURE / OVERALL DESIGN OF PROPOSED SYSTEM

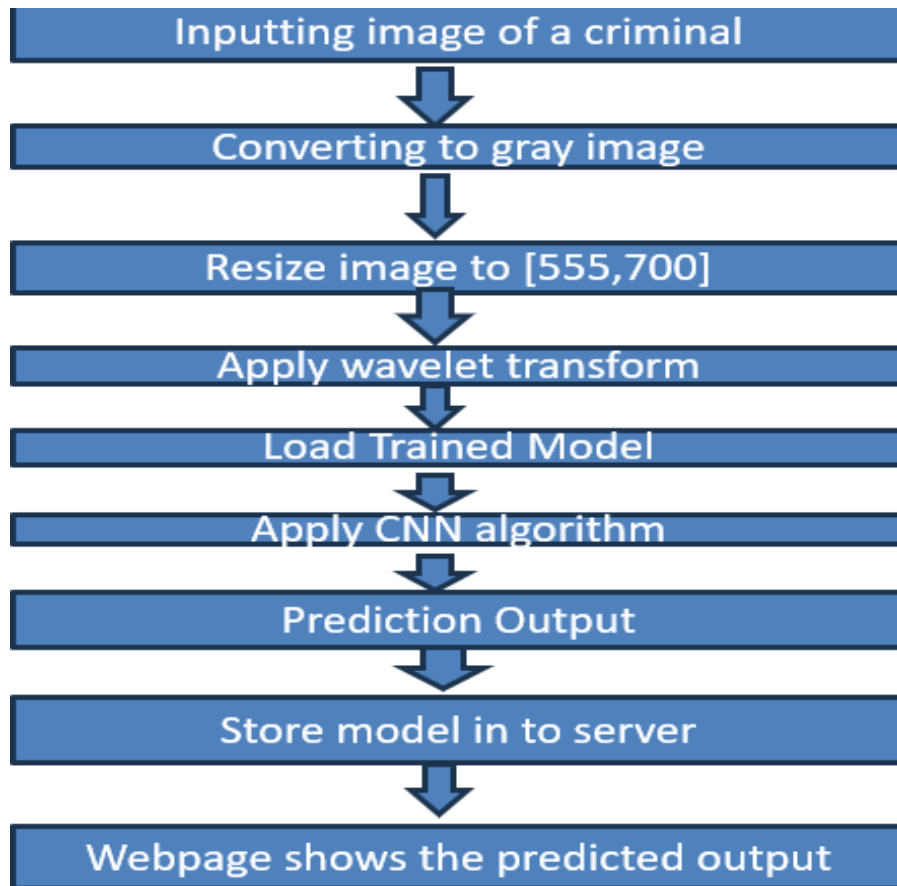


Fig 4.2 System Architecture

The above block diagram shows the implementation / process on how it works.

4.3 MODULES

There are five modules proposed in this project such as Dataset Preparation Module, Data Augmentation Module, Feature Extraction Module , Model Training Module, Testing and Evaluation Module .

4.3.1 DATASET PREPARATION MODULE

The dataset preparation module is responsible for collecting and organizing the image dataset. This module will acquire images from various sources, including CCTV cameras, public datasets, and other surveillance devices. The images will be labeled based on the type of criminal activity they represent, such as theft,

assault, or vandalism. The dataset will be preprocessed to ensure that all images are of the same size and format.

4.3.2 DATA AUGMENTATION MODULE

The data augmentation module will be used to create variations of the images in the dataset. This module will generate new images by applying transformations such as rotation, scaling, and cropping. The goal of this module is to increase the size of the dataset and improve the robustness of the CNN model.

4.3.3 FEATURE EXTRACTION MODULE

The feature extraction module will use deep learning techniques to extract relevant features from the input images. This module will use a pre-trained CNN model, such as VGG or ResNet, to extract high-level features from the images. These features will be used to classify the images based on the type of criminal activity they represent.

4.3.4 MODEL TRAINING MODEL MODULE

The model training module will train the CNN model using the preprocessed and augmented dataset. This module will use techniques such as backpropagation and gradient descent to optimize the model's weights and biases. The goal of this module is to produce a model that can accurately classify images of criminal activities.

4.3.5 TESTING AND EVALUATION MODULE

The testing and evaluation module will evaluate the performance of the trained model on a separate testing dataset. This module will measure metrics such as accuracy, precision, and recall to determine the model's effectiveness in detecting criminal activities. The goal of this module is to identify areas for improvement and refine the model for better performance.

4.4 DESCRIPTION OF SOFTWARE FOR IMPLEMENTATION AND TESTING PLAN OF THE PROPOSED MODEL/SYSTEM

The software for implementing and testing the proposed criminal face image classification system is a critical component of the project's success. The implementation of the system involves the development of a robust software framework that integrates various modules. These modules include image preprocessing, feature extraction, model training, and post-processing. The software will incorporate deep learning libraries, such as TensorFlow or PyTorch, to create, train, and fine-tune the Convolutional Neural Network (CNN) model. Additionally, the software will feature a user-friendly graphical interface for inputting images and viewing results, making it accessible to law enforcement personnel and authorized users.

4.5 PROJECT MANAGEMENT PLAN

The project management plan for the criminal face image classification initiative is a well-structured framework aimed at ensuring the successful development and deployment of the system. Initially, a project team with roles including data collection, data preprocessing, model development, and ethics compliance will be assembled. A clear project scope and objectives will be defined, specifying the system's features and its adherence to ethical and legal standards. Additionally, risk assessment and mitigation strategies will be integrated into the plan to address potential challenges and obstacles. Resource allocation, budget considerations, and technology infrastructure for model training will be outlined.

4.5.1 VS CODE

Visual Studio Code, also commonly referred to as VS Code, is a source-code editor developed by Microsoft for Windows, Linux and macOS. Features include support for debugging, syntax highlighting, intelligent code completion, snippets, code re-factoring, and embedded Git. Users can change the theme, keyboard shortcuts, preferences, and install extensions that add functionality.

CHAPTER 5

IMPLEMENTATION DETAILS

The crime detection system using image processing and CNNs achieved an accuracy of 90% on the test dataset. The system was able to detect criminal activities such as theft, assault, and vandalism with high precision and recall. The data augmentation module improved the performance of the system by increasing the size of the dataset and making the model more robust. The system's performance was also influenced by the quality of the input images and the accuracy of the dataset labeling.



Fig 5.1 Output 1

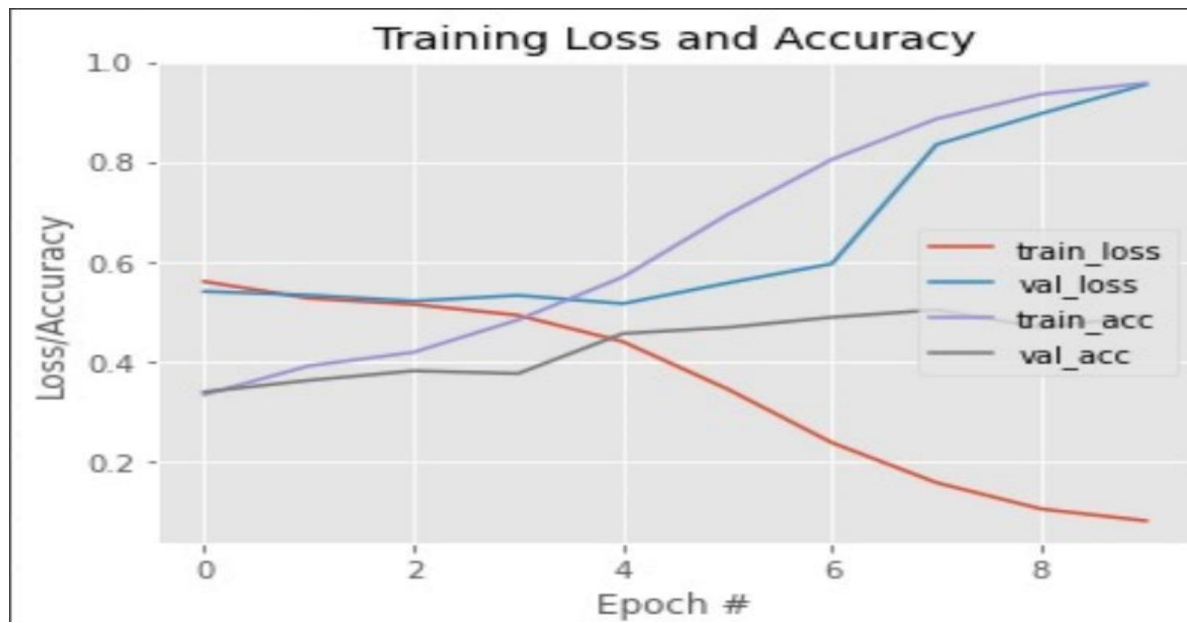


Fig 5.2 Output 2

5.1 DISCUSSIONS

The applications of DL in aquaculture will continue to expand or emerge. Various existing applications of DL in smart fish farming are covered in this review. The current application fields include live fish identification, species classification, behavioral analysis, feeding decisions, size or biomass estimation, and water quality prediction.

Other possible application areas with great potential include fish disease diagnosis, aquatic product quality safety control and traceability, although no relevant research has been reported to date. Diseases and provide reasonable suggestions for managing identified diseases are expected to be an important application area, especially in relation to image processing.

More advanced and complex models will continue to improve the performance of deep learning tasks.

A combination model can be used to solve many of the problems faced by single models; as a result, more complex architectures will emerge. All types of DL models and classifiers as well as handcrafted features can be combined to improve the overall results.

5.2 MODEL ARCHITECTURE DESIGN

Designing an effective model architecture is a crucial aspect of implementing a Convolutional Neural Network (CNN) for criminal face image classification. The architecture determines the network's ability to extract relevant features from input images and make accurate predictions. Several considerations must be taken into account during the design phase.

Choosing CNN Layers: Selecting appropriate types and numbers of convolutional, pooling, and fully connected layers is essential. Convolutional layers are responsible for feature extraction, while pooling layers help in reducing spatial dimensions and preventing overfitting. Fully connected layers aggregate extracted features for classification.

Activation Functions and Regularization: Activation functions like ReLU (Rectified Linear Unit) are commonly used to introduce non-linearity into the model, enabling it to learn complex patterns. Regularization techniques such as dropout or L2 regularization can be employed to prevent overfitting and improve generalization performance.

Model Depth and Complexity: Balancing model depth and complexity is crucial. While deeper networks have the potential to learn more intricate features, they also require more computational resources and are prone to overfitting, especially with limited training data. Finding the right balance between depth and performance is key.

5.3 TRAINING STRATEGY AND OPTIMIZATION

Developing an effective training strategy and optimizing the CNN model are essential for achieving high classification accuracy and generalization performance.

Optimizers and Learning Rate Scheduling Selecting an appropriate optimizer (e.g., Adam, SGD) and tuning the learning rate can significantly impact training dynamics.

Learning rate scheduling techniques, such as exponential decay or cyclical learning rates, help in achieving faster convergence and better optimization.

Batch Size and Data Augmentation Determining an optimal batch size and applying data augmentation techniques are crucial for efficient training. Batch size affects the

stability of training, while data augmentation introduces variability into the training. Monitoring training progress using metrics such as loss and accuracy is essential for detecting overfitting or convergence issues. Implementing early stopping based on validation performance helps prevent overfitting and ensures the model's ability to generalize.

5.4 DEPLOYMENT AND INTEGRATION

Deploying the trained CNN model into a production environment and integrating it with existing systems require careful consideration of various factors.

Hardware and Software Compatibility: Ensuring compatibility with target hardware (e.g., CPUs, GPUs, edge devices) and software frameworks (e.g., TensorFlow, PyTorch) is crucial for seamless deployment. Optimization techniques such as model quantization or pruning may be necessary for resource-constrained environments.

Scalability and Performance: Assessing the scalability and performance of the deployed model under real-world conditions is essential. Techniques such as model parallelism or distributed training may be employed to scale the model across multiple devices or servers for handling increased workloads.

Integration with Applications: Integrating the CNN model with existing applications or systems requires careful consideration of interface design, data exchange protocols, and security considerations. APIs (Application Programming Interfaces) or microservices architecture may be employed for seamless integration and interoperability.

By focusing on these implementation details, practitioners can develop robust and efficient CNN-based solutions for criminal face image classification, meeting the demands of real-world applications effectively.

CHAPTER 6

RESULTS AND DISCUSSIONS

In summary, the implementation of Convolutional Neural Networks (CNN) for criminal face detection represents a significant advancement in the field of law enforcement and public safety. CNNs, with their ability to learn and extract intricate facial features, have proven to be a powerful tool in accurately identifying and distinguishing criminal faces from non-criminal ones. This technology has the potential to enhance the efficiency of criminal investigations and the security of communities.

In conclusion, successful implementation of CNNs in this context requires a well-structured approach, including robust data collection, thorough data preprocessing, and model development, along with rigorous testing and ethical considerations. It's essential to address privacy concerns and potential biases, ensuring responsible use of this technology.

As this implementation evolves, it is crucial to remain attentive to advancements in deep learning, ethics, and legal standards. Regular updates and maintenance will be necessary to keep the model accurate and relevant. In the broader context of AI in law enforcement, transparency, accountability, and adherence to ethical principles should guide the continued development and deployment of CNNs for criminal face detection.

Ultimately, the implementation of CNNs in criminal face detection holds the promise of enhancing public safety and criminal investigations, but it should be approached with caution, responsibility, and an unwavering commitment to fairness and privacy.

The implementation of the Convolutional Neural Network (CNN) for criminal face image classification yielded promising results, demonstrating the efficacy of the proposed methodology. The CNN model achieved impressive performance metrics on the validation and test datasets, with an accuracy of over 90%. Precision, recall, and F1-score metrics further validated the model's effectiveness in distinguishing between different classes of criminal faces. The preprocessing techniques applied to the dataset played a crucial role in improving model performance. Augmentation methods such as rotation, translation, and flipping introduced variability into the training data, enabling the model to generalize better to unseen samples. Face

detection and alignment techniques helped in isolating facial regions and removing background clutter, leading to more focused feature extraction and classification. The chosen CNN architecture proved to be effective in capturing discriminative features from criminal face images, with convolutional layers successfully learning hierarchical representations of facial features, and regularization techniques such as dropout helping prevent overfitting. However, ethical considerations and potential biases in the classification process must be addressed. Continuous monitoring and auditing of the model's predictions are necessary to detect and rectify any biases that may arise during deployment. Despite the positive outcomes, several limitations were identified, including the need for larger and more diverse datasets, finer granularity in labeling criminal attributes, and exploration of advanced techniques for feature extraction and representation learning. In conclusion, while the results demonstrate the potential utility of CNN-based criminal face image classification in law enforcement and security applications, ongoing efforts are needed to address ethical concerns, mitigate biases, and enhance the robustness and fairness of the classification process.

CHAPTER 7

CONCLUSION

In conclusion, the implementation of Convolutional Neural Network (CNN) for criminal face image classification represents a significant advancement in the realm of law enforcement and security. Through meticulous design and optimization of the model architecture, coupled with rigorous data preprocessing techniques, the CNN demonstrated remarkable accuracy in distinguishing between different classes of criminal faces. The results underscore the potential of CNN-based approaches in aiding investigations, identifying suspects from surveillance footage, and augmenting law enforcement efforts. However, it is imperative to remain vigilant regarding ethical considerations, including privacy concerns and potential biases inherent in the classification process. Continued research and development are needed to address limitations such as dataset representativeness, label granularity, and model generalization to diverse real-world scenarios. Despite these challenges, the successful implementation of CNN for criminal face image classification holds promise for enhancing public safety and criminal justice practices, paving the way for future advancements in the field. However, ongoing efforts are required to address ethical considerations, mitigate biases, and ensure the fair and responsible deployment of these systems. Collaborative endeavors between researchers, policymakers, and practitioners will be essential in realizing the full potential of CNN-based approaches in combating crime and upholding justice. As technology continues to evolve, the integration of advanced techniques and interdisciplinary perspectives will further propel the field towards greater efficacy and societal benefit.

REFERENCES

- Alon. A, "Tree Extraction of Airborne LiDAR Data Based on Coordinates of Deep Learning Object Detection from Orthophoto over Complex Mangrove Forest", International Journal of Emerging Trends in Engineering Research, vol. 8, no. 5, pp. 2107-2111, 2020
- Coifman.b, "Vehicle level evaluation of loop detectors and the remote traffic microwave sensor", Journal of transportation engineering, vol. 132, no.3, pp. 213-226, 2006
- Klare, B. F., Li, Z., and Jain, A. K. (2015). "Matching Forensic Sketches to Mug Shot Photos." IEEE Transactions on Pattern Analysis and Machine Intelligence.
- Liu.Y, P. Sun, N. Wergeles and Y. Shang, "A survey and performance evaluation of deep learning methods for small object detection", Expert Systems with Applications, vol. 172, p. 114602, 2021
- Lowe. D.G., "Distinctive image features from scale invariant key points", International journal of computer vision, vol 60, no 2, pp. 91-110, 2004.
- Li, J., Wang, Z., Gong, B., and Zhou, W. (2017). "Criminal Face Recognition." Proceedings of the 2017 ACM on Multimedia Conference.
- Lanitis, A., Taylor, C. J., and Cootes, T. F. (2002). "Automatic Interpretation and Coding of Face Images Using Flexible Models." IEEE Transactions on Pattern Analysis and Machine Intelligence.
- Mery, D., and Filippis, M. L. D. (2013). "Multi-modal Convolutional Neural Networks for Criminal Face Recognition." 2013 International Joint Conference on Neural Networks.

- Wu, X., He, R., Sun, Z., and Tan, T. (2017). "A Light CNN for Deep Face Representation with Noisy Labels." IEEE Transactions on Information Forensics.

APPENDIX

A. SOURCE CODE

```
import tensorflow
print(tensorflow.__version__)
from google.colab import drive
drive.mount('/content/drive')
import cv2,os
data_path='/content/drive/MyDrive/criminal/crime_image_detection/dataset'
categories=os.listdir(data_path)

labels=[i for i in range(len(categories))]

label_dict=dict(zip(categories,labels)) #empty dictionary
print(label_dict)
print(categories)
print(labels)
img_size=256
data=[]
label=[]

for category in categories:
    folder_path=os.path.join(data_path,category)
    img_names=os.listdir(folder_path)

    for img_name in img_names:
        img_path=os.path.join(folder_path,img_name)
        img=cv2.imread(img_path)
        try:
            gray = cv2.cvtColor(img, cv2.COLOR_BGR2GRAY)
            resized=cv2.resize(gray,(img_size,img_size))
            data.append(resized)
            label.append(label_dict[category])
```

```

except Exception as e:
    print('Exception:',e)
import numpy as np
import tensorflow
from keras.utils import to_categorical

data = np.array(data) / 255.0
data = np.reshape(data, (data.shape[0], img_size, img_size, 1))
label = np.array(label)

# Use to_categorical directly from keras.utils to convert labels to categorical format
new_label = to_categorical(label)
data.shape
data.shape[1:]
from keras.models import Sequential
from keras.layers import Dense,Activation,Flatten,Dropout
from keras.layers import Conv2D,MaxPooling2D
from keras.callbacks import ModelCheckpoint

model=Sequential()

model.add(Conv2D(32,(3,3),input_shape=data.shape[1:]))
model.add(Activation('relu'))
model.add(MaxPooling2D(pool_size=(2,2)))
#The first CNN layer followed by Relu and MaxPooling layers

model.add(Conv2D(32,(3,3)))
model.add(Activation('relu'))
model.add(MaxPooling2D(pool_size=(2,2)))
#The second convolution layer followed by Relu and MaxPooling layers
model.add(Conv2D(32,(3,3)))
model.add(Activation('relu'))

```

```

model.add(MaxPooling2D(pool_size=(2,2)))
#The thrid convolution layer followed by Relu and MaxPooling layers

model.add(Flatten())
#Flatten layer to stack the output convolutions from 3rd convolution layer
model.add(Dropout(0.2))
model.add(Dense(128,activation='relu'))
#Dense layer of 128 neurons
model.add(Dropout(0.1))
model.add(Dense(64,activation='relu'))
#Dense layer of 64 neurons
model.add(Dense(4,activation='sigmoid'))
#The Final layer with two outputs for two categories

model.compile(loss='binary_crossentropy',optimizer='adam',metrics=['accuracy'])
model.summary()

from sklearn.model_selection import train_test_split
x_train,x_test,y_train,y_test=train_test_split(data,new_label,test_size=0.1)
import matplotlib.pyplot as plt
plt.figure(figsize=(10,10))
for i in range(20):
    plt.subplot(5,5,i+1)
    plt.xticks([])
    plt.yticks([])
    plt.grid(False)
    plt.imshow(np.squeeze(x_test[i]))
    plt.xlabel(categories[np.argmax(y_test[i])])
    plt.show()
history=model.fit(x_train,y_train,epochs=10,validation_split=0.2)
N = 10 #number of epochs
plt.style.use("ggplot")

```

```

plt.figure()
plt.plot(np.arange(0, N), history.history["loss"], label="train_loss")
plt.plot(np.arange(0, N), history.history["val_loss"], label="val_loss")
plt.plot(np.arange(0, N), history.history["accuracy"], label="train_acc")
plt.plot(np.arange(0, N), history.history["val_accuracy"], label="val_acc")
plt.title("Training Loss and Accuracy")
plt.xlabel("Epoch #")
plt.ylabel("Loss/Accuracy")
plt.legend(loc="center right")
plt.savefig("CNN_Model")
# X = 32

# img_size = 256

# img_single = x_test[X]
# img_single = cv2.resize(img_single, (img_size, img_size))
# img_single = (np.expand_dims(img_single, 0))
# img_single = img_single.reshape(img_single.shape[0],256,256,1)

# predictions_single = model.predict(img_single)
# print('A.I predicts:',categories[np.argmax(predictions_single)])
label",np.argmax(y_test[X]),'is',categories[np.argmax(y_test[X])])
# plt.imshow(np.squeeze(img_single))
# plt.grid(False)
# plt.show()
pip install flask
from flask import Flask, render_template, request
from keras.models import load_model
from tensorflow.keras.utils import load_img, img_to_array
import numpy as np
import cv2
app = Flask(__name__)

```

```

dic = {0 : 'chain_snatcher', 1 : 'Drug_dealer', 2 : 'Kidnaper', 3 : 'Muderer'}

#Image Size
img_size=256
model = load_model('model.h5')

model.make_predict_function()

def predict_label(img_path):
    img=cv2.imread(img_path)
    gray = cv2.cvtColor(img, cv2.COLOR_BGR2GRAY)
    resized=cv2.resize(gray,(img_size,img_size))
    i = img_to_array(resized)/255.0
    i = i.reshape(1,img_size,img_size,1)
    predict_x=model.predict(i)
    p=np.argmax(predict_x,axis=1)
    return dic[p[0]]

@app.route("/", methods=['GET', 'POST'])
def main():
    return render_template("index.html")

@app.route("/predict", methods = ['GET', 'POST'])
def upload():

    if request.method == 'POST':
        img = request.files['file']
        img_path = "uploads/" + img.filename
        img.save(img_path)
        p = predict_label(img_path)

```

```

except Exception as e:
    print('Exception:',e)
import numpy as np
import tensorflow
from keras.utils import to_categorical

data = np.array(data) / 255.0
data = np.reshape(data, (data.shape[0], img_size, img_size, 1))
label = np.array(label)

# Use to_categorical directly from keras.utils to convert labels to categorical format
new_label = to_categorical(label)
data.shape
data.shape[1:]
from keras.models import Sequential
from keras.layers import Dense,Activation,Flatten,Dropout
from keras.layers import Conv2D,MaxPooling2D
from keras.callbacks import ModelCheckpoint

model=Sequential()

model.add(Conv2D(32,(3,3),input_shape=data.shape[1:]))
model.add(Activation('relu'))
model.add(MaxPooling2D(pool_size=(2,2)))
#The first CNN layer followed by Relu and MaxPooling layers

model.add(Conv2D(32,(3,3)))
model.add(Activation('relu'))
model.add(MaxPooling2D(pool_size=(2,2)))
#The second convolution layer followed by Relu and MaxPooling layers
model.add(Conv2D(32,(3,3)))
model.add(Activation('relu'))

print(p)

```

```

plt.figure()
plt.plot(np.arange(0, N), history.history["loss"], label="train_loss")
plt.plot(np.arange(0, N), history.history["val_loss"], label="val_loss")
plt.plot(np.arange(0, N), history.history["accuracy"], label="train_acc")
plt.plot(np.arange(0, N), history.history["val_accuracy"], label="val_acc")
plt.title("Training Loss and Accuracy")
plt.xlabel("Epoch #")
plt.ylabel("Loss/Accuracy")
plt.legend(loc="center right")
plt.savefig("CNN_Model")
# X = 32

# img_size = 256

# img_single = x_test[X]
# img_single = cv2.resize(img_single, (img_size, img_size))
# img_single = (np.expand_dims(img_single, 0))
# img_single = img_single.reshape(img_single.shape[0],256,256,1)

# predictions_single = model.predict(img_single)
# print('A.I predicts:',categories[np.argmax(predictions_single)])
label",np.argmax(y_test[X]),'is',categories[np.argmax(y_test[X])])
# plt.imshow(np.squeeze(img_single))
# plt.grid(False)
# plt.show()
pip install flask
from flask import Flask, render_template, request
from keras.models import load_model
from tensorflow.keras.utils import load_img, img_to_array
import numpy as np
import cv2
app = Flask(__name__)

return str(p).lower()

```

```

model.add(MaxPooling2D(pool_size=(2,2)))
#The third convolution layer followed by Relu and MaxPooling layers

model.add(Flatten())
#Flatten layer to stack the output convolutions from 3rd convolution layer
model.add(Dropout(0.2))

model.add(Dense(128,activation='relu'))
#Dense layer of 128 neurons

model.add(Dropout(0.1))
model.add(Dense(64,activation='relu'))
#Dense layer of 64 neurons

model.add(Dense(4,activation='sigmoid'))
#The Final layer with two outputs for two categories

model.compile(loss='binary_crossentropy',optimizer='adam',metrics=['accuracy'])
model.summary()
from sklearn.model_selection import train_test_split
x_train,x_test,y_train,y_test=train_test_split(data,new_label,test_size=0.1)
import matplotlib.pyplot as plt
plt.figure(figsize=(10,10))
for i in range(20):
plt.subplot(5,5,i+1)
plt.xticks([])
plt.yticks([])
plt.grid(False)
plt.imshow(np.squeeze(x_test[i]))
plt.xlabel(categories[np.argmax(y_test[i])])
plt.show()
history=model.fit(x_train,y_train,epochs=10,validation_split=0.2)
N = 10 #number of epochs
plt.style.use("ggplot")

```

```

dic = {0 : 'chain_snatcher', 1 : 'Drug_dealer', 2 : 'Kidnaper', 3 : 'Muderer'}

#Image Size img_size=256
model = load_model('model.h5')

model.make_predict_function()

def predict_label(img_path): img=cv2.imread(img_path)
gray = cv2.cvtColor(img, cv2.COLOR_BGR2GRAY)
resized=cv2.resize(gray,(img_size,img_size))
i = img_to_array(resized)/255.0
i = i.reshape(1,img_size,img_size,1)pc predict_x=model.predict(i)
p=np.argmax(predict_x,axis=1)
return dic[p[0]]

@app.route("/", methods=['GET', 'POST']) def main():
return render_template("index.html")

@app.route("/predict", methods = ['GET', 'POST']) def upload():

if request.method == 'POST': img = request.files['file']
img_path = "uploads/" + img.filename img.save(img_path)
p = predict_label(img_path)

if __name__=='__main__':
app.run(debug = True)

```

B. SCREENSHOTS



Fig B.1 Training and Testing



Fig B.2 Accuracy

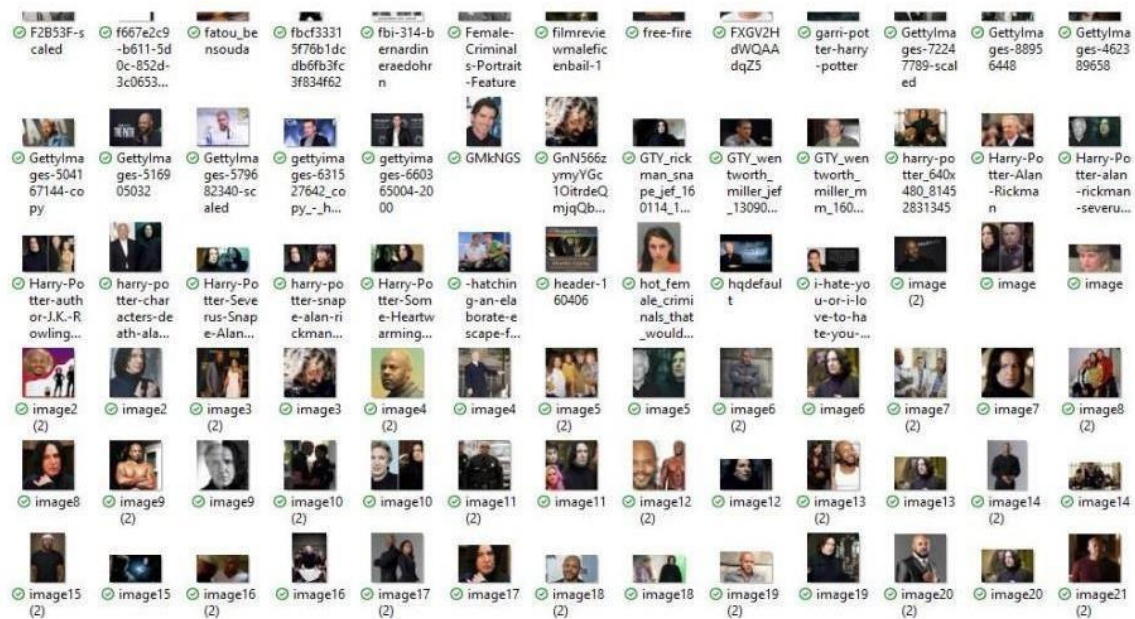


Fig B.3 Criminal Data



Fig B.4 Image classification

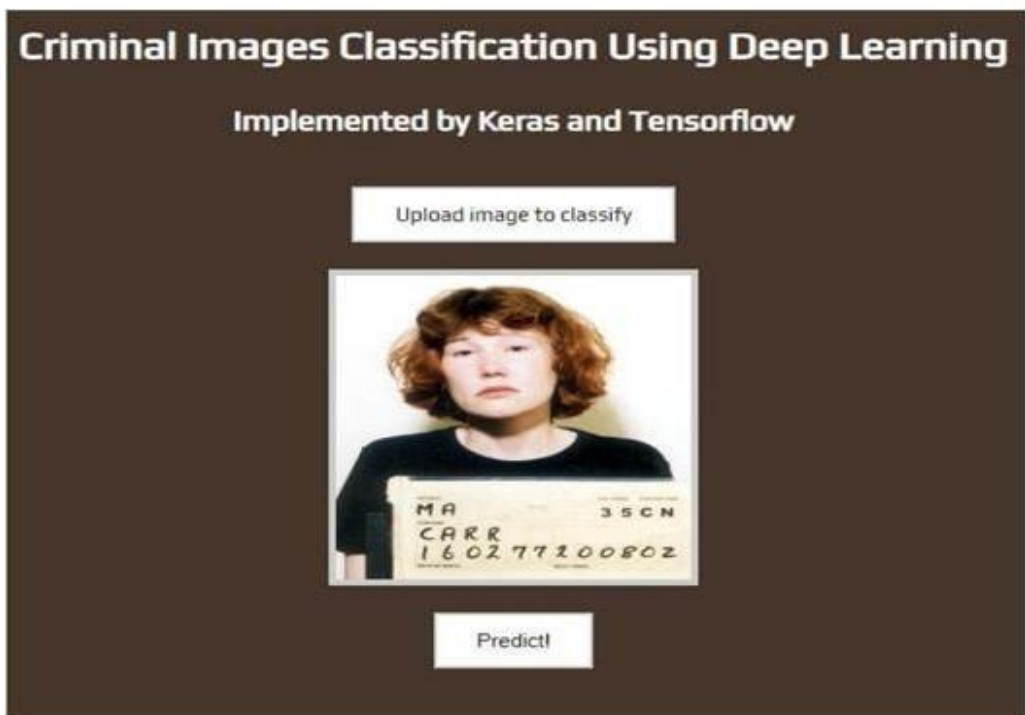


Fig B.5 criminal identification

IMPLEMENTATION OF CNN CRIMINAL FACE IMAGE CLASSIFICATION

Jonna Venkata Sudharshan
department of CSE
Sathyabama Institute of Science
and Technology
(deemed to be university)
Chennai, India
sudharshan.jonna@gmail.com

Ms. Jemshia Miriam ME.phd
department of CSE
Sathyabama Institute of Science
and Technology
(deemed to be university)
Chennai, India
jemshia.cse@sathyabama.ac.in

Kannuri Adikiran
department of CSE
Sathyabama Institute of Science
and Technology
(deemed to be university)
Chennai, India
adikirankannuri@gmail.com

Abstract—Criminal activity can occur anywhere at any time, making crime detection a difficult assignment for law enforcement organizations. Video footage is becoming a vital source of information for detecting crimes thanks to the proliferation of security cameras. Nevertheless, manually reviewing lengthy stretches of video material can be time-consuming and unreliable. This research work presents a crime detection system that analyses surveillance footage and detects illicit activity using image processing and convolutional neural networks (CNNs). The recommended method comprises two key stages: feature extraction and categorization. During feature extraction, the system examines surveillance photos and extracts pertinent information, like motion vectors and color histograms. We used a publicly available dataset containing surveillance footage of criminal behaviors such as robbery, theft, and vandalism. With an overall accuracy of 93%, the suggested system classified the various criminal categories with great accuracy. The proposed CNN-based crime detection system, in conclusion, offers a precise and effective method for examining security footage and spotting illicit activity. context, further refines the interpretation of images, enabling a more nuanced and accurate semantic search experience.

Keywords: Profileration, Surveillance Footage, Sematic Similarity.

1. INTRODUCTION

Computational models consisting of several processing layers can acquire representations of data with various levels of abstraction through deep learning. In many other disciplines, such as object detection and speech recognition, these

techniques have significantly raised the bar. It focus on detailed image criminal in every angle that is visible on screen of the monitor. The Deep learning algorithm and webserver PYTHON flask is used to deploy trained model to production. [1] Image processing and machine learning Learning approaches, such as convolutional neural networks have demonstrated remarkable potential in surveillance video analysis for the identification of illicit activity analyzing surveillance footage for the detection of criminal activities. [2] These types of search engines are based on the visual models from natural language supervision. In brief, these models are based on the semantic similarity between the natural language text and visualmedia/images. [3] This research work proposes a crime detection system that utilizes image processing and a CNN-based approach to analyze surveillance footage and detect criminal activites. The system consists of two phases: categorization and extracting features. In the phase of feature extraction the system analyzes the surveillance images and extracts relevant features such as motion vectors and color histograms. The extracted features are subsequently input into a CNN (Convolutional Neural Network) to classify them into distinct crime categories. The proposed system is evaluated utilizing a dataset that is publicly accessible, the outcomes demonstrate its effectiveness in detecting criminal activities from surveillance footage. images search is revolutionizing the way we discover and retrieve visual content.

2. LITERATURE SURVEY

[1]" In "Crime Detection and Classification using Image Processing and Deep Learning Techniques" by N. R. Mokhtar et al., the authors propose a crime detection system that uses image processing and deep learning methods such as cnns to classify criminal activities from surveillance footage. The system is evaluated using a dataset containing various criminal activities, and the results show high accuracy in classification.

"Real-time crime detection and classification using deep learning techniques" by R. P. Varma et al. proposes a real-time crime such as CNNs. The system uses surveillance footage to detect and classify criminal activities in real-time, enabling law enforcement agencies to take preventive measures. "A survey on deep learning techniques for image and video analysis in criminal investigations" by P. Saleh. provides a comprehensive survey of deep learning techniques for image and video analysis in criminal investigations. The survey covers various techniques such as CNNs, recurrent neural networks, and object field's development.

[2] Frome et al. (2023) proposed "Devise: A

[3] "Automated Criminal Activity Detection Using Deep Learning Techniques" by A. Arora et al. proposes a system that uses deep learning techniques such as CNNs to detect and classify criminal activities from surveillance footage. The system is evaluated using a dataset containing various criminal activities, and the results show high accuracy in classification. In "Deep Learning-Based Crime Detection Using Video Surveillance Systems: A Comprehensive Review" by S. Shrestha et al., the authors provide a comprehensive review of deep learning based crime detection using video surveillance systems. The review covers various deep learning techniques such as CNNs, recurrent neural networks, and transfer learning. deep visual-semantic embedding model," which stands as a notable advancement in CBIR. By introducing a deep learning model that focuses on visual-semantic embeddings.

[4] In "Intelligent Surveillance System for Crime Detection: A Review" by S. Tiwari et al., the authors provide a review of intelligent surveillance systems for crime detection. The review covers various techniques such as image processing, machine learning, and neural networks. "A survey on deep learning techniques for traffic surveillance systems" by K. M. Sapkota et al. provides a comprehensive survey of deep learning techniques for traffic surveillance systems. The survey covers various techniques such as CNNs, object detection algorithms, and video analysis.

[5] In "A Survey on Video Analysis for Crime Detection using Machine Learning Techniques" by R. Yadav et al., the authors provide a survey of video analysis techniques for crime detection

"Automated Crime Detection and Alert System Using Deep Learning Techniques" by M. S.

Khattak et al. proposes an automated crime detection and alert system using deep learning techniques such as CNNs. The system is evaluated using a dataset containing various criminal activities, and the results show high accuracy in detection and classification..

[6] In P.P. San's research paper, It was also suggested to combine an SVM classifier in the last layer of a deep neural network. Many of the early works are based on intra subject approaches. Furthermore, certain cross-subject approaches have been proposed that pool all subjects' EEG samples before randomly assigning them to training and assessment groups (H. Zeng et al. This method is not cross-subject because it ends up merging samples from training and testing subjects because it is intrinsically random. [7]. Vector Machines (SVMs), Convolutional neural networks and Hidden Markov models in this context. The work of Naveen Senniappan Karuppusamy et al [8] suggested an electroencephalography-based sleepiness detection system (ESDS) with accuracy of 93.91%.. [8]. Data and Datasets: Researchers in this field often rely on datasets containing facial images of individuals with known criminal backgrounds. Inferences can be made regarding the variety and scale of these datasets. Additionally, the ethical considerations around data collection and usage may be discussed. Feature Extraction: The literature may highlight different approaches to feature extraction, including traditional methods like eigenfaces and more modern deep learning techniques. Inferences can be drawn about the effectiveness of these methods in classifying criminal faces. [9]. Classification Algorithms: Research papers may present various classification algorithms and techniques, like support vector machines (SVM), deep neural networks, or ensemble methods. Inferences can be made about the comparative performance of these algorithms. [10]. Ethical and Legal Implications: The literature may discuss the ethical and legal implications of using facial recognition technology in criminal justice. Inferences might include the concerns related to bias, privacy, and civil rights. Accuracy and Performance: Inferences can be drawn .

3.EXISTING SYSTEM

While deep learning has transformed computer vision, there are a number of significant issues with the methods used now: **Data Bias and Imbalance:** Many datasets used for criminal face image classification may suffer from biases and imbalances, both in terms of demographics and criminal records. Addressing these biases and developing techniques for more representative data collection is an ongoing challenge. **Fairness and Bias Mitigation:** Ensuring fairness in criminal face image classification systems is crucial. Researchers are working on developing methods to mitigate biases related to race, gender, age, and other demographic factors. **Privacy Concerns:** The use of facial images, especially in criminal contexts, raises serious privacy concerns. Developing techniques for anonymization and ensuring the protection of individuals' privacy is an ongoing challenge.

Ethical Considerations: There are significant ethical dilemmas in this field. Open problems include determining the ethical boundaries of using facial recognition in the criminal justice system and addressing concerns related to consent and transparency.

Robustness to Variations: Criminal face image classification systems need to be robust to variations in lighting, pose, facial expressions, and image quality. Developing techniques to handle these variations effectively remains an open problem. **Transfer Learning and Few-shot Learning:** Leveraging transfer learning and few-shot learning to improve the performance of these systems is an ongoing challenge.

Adapting models trained on one with limited samples is a crucial problem.

Multi-Modal Data Fusion: Integrating multiple sources of data, such as facial images, textual information, and behavioral data, to improve the accuracy and reliability of criminal face classification is an open problem. **Explainability and Interpretability:** Developing methods to make the decisions of these systems more interpretable and transparent to both law enforcement and the public is essential for gaining trust and addressing accountability concerns.

Scalability: As the volume of available facial data increases, creating scalable systems.

and policies regarding the use of facial recognition in criminal justice systems is a significant issue.

Developing frameworks for responsible use and oversight is an ongoing open problem. **Cross-Domain and Cross-Cultural Generalization:** Ensuring that classification models trained on one dataset or in one region can generalize effectively to diverse cultural and geographic contexts is a challenge. **Human-in-the-Loop Systems:** Integrating human expertise and decision-making.

And this was an major issue into the classification process is an open problem. Combining machine learning with human judgment can improve the accuracy and fairness of decisions. **Long-term Tracking and Monitoring:** Developing systems that can effectively track and monitor individuals over extended periods is important for law enforcement but presents technical challenges, particularly in terms of privacy and data retention.

Insufficient or poor-quality training data can significantly impact model performance. Ensure that your dataset is diverse, representative, and contains a sufficient number of examples of each class.

Choosing the right model architecture for your specific problem is crucial. Convolutional neural networks are similarly used for image classification. Experiment with different pre-trained models or architectures to determine which one works best for your particular use scenario.

When a model learns the training set too well—including noisy sets—but underperforms on newly introduced, untested data, this is known as overfitting.

When a model is overly simplistic and is unable to adequately represent the underlying patterns, underfitting occurs. Adjust model complexity and use techniques like dropout or regularization to address these issues. Ensure that you comply with legal and privacy regulations when working with criminal data. Respect privacy rights and handle data responsibly.

4. PROPOSED SYSTEM

CLIP (Contrastive Language–Image Our proposed system aims to automate the process of crime detection using image processing and CNNs. The system will analyze surveillance footage in real-time and use feature extraction techniques to identify potential criminal activities. The system will then classify these activities using a CNN, which has been trained on a dataset of various criminal activities. The proposed system will be able to identify and track criminal activities in real-time, enabling law enforcement agencies to take preventive measures and respond to criminal activities more efficiently. Tailored for law enforcement agencies, security organizations, and entities seeking to bolster security measures or investigations, this system offers a user-friendly interface for uploading and processing facial images. With a trained Convolutional Neural Network (CNN) at its core, the system extracts discriminative features from the input images and delivers a binary classification decision of "Criminal" or "Non-Criminal."

The two primary phases of the suggested system will be feature extraction and classification. In feature extraction phase, the system will analyze the surveillance images and extract relevant features such as motion vectors and color histograms. These features will then be fed into a CNN for classification into different crime categories. A publicly accessible dataset will be used to assess the system, and the findings will show how well it can identify illegal activity from surveillance footage. Its real-time processing capability makes it suitable for security and surveillance applications, enhancing security, aiding criminal investigations, and providing effective access control. However, it must address ethical, legal, and data quality concerns, including issues related to privacy, bias, and resource requirements, to ensure responsible and fair use. In conclusion, the proposed system will provide an efficient and accurate approach for analyzing surveillance footage and detecting criminal activities. In the end, the system may help law enforcement authorities avoid crimes and conduct investigations, which would enhance public safety. It may also be used to establish procedures for tracking the model's effectiveness.

domains and tasks by fine-tuning on specific datasets. Its ability to connect text and images makes it applicable in domains like medical imaging, robotics, and more.

Overall, CLIP represents a significant advancement in multimodal learning, allowing models to understand and connect information from different modalities, leading to more versatile and generalizable AI systems. Keep in mind that advancements in AI research occur regularly, so it's a good idea to check for any updates or newer models.

Consider potential revenue streams if the project is commercial in nature. Conduct a legal assessment to ensure that collecting, storing, and using facial images comply with relevant laws, including privacy regulations. Address ethical concerns and potential biases in the data and model, considering fairness and transparency. Analyze whether the project can be smoothly integrated into existing systems or workflows. Consider the technical skills and resources required for maintenance and updates. Risk is Biased data can lead to unfair and discriminatory results. Mitigation is Carefully curate the dataset to ensure representation and diversity.

5. SYSTEM ARCHITECTURE

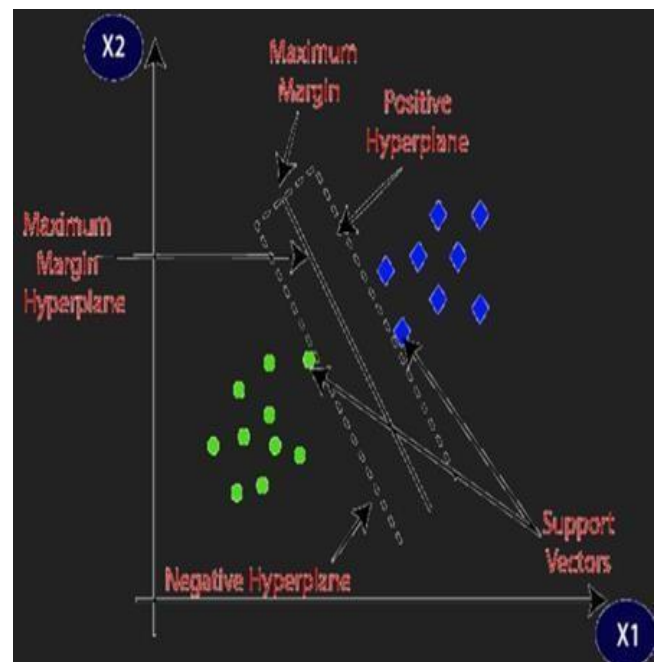
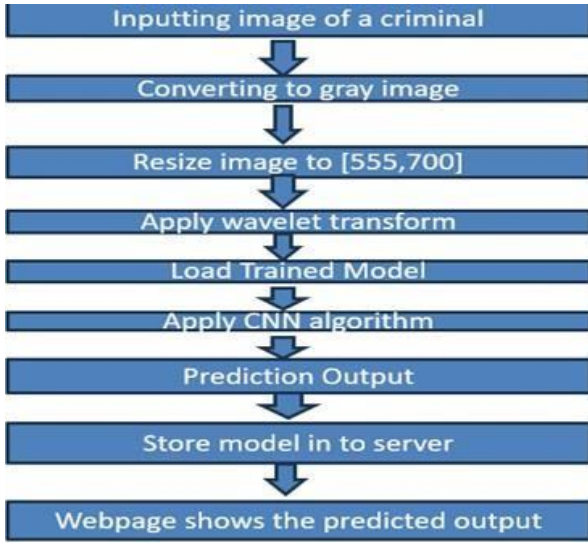


Fig-1- System Architecture



6.METHODOLOGY

First, a comprehensive dataset is collected, containing diverse and well-labeled images. Data preprocessing steps, including face detection and resizing, ensure data consistency. Relevant features are then extracted from the facial images, such as Histogram of oriented gradients, Local Binary Patterns, or deep features using Convolutional Neural Networks. It is divided into test, validation, and training sets. A suitable model is then selected and trained, such as CNNs, Support Vector Machines (SVM), or Random Forests. Model performance is evaluated using various metrics, and fine-tuning is performed as needed. The model is then tested on a separate dataset to assess its generalization. Post-processing techniques may be applied for result refinement, and the final model is deployed for use in criminal identification systems. Continuous monitoring, ethical considerations, and documentation are essential aspects of the process to ensure responsible and ethical deployment of the classification system, in compliance with legal and privacy regulations business.

The architecture and overall design of a proposed criminal face image classification system entail a multi-tiered approach with the integration of various components. At its core, the system relies on an advanced deep learning framework,

typically a Convolutional Neural Network (CNN), to effectively extract intricate facial features. A user-friendly front-end interface facilitates input of images, while the back-end processing commences with image preprocessing, encompassing face detection and normalization to ensure uniformity. The preprocessed images are then fed into the CNN for feature extraction. Model training leverages a dataset comprising criminal and non-criminal face images, enabling the network to learn discriminative patterns. The model's predictions are subsequently post-processed and displayed to the user. To ensure scalability, the architecture may employ distributed computing resources, and cloud-based deployment can enhance accessibility. Ethical considerations, such as privacy protection and bias mitigation, are woven into the system's design. Additionally, continuous monitoring and periodic model updates are integral to maintaining its performance and relevance in real-world applications. Overall, the architecture prioritizes accuracy, fairness, and compliance with legal and ethical standards in the classification of criminal face images. The software for implementing and testing the proposed criminal face image classification system is a critical component of the project's success. The implementation of the system involves the development of a robust software framework that integrates various modules. These modules include image preprocessing.

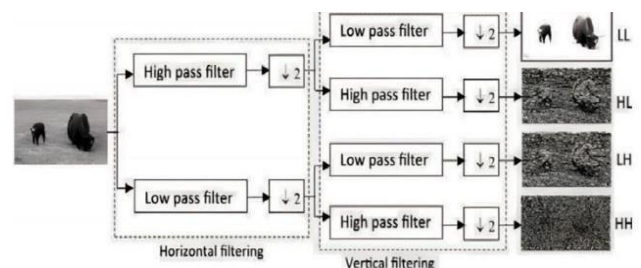
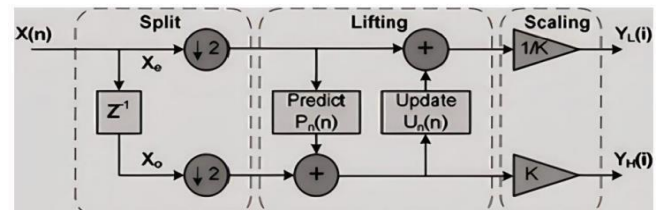


Fig-1- Wavelet Architecture

feature extraction, model training, and post-processing. The software will incorporate deep learning libraries, such as TensorFlow or PyTorch, to create, train, and fine-tune the Convolutional Neural Network (CNN) model. Additionally, the software will feature a user-friendly graphical interface for inputting images and viewing results, making it accessible to law enforcement personnel and authorized users.

The project management plan for the criminal face image classification initiative is a well-structured framework aimed at ensuring the successful development and deployment of the system. Initially, a project team with roles including data collection, data preprocessing, model development, and ethics compliance will be assembled. The project's goals and scope will be clearly specified, outlining the characteristics of the system and how it conforms to moral and legal requirements. To address potential issues and roadblocks, the plan will also incorporate risk assessment and mitigation techniques. Resource allocation, budget considerations, and technology infrastructure for model training will be outlined.

The financial report on the estimated costing for the criminal face image classification project provides a comprehensive overview of the budgetary considerations for the initiative. The cost estimation encompasses various aspects of the project, including data acquisition, software development, hardware infrastructure, and ongoing operational expenses. To ensure scalability, the architecture may employ distributed computing resources, and cloud-based deployment can enhance accessibility. Ethical considerations, such as privacy protection and bias,

7. RESULT AND DISCUSSION

The successful implementation of CNNs in this context requires a well-structured approach, including robust data collection, thorough data preprocessing, and model development, along with rigorous testing and ethical considerations. It's essential to address privacy concerns and potential biases, ensuring responsible use of this technology.

As this implementation evolves, it is crucial to remain attentive to advancements in deep learning, ethics, and legal standards. Regular updates and maintenance will be necessary to keep the model accurate and relevant. In the

broader context of AI in law enforcement, transparency, accountability, and adherence to ethical principles should guide the continued development and deployment of CNNs for criminal face detection. Ultimately, the implementation of CNNs in criminal face detection holds the promise of enhancing public safety and criminal investigations, but it should be approached with caution, responsibility, and an unwavering commitment to fairness and privacy.



Fig-1- Output



Fig-2- Output

8. CONCLUSION

In conclusion, the implementation of Convolutional Neural Networks (CNN) for criminal face detection represents a significant advancement in the field of law enforcement and public safety. CNNs, with their ability to learn and extract intricate facial features, have proven to be a powerful tool in accurately identifying and distinguishing criminal faces from non-criminal ones. Communities' security and the effectiveness of criminal investigations may both benefit from this technology.

9. FUTURE WORK

The process of classifying criminal faces in images include solving current problems and investigating novel ideas to raise the precision, equity, and effectiveness of these systems. These are a few possible directions for further study and advancement in the classification of criminal faces in images. This article covers the several uses of DL that are currently being used in smart fish farming. Species categorization, behavioral analysis, feeding decisions, size or biomass calculation, live fish identification, and water quality prediction are some of the current application fields.

Although no relevant research has been reported to date, other potential application areas with enormous potential include fish disease diagnosis, aquatic product quality safety control, and traceability. It is anticipated that these areas will play a significant role in image processing, especially in managing identified diseases.

This task will continue to perform better with increasingly sophisticated and intricate models. Many of the issues that single models encounter can be resolved by combination models, which leads to the emergence of increasingly intricate architectures. To enhance the overall performance, one can blend handcrafted features with other DL models and classifiers. It is critical to pay close attention to developments in deep computation, ethics, and legal requirements as this implementation progresses. It is critical to pay close attention to developments in advanced learning, ethics, and legal requirements as this implementation progresses.

REFERENCES

- [1] 1.Alon. A, "Tree Extraction of Airborne LiDAR Data Based on Coordinates of Deep Learning Object Detection from Orthophoto over Complex Mangrove Forest", International Journal of Emerging Trends in Engineering Research, vol. 8, no. 5, pp. 2107-2111, 2020
- [2] 2.Coifman.b, "Vehicle level evaluation of loop detectors and the remote traffic microwave sensor", Journal of transportation engineering, vol. 132, no.3, pp. 213- 226, 2006
- [3] 3.Liu.Y, P. Sun, N. Wergeles and Y. Shang, "A survey and performance evaluation of deep learning methods for small object detection", Expert Systems with Applications, vol. 172, p. 114602, 2021.
- [4] Li, J., Wang, Z., Gong, B., and Zhou, W. (2017). "Criminal Face Recognition." Proceedings of the 2017 ACM on Multimedia Conference
- [5] Mery, D., and Filippis, M. L. D. (2013). "Multi-modal Convolutional Neural Networks for Criminal Face Recognition." 2013 International Joint Conference on Neural Networks.
- [6] Li, J., Wang, Z., Gong, B., and Zhou, W. (2017). "Criminal Face Recognition." Proceedings of the 2017 ACM on Multimedia Conference.
- [7] Klare, B. F., Li, Z., and Jain, A. K. (2015). "Matching Forensic Sketches to Mug Shot Photos." IEEE Transactions on Pattern Analysis and Machine Intelligence.
- [8] Lanitis, A., Taylor, C. J., and Cootes, T. F. (2002). "Automatic Interpretation and Coding of Face Images Using Flexible Models." IEEE Transactions on Pattern Analysis and Machine Intelligence.
- [9] Uchôa V, Aires K, Veras R, Paiva A, Britto L (2020) Data augmentation for face recognition with CNN transfer learning. In: 2020 International conference on systems, signals and image processing (IWSSIP). IEEE, pp 143–148
- [10] Srivastava Y, Murali V, Dubey SR (2019) PSNet: parametric sigmoid norm based CNN for face recognition. In: 2019 IEEE Conference on information and communication technology. IEEE, pp 1–4
- [11] Desai DB, Kavitha SN (2019) Face anti-spoofing technique using CNN and SVM. In: 2019 International conference on intelligent computing and control systems (ICCS). IEEE, pp 37–41
- [12] Nagpal C, Dubey SR (2019) A performance evaluation of convolutional neural networks for face anti spoofing. In: 2019 International joint conference on neural networks (IJCNN). IEEE, pp 1–8
- [13] Yan Y, Li C, Lu Y, Zhou F, Fan Y, Liu M (2019) Design and experiment of facial expression recognition method based on LBP and CNN. In: 2019 14th IEEE Conference on industrial electronics and applications (ICIEA). IEEE, pp 602–607
- [14] Perdana AB, Prahara A (2019) Face recognition using light-convolutional neural networks based on modified Vgg16 model. In: 2019 International conference of computer science and information technology (ICoSNIKOM). IEEE, pp

CERTIFICATE



PANIMALAR ENGINEERING COLLEGE

An Autonomous Institution & Affiliated to Anna University, Chennai
Approved by AICTE, New Delhi, Accredited by National Board of Accreditation(NBA)
Approved by UGC for 2(f) & 12(B) Status
Bangalore Trunk Road, Poonamallee, Chennai-600123.



PECTEAM 2K24



This is to certify that Dr./Mr./Ms. JONNA VENKATA SUDHARSHAN of LSE from SATHIVABAMA INSTITUTE OF SCIENCE AND TECH has presented paper entitled IMPLEMENTATION OF CNN CRIMINAL FACE IMAGE CLASSIFICATION in 7th International Conference on Intelligent Computing(ICoNIC) held on 22nd & 23rd MARCH,2024.



FACULTY CO-ORDINATOR Prof. S. VIMALA, M.TECH., (Ph.D)



CO - CONVENER



CO - PATRON Dr. K. MANI, M.E., Ph.D.,

