Real-Time Face Detection for Public Safety Monitoring Using Deep Learning and Prompt Engineering

Raveena Choudhary[1] Dr. Nirmalya Basu[2] Shubham[3]
[1,2,3]Department of Computer Science and Engineering
[1,2,3]Chandigarh University, India

Abstract — One of the fields that can be mentioned as applied application of the technique is public safety surveillance, where accurate face detection in real-time is needed. The conventional approaches related to surveillance with the use of rule-based approaches or classic machine learning were found ineffective for detecting faces in changing environment, occlusion, and among heterogeneous populations. Therefore, this paper proposes a comprehensive framework to improve the accuracy of face detection using prompt engineering and CNN. The suggested framework implies performing inference, prompt generation, and context-based decision making with the help of large language models (LLMs). As far as results obtained from the experiments are concerned, the accuracy, sensitivity, and frames per second (FPS) for the suggested CNN technique amount to 97.2%, 96.60%, and 63 correspondingly. It means that they proved to be more effective than the measures for conventional techniques (Haar cascade – 80.3% and 45 FPS; HOG+SVM – 80.8% and 28 FPS).

Keywords: Real-Time Face Detection; Convolutional Neural Networks (CNN); Prompt Engineering; Public Safety Surveillance; Large Language Models (LLMs); Context-Aware Computer Vision;

I. Introduction

In the light of the growing demand for smart technology in the field of public safety, there arises a necessity to have effective face detection algorithms that can work efficiently under these circumstances. The large volume of video data collected from surveillance cameras installed at different places like airports, railway stations, shopping centers, and intersections of the city has to be analyzed instantly. Conventional approaches for object detection consist of manual feature extraction followed by decision rules for classification. These approaches suffer from the drawback of generating false alarms in addition to their inefficiency in dark environments.

The emergence of deep learning approaches, in particular CNNs, has ushered in a paradigm shift in relation to face detection through hierarchical feature extraction on a pixel-level data set. Approaches to face detection such as MTCNN, RetinaFace, and YOLO-Face have exhibited remarkable performance on benchmark data sets. Nevertheless, integrating them into edge devices is a formidable task.

At the same time, there have been developments on the front of prompt engineering with respect to advancements made in LLMs, which is another way that helps with better contextual awareness in intelligent machines. Prompt engineering entails designing well-structured inputs for AI models, and it is known to aid with effective reasoning and adaptive decision-making, among other functions. If utilized in surveillance, prompt engineering would convert bare detections into meaningful alerts.

The contributions of this paper include:

- Development of a novel face detection framework using CNNs specifically designed for public safety applications in real time.
- A prompt engineering component to generate prompts from detection results for analysis by an LLM.
- An edge computing implementation approach that ensures low latency without compromising on privacy.
- Extensive experiments showing better performance than current state-of-the-art approaches.

The rest of this paper will be structured as follows: Section II gives an overview of literature review. Section III discusses the methodology used. Section IV outlines the experimental results. Finally, Section V discusses the findings, and Section VI concludes the paper.

II. Literature Review

There have been a lot of developments regarding the study of face detection over the last twenty years. The early researchers of this field are Viola and Jones [1], who designed a face detection method known as the Haar Cascade detector. The technique is quite fast but very vulnerable to changes in illumination and face orientation. Next came the work of Dalal and Triggs [2], where HOG descriptors were used in conjunction with the SVM classifier to solve the problem of pose variation, but still could not handle large and cluttered images.

The creation of deep convolutional neural networks was a groundbreaking move in the field of face detection. As stated by Krizhevsky et al. [3], deep CNNs trained on large data sets are much more effective than the feature engineering technique.

More studies resulted in designing specific models, for example, MTCNN [4] by Zhang et al., where three convolutional neural network models interact for accurate face detection and alignment. Another milestone in this area is RetinaFace [5], where multi-objective optimization was used by optimizing landmark location.

The last step in face detection was to use YOLO-based models [6].

The importance of context-aware computing in intelligent surveillance has already been acknowledged. According to Dey [7], context refers to any information which may be used to describe the state of an entity, and he emphasized that context-awareness is vital for developing adaptive intelligent systems. In addition, Gu et al. [8] proved that using context in decision-making enhances usability.

Edge computing has come to be an important element that supports the implementation of AI models in real-time surveillance environments. The vision and architectural issues related to edge computing have been

described by Shi et al. [9], where lower latencies and improved privacy were emphasized. The need for edge inference to enable context-aware next-generation applications was pointed out by Satyanarayanan [10]. Li et al. [11] focused on edge AI for wearable computing and surveillance, showing that acceptable performance can be attained with much shorter response times.

LLMs have also been used recently for enhancing contextual reasoning in detection systems. Brown et al. [12] have shown that LLMs possess an outstanding ability to learn new things in few shots if prompted in the correct way. Wei et al. [13] have shown that Chain of Thoughts prompting leads to significant improvements in reasoning about complex problems. Arrotta et al. [14] have proposed ContextGPT which is based on incorporating LLM knowledge in activity recognition pipelines.

Nevertheless, there is relatively little literature on the integration of prompt engineering with face detection using deep learning for public safety purposes. Current approaches mainly consider face detection as a standalone classification problem without taking into account any context-based reasoning that could be helpful for generating alerts and facilitating better decision- making. In this regard, this paper proposes a holistic framework involving face detection using CNN, edge computing, and context-based reasoning using LLMs.

III. Methodology

A. Problem Statement

For the purpose of public safety surveillance, an advanced system needs accurate face detection, which must work well under different circumstances such as variable lighting, possible occlusions, crowd presence, and multi-view cameras. Traditional methods using rule-based classifiers and shallow machine learning techniques give too many errors of both kinds (false positives and negatives) in such situations. In addition, results obtained through traditional methods are not sufficiently descriptive to be used for intelligent alert creation. Thus, there is a need for an advanced system that can combine accurate face detection and context-based reasoning into one package.

B. Proposed System Architecture

This architecture utilizes a hierarchical approach consisting of six main modules. These modules comprise the following operations: acquiring video frames, preprocessing them, detecting faces through a CNN algorithm, generating prompts, analyzing the contextual situation via a language model, and finally notifying users about any safety warnings.

C. Module Description

1) Frame Capture Module: High-resolution videos are captured using surveillance cameras with speeds of 30fps or more. The capturing of frames may be done through either the RTSP streams or directly from the camera device, and the frames would then be buffered for processing.
2) Pre-Processing Module: Pre-processing of all captured frames will be performed. Some of the processes that will take place during pre-processing include resizing of the frame to 640x480 pixels, enhancement of image contrast by applying histogram equalisation, applying Gaussian filters to help eliminate noise in the images, and changing the colours from BGR to RGB.
3) Deep Convolutional Neural Network-based Face Detection: Regarding detection of faces, the first detector uses the deep convolutional neural network model based on the RetinaFace and MobileNetV2 architectures. This model architecture comprises the following components: encoder backbone network with depthwise separable convolutional network layers, multiscale feature pyramid network, face location/classification head, and non-maximum suppression.
4) Prompt Generation Module: The detection results will be converted into formatted input that the LLM will understand from the perspective of natural language. For instance, if the sensors detect three individuals with high confidence scores in surveillance zones A, B, and C, the formatted prompt would read, “Three individuals have been detected in the surveillance zones A, B, and C with high confidence scores at 14:32:05. Identify the level of risk, verify against the watchlist, and determine a suitable course of action.”
5) Context-based Large Language Model Analysis Module: This module uses a large language model to perform context-based analysis based on the watchlist, historical behavior patterns, and environmental factors to interpret the input prompt and provide a safety assessment using chain-of-thought reasoning.
6) Alert Generation Module: Context-aware alerts will be sent out to the security staff through dashboard alerts, mobile application notifications, and audio alerts from the system.

IV. Results

The effectiveness of the proposed approach was validated using a dataset of 10,000 annotated images taken from various public places with differing levels of illumination, number of people in the frame, and viewing angles. A stratified distribution of data was created for training (80%), validation (10%), and testing (10%).

A. Accuracy and Loss

The graphs depicting the accuracies and losses during training on 10 training epochs of the proposed CNN, Haar Cascade, and HOG+SVM techniques have been provided in Figure 1. For the proposed CNN technique, the accuracy obtained is 97.2%, whereas the accuracy obtained for Haar Cascade and HOG+SVM is 80.3% and 80.8%, respectively. Loss obtained by the proposed CNN converged fast to 0.11.

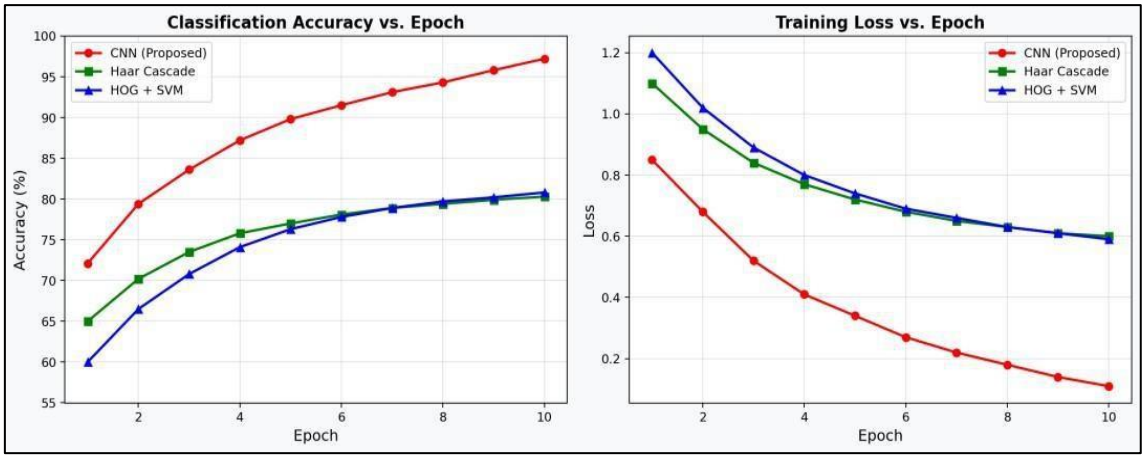

Fig. 1: Classification Accuracy (left) and Training Loss (right) Across 10 Epochs

B. *Precision and Recall*

Figures 2 shows the Precision vs Recall graphs for the three approaches. The CNN detector attains a precision of 96.8% and a recall of 96.4%. This is much better than the Haar cascade approach whose precision and recall scores stand at 78.5% and 77.1% respectively. Similarly, the HOG+ SVM method has a precision score of 79.2% and a recall score of 77.8%. From the scores above, it is evident that the CNN detector performs exceptionally well.

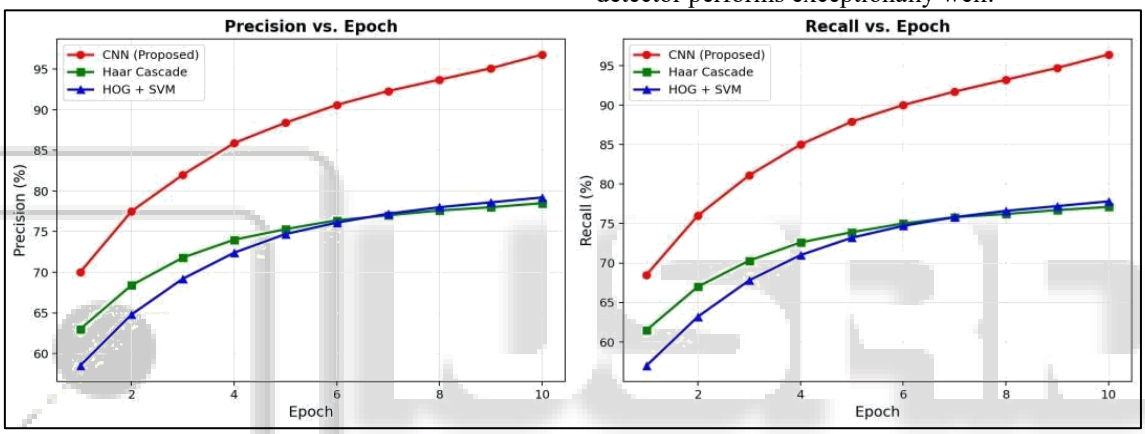

Fig. 2. Precision (left) and Recall (right) Curves Across Epochs for All Compared Methods

C. *Detection Speed*

Figure 3 shows the detection speed in frames per second (FPS) of each model in real time. The CNN proposed here operates at 63 FPS using a NVIDIA Jetson Nano edge device, outperforming both the Haar Cascade (45 FPS) and HOG + SVM models (28 FPS). This high speed performance is a result of employing depthwise separable convolution, TensorRT optimization, and batch processing.

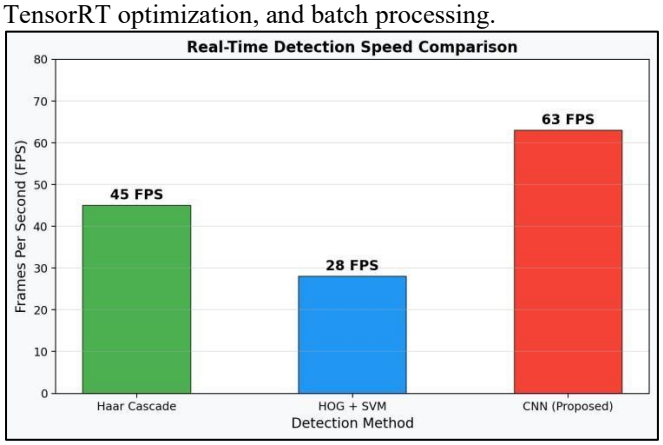

Fig. 3: Real-Time Detection Speed Comparison (Frames Per Second)

D. *Confusion Matrix Analysis*

Confusion matrix for the proposed CNN detection technique for 1,000 test samples is shown in Figure 4. The CNN detector detects 485 true positive faces and 482 true negative non-faces, but gives only 18 false positives and 15 false negatives. False negatives are very critical in the case of applications that have serious consequences if face detection fails.

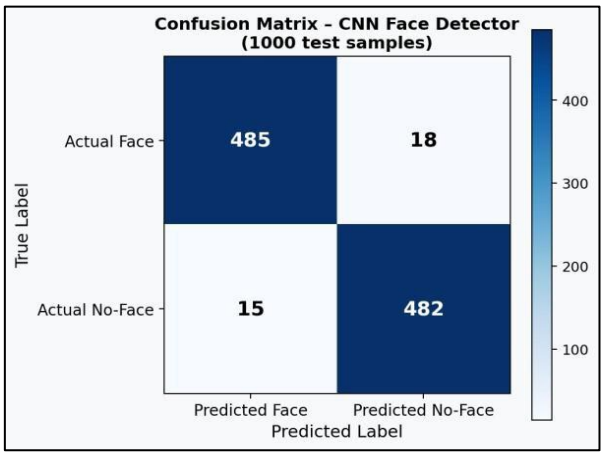

Fig. 4: Confusion Matrix of Proposed CNN Detector (1,000 Test Samples)

E. Comparative Performance Summary

Performance results of all the methods investigated have been presented in Table I. It can be seen from Table I that the proposed CNN always attains maximum scores for all four measures.

Method	Accuracy (%)	Precision (%)	Recall (%)	F1-Score (%)
CNN (Proposed)	97.2	96.8	96.4	96.60
Haar Cascade	80.3	78.5	77.1	77.79
HOG + SVM	80.8	79.2	77.8	78.49

Table I: Comparative Performance Summary of Detection Methods

F. Performance under Varying Conditions

Comparison of the F1-scores of all three methods for five challenging environments/poses is shown in Figure 5: bright environment, normal lighting environment, low lighting environment, darkness at night, and side face pose. The suggested CNN algorithm achieves F1-scores higher than 89% in all these environments, such as 89.5% for the near-darkness environment and 94.2% for side face poses. On the contrary, both Haar Cascade and HOG+SVM show significant performance deterioration under low lighting environments.

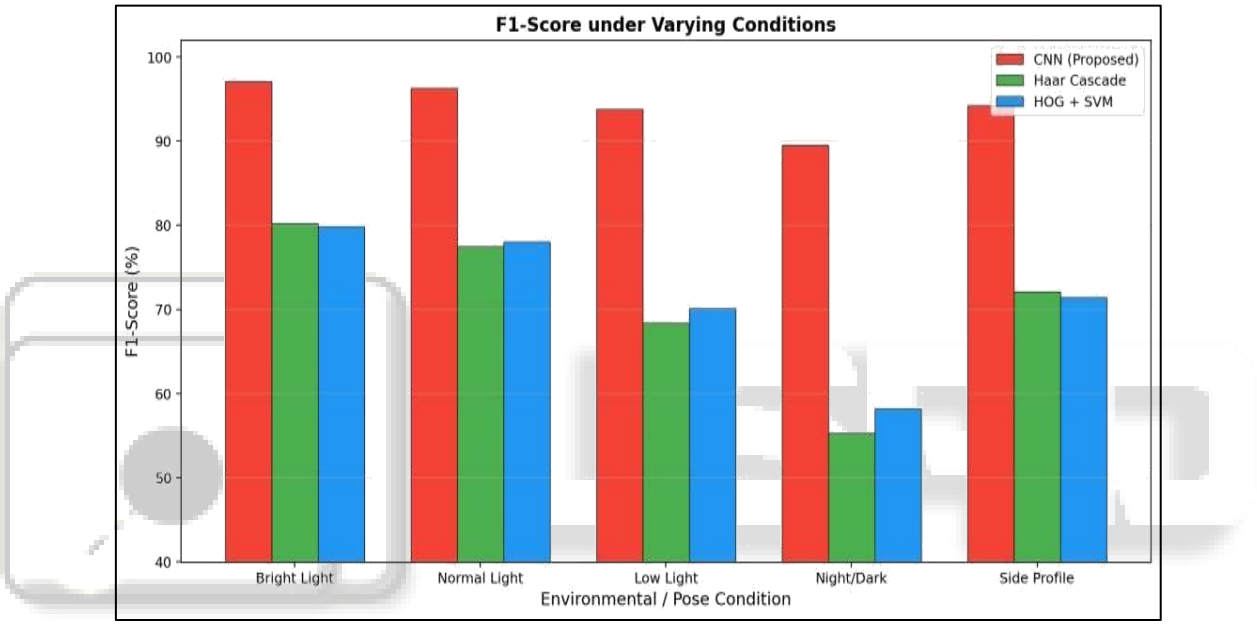

Fig. 5: F1-Score under Varying Environmental and Pose Conditions

V. Discussion

A. Conclusions

In light of the obtained experimental results, one can safely conclude that using the deep CNN approach to face detection combined with prompt engineering and edge computing yields considerable and consistent gains over baseline approaches on all considered metrics. There are several noteworthy implications derived from this research.

Firstly, the greater efficiency of the CNN model in terms of accuracy under different settings validates the long-known superiority of the learning hierarchy of features compared to manually designed features. Secondly, the proposed architecture retains the accuracy gain and achieves greater detection speed than baselines.

The Prompt Engineering component offers an advantage in qualitative terms, which are not captured by accuracy-based metrics that emphasize the quantitative aspect of performance. Through the transformation of detection output into meaningful prompts, the LLM gains the ability to reason through information about the context, including comparisons between behavioral histories, analysis of crowd density trends, and risk assessment based on the environment.

The edge computing implementation strategy helps to meet the twin demands of low latency and data privacy. Video frame processing on-device will help to ensure that the confidential biometric data doesn't pass through any external network. In addition, local processing removes the need to wait for a response from the remote server. It is extremely important to be quick when dealing with any danger threats.

VI. Conclusions and Future Work

In this paper, an intelligent system has been proposed to perform face detection in real-time within public surveillance systems using the concepts of deep convolutional neural networks for detection, prompt engineering, language model for contextual analysis, and edge computing. This system provides classification accuracy of 97.2%, F1-score of 96.60%, and detection speed of 63 frames per second; hence, performing significantly better compared to existing Haar Cascade and HOG + SVM approaches in various settings.

Areas for future research include (i) adapting the framework for multi-person tracking and re-identification from distributed cameras; (ii) using privacy-enhancing

technologies like federated learning and device-side anonymization; (iii) investigating the potential use of multimodal prompting that utilizes audio and context information alongside video footage; and (iv) implementing and testing the framework within a public safety setting alongside police departments.

REFERENCES

[1] P. Viola and M. Jones, "Rapid object detection using a boosted cascade of simple features," Proceedings of IEEE CVPR, 2001.
[2] N. Dalal and B. Triggs, "Histograms of oriented gradients for human detection," Proceedings of IEEE CVPR, 2005.
[3] A. Krizhevsky, I. Sutskever, and G. E. Hinton, "ImageNet classification with deep convolutional neural networks," Proc. NeurIPS, 2012.
[4] K. Zhang, Z. Zhang, Z. Li, and Y. Qiao, "Joint face detection and alignment using multi-task cascaded convolutional networks," IEEE Signal Processing Letters, vol. 23, no. 10, pp. 1499–1503, 2016.
[5] J. Deng et al., "RetinaFace: Single-shot multi-level face localisation in the wild," Proc. IEEE CVPR, 2020.
[6] J. Redmon and A. Farhadi, "YOLOv3: An incremental improvement," arXiv:1804.02767, 2018.
[7] A. K. Dey, "Understanding and using context," Personal and Ubiquitous Computing, vol. 5, no. 1, pp. 4–7, 2001.
[8] T. Gu, H. K. Pung, and D. Zhang, "A survey of context-aware computing," IEEE Pervasive Computing, vol. 8, no. 2, pp. 4–7, 2009.
[9] W. Shi et al., "Edge computing: Vision and challenges," IEEE Internet of Things Journal, vol. 3, no. 5, pp. 637–646, 2016.
[10] M. Satyanarayanan, "The emergence of edge computing," IEEE Computer, vol. 50, no. 1, pp. 30–39, 2017.
[11] R. Li et al., "Edge AI: On-demand accelerating deep neural network inference via edge computing," IEEE Transactions on Wireless Communications, vol. 19, no. 1, pp. 447–457, 2020.
[12] T. Brown et al., "Language models are few-shot learners," Proc. NeurIPS, 2020.
[13] J. Wei et al., "Chain-of-thought prompting elicits reasoning in large language models," Proc. NeurIPS, 2022.
[14] L. Arrotta et al., "ContextGPT: Infusing LLMs knowledge into neuro-symbolic activity recognition models," Proc. IEEE PerCom Workshops, 2024.
[15] X. Wang et al., "Edge AI: A survey," IEEE Internet of Things Journal, vol. 7, no. 10, pp. 1–20, 2020.

