APPROVAL PAGE

This research work titled:

"DEVELOPMENT OF AN AI-BASED CROWD DENSITY ESTIMATION SYSTEM"

Was carried out by CHIBUKO SAMUEL with matriculation number GOU/U22/CSC/801 and has been read, supervised, and approved for the Department of Computer Science and Mathematics, Faculty of Computing and Information Technology, Godfrey Okoye University, in partial fulfillment of the requirements for the award of Bachelor of Science (B.Sc.) Degree in Computer Science.

Engr. Kingsley Chibueze ....

Project Supervisor Signature Date

External Examiner ....

Signature Date

Dr. Frank Okebanama

HOD. Department of Computer Science	....	....
And Mathematics	Signature	Date

Dr. I.A. Ajah		
Dean, Faculty of Computer Science and	....	....
Mathematics	Signature	Date

CERTIFICATION

I hereby certify that this project work titled:

"DEVELOPMENT OF AN AI-BASED CROWD DENSITY ESTIMATION SYSTEM"

Was carried out by me, CHIBUKO SAMUEL with matriculation number GOU/U22/CSC/801, in the Department of Computer Science and Mathematics, Faculty of Computing and Information Technology, Godfrey Okoye University.

I further certify that this research work is original and has not been submitted either wholly or partly for the award of any degree in any institution. All sources of information used have been duly acknowledged through references.

CHIBUKO SAMUEL

GOU/U22/CSC/801

....

ACKNOWLEDGEMENTS

First and foremost, I give all glory and gratitude to Almighty God for His grace, strength, wisdom, and guidance throughout the course of this research work and my entire undergraduate studies.

I sincerely express my profound appreciation to my supervisor, Engr. Kingsley Chibueze, for his invaluable guidance, patience, constructive corrections, and continuous support towards the successful completion of this project work. His contributions and encouragement played a major role in shaping this research.

I also wish to appreciate the Head of Department, Dr. Frank Okebamama, alongside all the lecturers and staff of the Department of Computer Science and Mathematics, Faculty of Computing and Information Technology, Godfrey Okoye University, for their academic support, knowledge, and dedication throughout my period of study.

My heartfelt appreciation goes to my parents, family members, and loved ones for their prayers, financial support, encouragement, and understanding during challenging moments of this academic journey. Their sacrifices and motivation were a constant source of strength.

I equally appreciate my friends and colleagues who contributed in one way or another towards the success of this project through advice, encouragement, and assistance.

Finally, I remain grateful to everyone who contributed directly or indirectly to the successful completion of this research work.

TABLE OF CONTENT

LIST OF FIGURES

DEVELOPMENT OF AN AI-BASED CROWD DENSITY ESTIMATION SYSTEM

BY

CHIBUKO SAMUEL

GOU/U22/CSC/801

DEPARTMENT OF COMPUTER SCIENCE AND MATHEMATICS, FACULTY OF COMPUTER SCIENCE, GODFREY OKOYE UNIVERSITY, ENUGU STATE.

IN PARTIAL FULFILLMENT OF THE AWARD OF BACHELOR OF SCIENCE (B.SC), DEGREE IN COMPUTER SCIENCE.

SUPERVISOR: DR. O.U OGUIKE

JULY, 2025

ABSTRACT

This project work of research focused mainly on developing a very simple AI-based crowd density estimation system that is very effective and efficient. During the process of the research, the researcher employed the use of the ShanghaiTech dataset as the underlying data source for training the AI model. The dataset was made up of diverse crowd images which were group into dense and sparse groups. The dataset was therefore cleaned and enhanced in other to optimize the training of the machine learning model. Using the cleaned dataset, a deep learning model was trained using the Convolutional Neural Networks algorithm along side Adam and Stochastic Gradient Descent optimizers in other to generate density maps under conditions that have heavy occlusion. However, at the end of the development phase, the system was tested using different evaluation metrices, where the system performed effectively, achieving a Mean Absolute Error (MAE) of 12.4 and a Root Mean Squared Error (RMSE) of 18.2, while being able to successfully classify crowd density into low, medium, high , and critical levels. Moreover, at the end of the research, all the system requirements were fully met successfully, thereby achieving the project's core objectives for public safety monitoring.

CHAPTER ONE

INTRODUCTION

1.1 BACKGROUND OF THE STUDY

The rapid growth in the population of urban areas, coupled with the increase in social, religious, political and entertainment activities has led to the gathering of large crowds frequently in public spaces. This type of crowds is most commonly seen in concerts, religious pilgrimages, events that are related to sporting, transportation hubs, marketplaces, and most especially political rallies (wang et al., 2025). However, while these gatherings are peaceful most times, they can quickly take a bad turn and become dangerous when the density of the crowd increases too much and exceeds the safe limits of the space, thereby leading to serious incidents such as stampedes, suffocation, injuries, or even loss of lives (Gao et al., 2025). It is important to note that several tragic disasters that are related to crowd, which have been recorded across the world have highlighted the need for effective crowd monitoring and management systems urgently.

However, the traditional methods of crowd monitoring have relied heavily on manual observation by security personnels and also the use of basic surveillance systems. But these methods of monitoring are most often subjective, prone to a lot of errors, and ineffective in environments that are highly congested, where individuals overlap one another or are partially occluded (Fan et al., 2022). Thus, the conventional techniques of counting people also struggle a lot in scenes that are highly dense sue to scale variation, distortion of perspective, and limited visibility. These limitations however join forces to make it hard to accurately estimate the size and density of a crowd in real-time, most especially in environments that are largescale (Hansen et al., 2022).

Consequently, recent advancements in technology joined with the continuous improvement in artificial intelligence (AI), computer vision and deep learning, new possibilities for automated analysis of crowd have opened (Patwal et al, 2023). This AI-based technique of crowd density estimation however, focuses mainly on learning the visual patterns that exist in images or video frames to generate density maps that represent the distribution of people within a particular scene (Kiruthika & Suchitra, 2023). However, unlike the traditional approaches to crowd density detection, the density estimation methods perform very well even in environments that are highly congested, where individual detection is nearly impossible. These techniques therefore provide estimates of crowd size that are more reliable and also enable for the continuous monitoring over time (wang et al., 2025).

Nonetheless, in line with all these recent improvements in technology, this project work focuses mainly on the design and implementation of an AI-Based Crowd Density Estimation System that would be capable of analyzing video footage from surveillance cameras or videos that were prerecorded by users to estimate the density of a crowd and the accurate count of the population (Sujihelen et al., 2025). The proposed system however integrates the acquisition of video, preprocessing of video, the extraction of features, the estimation of the crowd density, visualization, and most especially an alert mechanism into a single intelligent platform (Padmalal et al., 2025). And by providing real-time or near-real-time insights into the conditions of crowds, the system aims to support the safety of public agencies, event organizers, and urban planners in making decisions in an informed way and most especially prevent disasters that are related to large amount of crowd (Leisha et al., 2025).

1.2 STATEMENT OF PROBLEM

Notwithstanding the widespread introduction and deployment of surveillance cameras in public spaces, the effective monitoring of crowd and the estimation of its density have remained a challenge that is significant. The major problems that fueled this project work includes the following:

1. The traditional techniques of head counting are unreliable, subjective, inefficient and normally fail in environments that are highly congested due to occlusion, overlapping individuals and poor visibility of security personnels
2. The existing surveillance systems mainly provide raw video feeds without intelligently analyzing or providing actionable insights on the density level of crowd.
3. Lack of an automated real-time alert mechanisms in currently existing systems making it difficult to detect crowd buildup that are dangerous before exceeding critical thresholds.
4. The variation of camera angles, lighting conditions and scene perspectives significantly reduce the accuracy of conventional methods of crowd analysis.
5. A good number of the currently existing systems have a poor user interface and are not able to handle continuous video streams in real-world scenarios.

1.3 AIM AND OBJECTIVES OF THE PROJECT

The main aim of this project is to design and implement an AI-based system that can automatically estimate the density and the population count of a crowd, from a video footage to enhance the management of crowd and the safety of the public. However, the specific objectives of this study are to:

1. Design a video-based crowd monitoring system that is capable of processing live or recorded video footage.
2. Develop and incorporate a preprocessing module to enhance video frames through normalization, noise reduction and perspective correction.
3. Incorporated and AI-based crowd density estimation model that generates density maps that are accurate even in scenes that are highly congested.
4. Estimate crowd count by integrating density map values that were obtained from video frames and classifying the crowd density into predefined levels such as low, medium, high and critical.
5. Implement an alert mechanism that triggers warnings when the density of crowd exceeds safe thresholds.

1.4 SIGNIFICANCE OF THE PROJECT

This project is of high significance to a lot of stakeholders which includes the following:

1. Public safety agencies,
2. Event organizers,
3. Urban planners
4. Educators and researchers.

This project however strikes a tune of importance to the above-mentioned stakeholders through the following ways:

1. To the public safety agencies, the system provides a tool that is very intelligent for continuous monitoring of crowd, enabling the detection of dangerous buildup of crowd and also help in timely intervention.

2. For the event organizers, it will help them monitor the distribution of crowd in real time, improve their strategies of controlling crowd and most importantly help them enhance the overall safety of participants.
3. The urban planners and smart city administrators can leverage on the collected crowd density data to analyze the movement patterns of humans, improve the planning of infrastructure and also optimize the use of public spaces.
4. for the researchers and students, this project contributes to the actively growing body of knowledge that is existing in computer vision and artificial intelligence by demonstrating the practical application of AI-based density estimation techniques in a real world scenario.

1.5 SCOPE OF THE PROJECT

This project work focuses mainly on the design and implementation of an AI-Based Crowd Density Estimation system using the video data that was captured from surveillance cameras or footage that was prerecorded. The system is however limited to estimating the density and population count of a crowd without performing facial recognition or tracking of individual identity. This project therefore covers the preprocessing of video, extraction of feature, estimation of density and the generation of alerts. Moreover, the system implementation is therefore designed to support both real-time and offline analysis, depending on the availability of hardware. The system is therefore evaluated under different crowd scenarios, which includes sparse and dense environments, varying lighting conditions, and different perspectives of camera. However, it is important to note that this project implementation will not currently cover multi-camera data fusion or large-scale city-wide deployment.

1.6 LIMITATION OF THE PROJECT

During the course of this project research, the researcher encountered numerous certain limitations which may in one way or the other turn out to affect the overall performance and generalization of the proposed system. These limitations include but are not limited to the following:

1. **Data Dependency:** The accuracy of the crowd density estimation model depends heavily on the quality and diversity of the video data that was used for training and evaluating the model. Hence, poor resolution of videos, motion blur, or the limitedness of dataset diversity may reduce the accuracy of the model's estimation.
2. **Occlusion and Extreme Density:** although the approach of density-based estimation perform better than the traditional counting methods, extreme occlusion and very high density of crowd can still impact the precision of density map generation negatively.
3. **Camera Perspective Variations:** significant changes in the angles, heights and perspectives of cameras may affect the ability of the system to generalize well across different environments without the need for additionally retraining or calibration of the model.
4. **Computational Cost:** the estimation of the density of crowd in real-time using deep learning models required substantial computational resources. Therefore, limited hardware capacity may introduce latency and reduce the performance of the system when deployed in real-time.

1.7 OPERATIONAL DEFINITION OF TERMS

There are numerous technical terms that were used throughout this project work of research, this section therefore defines the core concepts that are essential to the understanding of the scope of the study for this system. some of these technical terms include but are not limited to the following

1. **Machine Learning:** This is a sub subset of artificial intelligence which allows the computer to learn and make decisions or predictions from a provided dataset without necessarily being programmed to do so.
2. **Prediction:** This can be described as a student degree result forecast as it involves the estimation of future result outcomes based on historical data and analytical techniques.
3. **Supervised Learning:** This involves a scenario in a machine learning used when a dataset is labeled, where predictive models usually in machine learning technique, use it to predict the outcome of the data.
4. **Data Preprocessing:** This is the step undertaken to clean and organize academic data in order to prepare the data as well as ensure it is in a machine learning friendly format.
5. **Accuracy:** This shows the performance of the predictive model and is evaluated generally in order to show how well the prediction was in relation to the actual outcome.
6. **Feature**: This is a single measurable attribute or characteristic that is based on the given data, for instance a student's exam score, attendance, or GPA and can be used by machine learning models to predict outcomes.
7. **User Interface (UI):** This is the visual appearance of the system that allows the user to interact with the system.
8. **Early intervention:** This is the act of identifying and supporting students that are at risk of poor academic performance before the final results are released.

CHAPTER TWO

LITERATURE REVIEW

2.1 INTRODUCTION

This chapter of this project presents both the theoretical and conceptual foundations that are related to crowd density estimation and the role that artificial intelligence plays in the analysis of crowds automatically. It discusses mainly, the underlying theories that supports the modeling of crowd behavior, reviews the conventional approaches that is used for estimating crowd density and also examines how machine learning techniques have successfully improved accuracy and reliability in an environment that it dense with crowd. This chapter of this chapter also highlights how this project contributes to the already existing knowledge in the domain of AI-based crowd monitoring systems.

2.2 THERORITICAL BACKGROUND

2.2.1 Overview of Degree Results

The term crowd density estimation is rooted in an array of theories which ranges from computer vision, recognition of pattern and the analysis of the behavior of human crowds (Wang et al., 2025). It is important to note that one of these theoretical concepts is the crowd flow theory, which stands to explain how individuals move collectively in a space that is shared, and also how congestion tends to emerge when the demand for space exceeds the capacity of a given space (Kushwaha & Guputa, 2023). The theory however, provides a basic knowledge for understanding how the density of crowd increases over time and how situations that are dangerous can easily arise in public gatherings (Tyagi et al., 2022).

However, another theory which stands as an important foundation is the theory image formation and visual perception theory, which explains how objects, textures, and motion patterns are being represented in digital images (Wang et al., 2025). It therefore puts forward that in crowds that are dense, individuals most of the time appear as repetitive texture patterns rather than distinct objects. At this, the density estimation techniques exploits this idea by learning and making use of spatial distributions instead of explicitly detecting individuals (Fan et al., 2022).

Moreover, from the perspective of artificial intelligence, the theory of statistical learning now underpins the crowd analysis that is based on machine learning (Kiruthika & Suchitra, 2023). This theory however leans forward to explain how models learn patterns from data and generalize to scenarios which are unseen. therefore in the estimation of crowd density, models learn mappings between visual features and density values of crowd, enabling accurate estimation even under occlusion and variation of scale (Wang et al., 2022). Thus, these theories when combined together, justify the use of density-based and learning driven approaches for estimation crowd, with particular docus in environments that are highly congested where traditional methods of detection fail.

2.2.2 Conventional Methods of Crowd Density Estimation

Before the wide adoption of artificial intelligence (AI), several conventional methods were used for counting the amount of people or estimating the size of people in a given crowd. These conventional methods however lack scalability and robustness, particularly in environments that are dense in the real world. Some of these methods include but are not limited to the following

- **Manual Counting and Visual Inspection:** This approach relies on the observation of humans to estimate the size of a crowd by visually inspecting a footage from a surveillance camera (Akhtar & Malhotra, 2022). But while this method has proved to be simple and straight forward, it is highly subjective, highly prone to errors, and most especially, unsuitable for a crowd that is large or dense.
- **Detection-Based Methods:** this approach crowd estimation tries to identify and count individual people using features such as their heads, bodies, or silhouettes (Alrowais et al., 2022). It commonly makes use of different techniques which include background subtraction, Haar cascades, and detectors that are based on histogram. However, these methods tends to perform well reasonably in scenes that are sparse but tend to fail in crowds that are dense due to the severe occlusion and individuals who are overlapping one another.
- **Regression-Based Methods:** this method estimating crowd density works by extracting features that are low-level such as texture, edges, and gradients and then use regression models to estimate the count of the crowd directly from the features. Although this method is much more robust than the approaches that are based on detection, they often lack information that are spatial and moreover struggle with variations of scenes (Khan et al., 2023).
- **Clustering and Motion-Based Methods:** This approach to crowd density estimation, estimates the density of a crowd by analysing the flow of motion and clustering movement patterns (Fan et al, 2023). This methods are however sensitive to the motion of cameras, changes in lighting and background noise.

2.2.3 machine learning and its application in crowd estimation system

Machine learning on the other hand, has significantly improved crowd estimation by enabling systems to learn visual patterns that are complex from data (Wang et al, 2025). Just like in contrast to the traditional methods, approaches that are based on machine learning do not rely the explicit detection of individuals but instead, learn a mapping between visual inputs and density information. The early machine learning approaches made use of handcrafted features that are combined with regression models such as support vector regression and random forests. But while these methods helped to improve the accuracy of crowd estimation systems, they still depended heavily on feature engineering (Khan et al, 2023).

Consequently, recent advances in deep learning, with a particular focus on the convolutional neural networks (CNNs), have transformed the estimation of crowd density. However, models that are based on CNN now learn hierarchical features automatically from images and at this generate density maps that represents the distribution of crowd across a scene (Alrowais et al., 2022). These models therefore, handle occlusion, scale variation, and perspective distortion more effectively than the methods that came before them. Moreover, in crowd density estimation systems that are based on AI, machine learning models are typically trained on datasets that are annotated and integrated into a video processing pipeline. The trained model now process video frames, generate density maps, estimates the count of a crowd, and supports monitoring and alerting in real-time (Wang et al., 2022).

2.3 REVIEW OF EXISTING RELATED WORKS

This section oof this project work will stand to review some of the works that are already existing which are related to this project work in one way or the other.

Alsubai et al., (2024), all contributed to propose a crowd density analysis technique that is driven by Artificial intelligence for smart cities which are sustainable. Their study focused mainly on using models of deep learning to analyze and classify the how dense a crowd is using images. The system made use of CLAHE for enhancing images, inception v3 for extracting features from images, and GRU for classifying the density of crowds, it also made use of optimization algorithms for tuning hyperparameters.

Dheepak et al., (2025), designed and built Smart Crowd, which was a framework that was driven by AI, for monitoring the density of a crowd in real-time in large public gatherings in India. The system was built to combine CCTV, footages from drone, and signals from Wi-Fi/Bluetooth. It was designed to make of YOLOv5 for detection, CNN-based density mapping, and LSTM for predicting congestion. The system was implemented on edge devices, and at that achieved real-time performance with low latency.

Mannickathan & shamsu (2025), proposed a predictive system that is driven by AI which could be used to monitor and manage the flow of crowd in environments that are highly dense such as airports, malls and venues of events. The researchers made use of CSRNet for estimating the density of a crowd and graph-based modelling in predicting congestion and suggesting measures of control proactively. They developed a simulator based on the python programming language in other to test the crowd flow strategies both in the indoor and outdoor settings.

Basthikodi et al., (2024), proposed a surveillance system that is based on AI and machine learning that integrates a network of CCTV for monitoring crowd and detecting crimes. The system was built to make use of images and live video feeds to detect anomaly, recognize patterns, and give security alerts so as to improve the safety of the public. The study however emphasized heavily on improving efficiency and accuracy over manual methods of surveillance.

Ashwini & Yakapur (2025), worked on investigating a surveillance system that is based on deep learning for monitoring crows and detecting violent behaviour in real time. The system combined MobileNetSSD for detecting crowd with YOLO-style CNN-RNN and vision models that are CLIP-based for classifying violence, alongside visualization in real-time using MongoDB and React. The result showed that the system was responsive in public surveillance enviroments.

Zhang et al., (2022), created CrowdOptim, which was a crowd collaborative framework that was based on AI and was made to be used in optimizing the hyperparameters of neural networks in smart urban sensing applications that are based on AI. The study made use of human input from systems that source crowd to help improve the selection of hyperparameters for smart city tasks such as monitoring infrastructure and assessing the cleanliness of urban areas. When the system was tested, it performed well and outperformed the other systems that were already existing at the time.

Jayasudha et al., (2025), developed a real-time crowd monitoring system that is driven by AI. The system combined social LSTM for predicting trajectory facial and emotion recognition features that are based on deep learning, and blockchain technology for managing data securely. The system performed very well in facial recognition and detecting emotions when it was tested in Varanasi, India. The study therefore showed practically, how the analysis of gesture and sentiment can help authorities to respond early to potential risks that may break out in crowds that are dense.

Zhu et al., (2024), proposed a framework for crowd analytics that was based on AI. The system was built to be used on transit stations so as to support the planning of operations and also manage and enforce safety protocols. The system integrated the use of YOLO and Deep SORT for detecting an tracking individuals from video footage, enabling it to accurately estimate crowd density, flow volume, and the walking speed of people in environments that are of high density. The system however performed well when and achieved a realistic density-speed and performed with great counting accuracy.

Maharajpet & Hegde (2025), developed a system that estimated the density of a crowd in real-time. The system combined the use of YOLOv8 for detecting people with CSRNet for estimating density using video feeds from CCTV and drones. The system however functioned by classifying the density of crowd into four different levels and in the end present the results through a web-based dashboards using heatmaps, analytics reports and historical data for supporting decision. The system was tested using benchmark datasets and videos from real world, and the result showed highly accurate detection and low errors in counting.

Basunduwah et al., (2024), proposed a model that is multi-visual for the sole purpose of predicting crowd and estimating the its density. The system was known as AIM-CP and was built to improve the accuracy of analysing crowd mobility dynamics. Their approach combined the detection ability of YOLOv8 with an expert system that is rule-based and multi-modal features from body, face and pose movements to enhance the prediction of mobility in real-time. at the end of the study, the model performed well when it was tested outperforming other methods that are single-modal.

Ilyas & Bawany (2025), developed a framework called CRAB-NET, which was based on AI and was mainly for recognising the behaviour of a crowd automatically with the use of videos from surveillance cameras. The study however, introduced a real-world crowd dataset that is diverse and applied the models of ConvLSTM and LRCN to classify the behaviour of crowds. It achieved a very high accuracy across celebratory, violent and formal scenarios, and results showed that deep learning can support proactive crowd safety and security interventions reliably.

Macriga et al (2024)., proposed a framework that is based on both AI and ML which was tailored to the urban environment of India and was aimed at improving the management of crowd, preventing crime, and supervision of workplace. The framework applied the use of predictive analytics and computer vision in monitoring the behaviour of crowd, detecting anomalies and anticipating safety risks while still accounting for both contextual and cultural factors. The system however, supported the prevention of crime through the analysis of pattern and therefore improves the safety of industrial environments by monitoring the activities and compliance of workers.

Cheong et al., (2019), presented a video surveillance approach that is low-cost and automated which uses object recognition that is computational to identify, track and count people from CCTV video streams. The study however compared the implementations of two software and validated their performance in both real-world and controlled environments, showing results that are reliable for monitoring crowd that is of medium-density.

Thangamani et al., (2025), examined the effective use of deep learning with main focus on the Convolutional Neural Networks (CNNs), for counting crowds in the surveillance of campus environments. The study however showed how models which are based on CNN tend to improve the estimation of crowd density by learning complex patterns that are spatial and adapting to environmental conditions that are changing, thereby supporting safety monitoring and alerts that are automated proactively.

2.4 SUMMARY OF REVIEW OF RELATED WORKS

S/N	AUTHOR(S)	CONTRIBUTION OR (WORK DONE)	TECHNIQUE OR METHODOLOGY	LIMITATIONS
1.	Alsubai et al., (2024)	Used models of deep learning to analyse and classify the how dense a crowd is using images.	Convolutional Neural Network (CNN)	Failed to implement any form of safety alerts system.
2	Dheepak et al., (2025),	Created an AI-driven framework for monitoring the density of a crowd in real-time in large public gatherings in India	Convolutional Neural Network (CNN)	The system was limited to India.
3	Mannickathan & shamsu (2025).	Developed an AI-based predictive system for monitoring and managing crowd flow in highly dense environments	CSRNet	Made use of multiple input sources.
4	Basthikodi et al., (2024).	Emphasized heavily on improving efficiency and accuracy over manual methods of surveillance	Generative Adversarial Network (GAN).	Raised privacy concerns limiting its adoption
5	Ashwini & Yakapur (2025).	Investigated a surveillance system that is based on deep learning for monitoring	Recurrent Neural Network (RNN)	Depended on high computational resources limiting

		crows and detecting violent behaviour in real time		real-time performace.
6	Zhang et al., (2022).	Created an AI framework for optimizing the hyperparameters of neural networks in smart urban sensing applications that are based on AI	Convolutional Neural Network (CNN)	Couldn't perform in resource-constrained deployments.
7	Jayasudha et al., (2025),	Highlighted how the analysis of gesture and sentiment can help authorities to respond early to potential risks that may break out in crowds that are dense.	Deep learning.	Was too expensive to implement.
8	Zhu et al., (2024).	Proposed a framework for crowd analytics that was based on AI for rail transit stations.	Deep SORT.	Could not support camera changes.
9	Maharajpet & Hegde (2025),	Developed a system that estimated the density of a crowd in real-time	CSRNet and YOLOv8	Could not support videos from high lightning conditions.
10	Basunduwah et al., (2024),	Proposed a model that is multi-visual for the sole purpose of predicting crowd and estimating the its density	YOLOv8	Relied on complex multi-modal data extraction.
11	Ilyas & Bawany (2025).	Developed a framework based on AI for recognising the behaviour of a crowd	Deep learning	Was too expensive to implement and deploy.

		automatically with the use of videos from surveillance cameras		
12	Macriga et al (2024)	Proposed a framework that is based on both AI and ML which was tailored to the urban environment of India and was aimed at improving the management of crowd, preventing crime, and supervision of workplace	Machine learning	The system had a limited scope
13	Cheong et al., (2019).	Reduced dependence on manual surveillance by providing automated human traffic analytics.	Object recognition.	Can not function in an environment that is highly dense.
14	Thangamani et al., (2025),	Examined the effective use of deep learning with main focus on the Convolutional Neural Networks (CNNs), for counting crowds in the surveillance of campus environments	Convolutional Neural Network (CNN).	Was trained to only accommodate large crowd.

2.5 CONTRIBUTION TO KNOWLEDGE

This project work contributes to the domain of research by presenting a practical implementation of an AI-based crowd density estimation system that is comprehensive and which goes beyond theorical model development. In contrast to many studies that are currently existing which focuses solely on algorithmic performance, this project closes the gap by integrating the whole process of video acquisition, preprocessing, estimation of density, alert generation and logging into a unified end-to-end system. The project however, proves that density-based artificial intelligence techniques can be very effective when applied to a crowd monitoring scenario in real world where traditional techniques fail due to occlusion and high congestion. This project also surges forward to ditch the existing lacuna in knowledge by emphasizing practical deployment considerations such as processing in real-time, system performance, and preservation of privacy, thereby bridging the gap between academic crowd analysis models and operational crowd management systems that are used in monitoring public safety during public events.

CHAPTER THREE

SYSTEM ANALYSIS AND DESIGN

3.1 OVERVIEW

Poor method of tacking academic records has for long now remained a factor that is significant in reducing the graduation rates of students across the education in the higher institutions, as the universities often face retention challenges that are severe when academic struggles are not detected early enough. The classification of degree therefore serves as a milestone that is critical for students that are in tertiary institutions thereby making the prediction of final results essential for an academic intervention that is timely. However, this chapter of this project focuses primarily on the analysis and design of the predictive system that makes use of the methodology of Object Oriented Analysis and Design and also the diagrams of the unified modelling language. It therefore explains the architectural design of the system that is internal, evaluated the solutions of academics that is forecastable, identifies their limitations that are structural, and then presents the models of design that guide the implementation of the systems of machine learning.

3.2 SYSTEM ANALYSIS

System analysis on its own entails the act of studying the crowd monitoring process that are already exiting in other to identify limitations and define the requirements of the proposed system. It involves breaking down the proposed system and carefully considering the induvial components and how they interact with one another. It therefore covers the detailed modelling of all the different processes that are taking place withing the system that have been proposed in this project work with the use of the use case diagrams, class diagram, and athe activity diagrams respectively.

3.2.1 ANALYSIS OF EXISTING SYSTEMS

- **Alsubai et al., (2024) Crowd Density Analysis System:**

  The system made use of use of models of deep learning to analyze and classify the how dense a crowd is using images. The system made use of CLAHE for enhancing images, inception v3 for extracting features from images, and GRU for classifying the density of crowds, it also made use of optimization algorithms for tuning hyperparameters.

- **Dheepak et al., (2025) – Smart Crowd:**

  The system was an AI-driven framework for monitoring the density of a crowd in real-time in large public gatherings in India. The system was built to combine CCTV, footages from drone, and signals from Wi-Fi/Bluetooth. It was designed to make of YOLOv5 for detection, CNN-based density mapping, and LSTM for predicting congestion. The system was implemented on edge devices, and at that achieved real-time performance with low latency.

- **Zhu et al., (2024) crowd Analytics framework:**

  The system was built to be used on transit stations so as to support the planning of operations and also manage and enforce safety protocols. The system integrated the use of YOLO and Deep SORT for detecting an tracking individuals from video footage, enabling it to accurately estimate crowd density, flow volume, and the walking speed of people in environments that are of high density. The system however performed well when and achieved a realistic density-speed and performed with great counting accuracy.

- **Maharajpet & Hegde (2025)crowd density estimation system:**

  The system estimated the density of a crowd in real-time. it also combined the use of YOLOv8 for detecting people with CSRNet for estimating density using video feeds from CCTV and drones. The system however functioned by classifying the density of crowd into four different levels and in the end present the results through a web-based dashboards using heatmaps, analytics reports and historical data for supporting decision. The system was tested using benchmark datasets and videos from real world, and the result showed highly accurate detection and low errors in counting.

3.2.2 WEAKNESSES OF EXISISTING SYSTEMS

- **Alsubai et al., (2024) Crowd Density Analysis System:** The shortfall of this system sterns from the fact that the system failed to implement any form of safety alerts.
- **Dheepak et al., (2025) – Smart Crowd:** The weakness of this system however comes from the system being limited to India.
- **Zhu et al., (2024) crowd Analytics framework:** The system had a pitfall as the system was not being able to support camera changes.
- **Maharajpet & Hegde (2025)crowd density estimation system:** The weakness of this system however comes from the system not being able to support videos from high lightning conditions.

3.2.3 BENEFITS OF THE PROPOSED SYSTEM

The AI-Based Crowd Density Estimation System that is proposed in this project presents significant benefits with respect to safety, efficiency, and support for decision making. The system completely eliminates the subjectivity and limitations that are associated with monitoring crowds manually by automating the process of estimating the density of a crowd. It therefore provides a continuous assessment of crowd conditions that is accurate, even in environments that are highly congested where detecting individually is not really possible. However, the system's use of density maps and heatmaps helps security personnels and event managers to quickly understand the distribution pattern or a crowd and also identify the areas that are potentially risky. This, therefore helps in enhancing situational awareness and also allows for timely interventions before the density of a crowd reaches levels that are dangerous.

In addition, the proposed system will help to improve the efficiency of operation by supporting crowd analysis in real-time with minimal involvement of humans. The alert mechanism on the other hand helps to ensure that crowd build up that are critical are detected early, thereby reducing the spent before responding during emergency situations. The system was designed to have a modular system architecture which gives room for scalability and also allows the system to be integrated easily with the surveillance infrastructures that are already existing, thereby making it suitable to be deployed in diverse environments like stadiums, transportation hubs, religious centres, and smart city applications. Furthermore, the system is designed to focus mainly on aggregate crowd behaviour rather than tracking the identify for people, thereby preserving the privacy of individuals and in turn making the system ethically stable for monitoring in public environments.

3.2.4 ANALYSIS OF THE PROPOSED SYSTEM

The system that have been proposed in this study have been designed as a modular video analytics platform that is intelligent and which processes visual data in a way that is structured and sequential. At the very first step of the systems operations, video streams or recorded footages are captured from surveillance cameras and is then passed into the processing module of the system. This processing module now performs some operations that are essential on the footage such as extraction of frame, normalization of resolution, noise reduction, and correction of perspective. These steps are taken to make sure that the data that is entered into the system is progressive and also inline for an analysis that is accurate, notwithstanding the different variations in the quality of the camera or conditions of the environmental.

However, after preprocessing the video, the frames which are now enhanced are forwarded to the crowd density estimation engine the stands as the intricate intelligence centre of the system. The engine now applies an AI-based model which have been trained using sample datasets, to generated density maps that shows the spatial way in which people are distributed within each frame. The values are now integrated into the density maps and with it, the system computes an estimated crowd count. Unlike the traditional methods which are solely based on detection, this method is very effective in scenes that have heavy occlusion and individuals that are overlapping each other, thereby improving the robustness and reliability of the system.

Moving on, the output that is gotten from the density estimation engine is then analysed by the crowd analysis and alert module. This module now classifies the density of the crowd into predefined levels such as low, medium, high and critical based on the thresholds which was configured during the training of the model. After the classification, visual results which includes heatmaps and different numerical indicators, are now displayed through the user interface for easy interpretation. Simultaneously, as this is going on, alert notifications are triggered when the density of the crowd are detected to be unsafe and also, all the relevant data is now stored in the logging module for future analysis and reporting.

3.3 SYSTEM REQUIREMENTS

System requirements is a term that points to the actions that a system should be able to execute or what quality a system is supposed to possess.

3.3.1. Functional requirements of the proposed system:

The functional requirements of the proposed system refer to all the functionalities that should be present in the in the proposed system. These functional requirements are not limited to the following:

a. The system should be able to accept live video streams or prerecorded video footages and extract carefully extract frames from the videos at defined intervals
b. The system should be able to preprocess video frames to enhance its quality and also generate crowd density maps from the video frames.
c. The system should be able to estimate the crowd count form density maps and therefore classify the density level of the crowd.
d. The system should be able to trigger alerts when the thresholds that were predefined are exceeded.

3.3.2. non-functional requirements of the proposed system:

The non-functional requirements of this system points to the qualities and attributes in which the system must have in other for it to be able to function properly. These non-functional requirements of the system that have been proposes in this project work are:

a. The layout of the proposed system must be intuitive and straight forward in other to allow users to navigate and operate the system efficiently
b. The system be scalable and be able to handle video streams while ensuring reliability and fault tolerance.
c. The system should not track the identity of people and must maintain data privacy.
d. The system should support the deployment on local servers and also on edge devices.

3.4 PROPOSED METHODOLOGY: OBJECT ORIENTED ANALYSIS AND DESIGN

Object-Oriented Analysis and Design (OOAD) is a term that is used to refer to the process software development that makes use of object-oriented concepts to create and design of software systems. It involves first identifying the different requirements of the system. After which the myriad classes that are involved in the system and how they collaborate with each another in other to keep the system functioning. However, this methodology makes use of a lot of practices and techniques for analysing system which includes, use cases, object-oriented programming, and UML diagrams. OOAD therefore uses object-oriented programming for the main conceptualization and development of software systems because UML diagrams are essential tools for demonstrating the numerous aspects and relationships among the different components that are within a software system.

3.4.1 UNIFIED MODELLING LANGUAGE

The unified Modelling Language (UML) diagrams is a schematics diagram that provide a vital and standardized framework for the purpose of mapping the intricacies of a systems architecture visually. These diagrams are graphical abstractions which enable architects, developers, and stakeholders to be able to conceptualize and communicate structural designs collaboratively. However, among these diagrams, the use case diagram is arguably the most critical as it delineates boundaries between external actors and internal operations explicitly and with this, the use case diagram establishes the behavioural blueprint for deciphering functional system requirements in a very essential manner.

3.4.2 USE CASE DIAGRAM OF PROPOSED SYSTEM

The use case diagram is one of the UML diagrams that illustrates the functional interactions that take place between the users of the system and the AI-Based Crowd Density Estimation system that is proposed in this project. It therefore identifies the primary actors, such as security personnel and system administrators and the specific tasks that they can perform within the system. These tasks however include switching on the system, uploading or streaming video footage, estimating crowd density, alerting the user and so on.

Use case diagram:

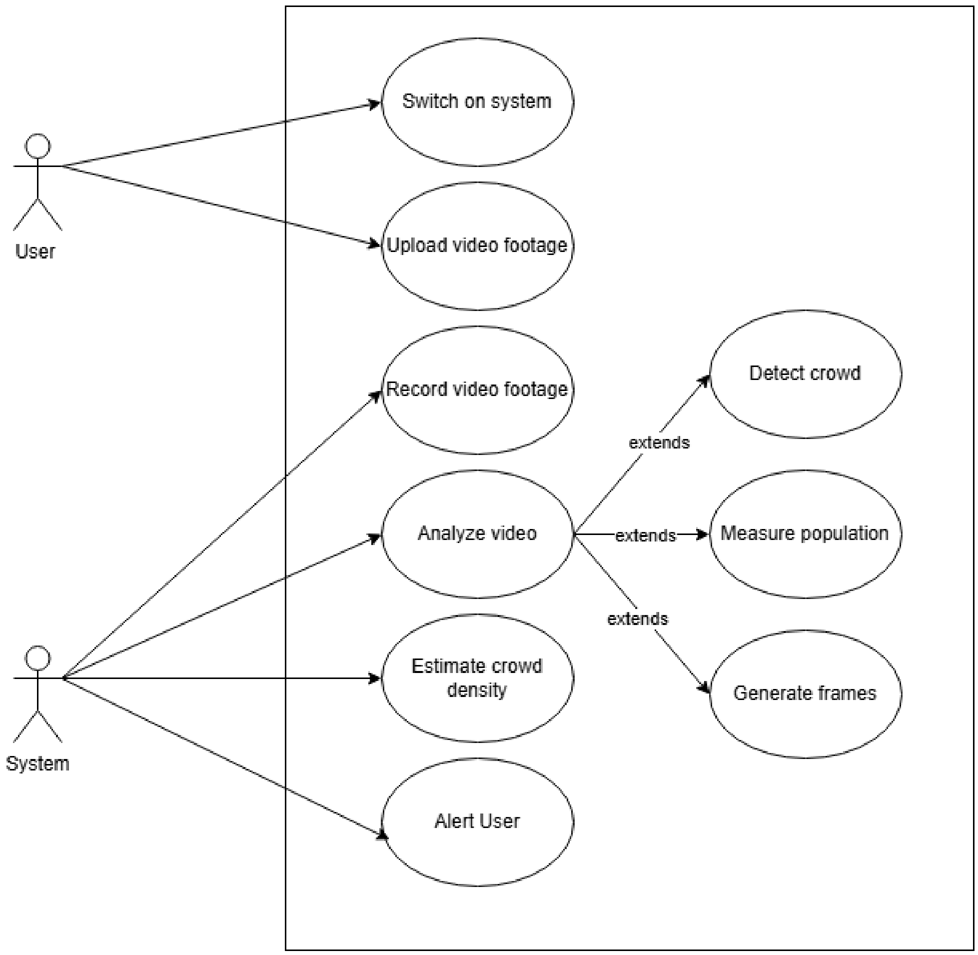


Figure 3.1 The proposed system's use case diagram of

The diagram that is attached portrays clearly a clear view of the different workings of the system that have been proposed in this project in a high-level, which clearly shows how users work togheter with the system to achieve specific objectives. It helps in defining the system boundaries and also ensures that all the functional and non-functional requirements that are required of the system are met. The diagram however shows the different actors (system and the user) and the actions that they perform which includes switching on the system, uploading or streaming video footage, estimating crowd density, alerting the user and so on.

3.4.3 ACTIVITY DIAGRAM OF THE PROPOSED SYSTEM

From the point of view that is structural, the use case diagrams is seen to comprise of four core elements which include the use cases who's function is to outline the system's discrete functionalities, the actors that denotes the external entities that play different roles in the system, the associations which functions to map the mutual interactions that happen between the actors, and the system boundaries which established the architectural scope. It therefore sows the workflow of operations that are involved in the crowd density estimation system that is proposed in this study. It describes in detail the sequential flow of activities from the point of acquiring the input video, through the preprocessing processes, estimation of the crowd density and down to the point of generating real-time safety alerts for users when the crowd exceeds the limit threshold. The diagram however highlights the point of decision making such as the classification of density and the triggering of alerts, chowing how the system will respond to different crowd conditions. However, by modelling the operational flow, the activity diagram provides a clear understanding of how the system functions internally and how the various processes are coordinated while the system is executing.

The proposed system's Activity diagram:

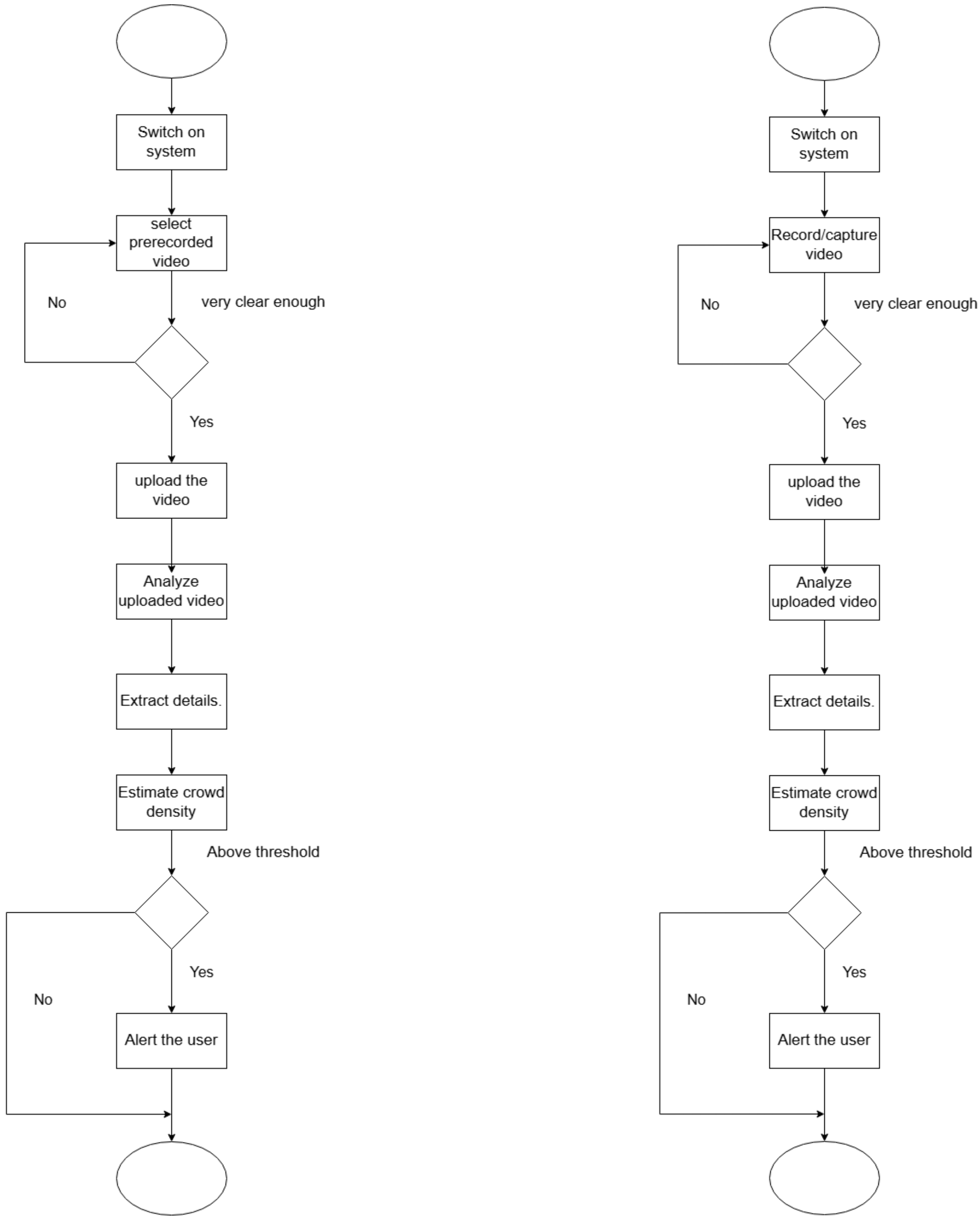


Figure 3.2 The System's operating with a prerecorded video and capturing video footage.

The operational workflow of the system is shown in detail in the activity diagram above, which stands to provide a step by step representation of all the processes that are going on internally in the system in a visual manner. The sequence of processes states when the user initializes the system and then either uploads a video file that have been pre-recorded or activates the capturing of live video. The process therefore culminates when the system have successfully analysed the footage that was uploaded or the one it captured and then transmits an alert about the crowd density to the user.

3.4.4 CLASS DIAGRAM OF THE PROPOSED SYSTEM

The internal configuration of the software system that is structural is represented using a class diagram of the Unified Modelling Language. The diagram therefore provides the reader with a blueprint of the system that is very detailed by organizing the data of the system into classed and subclasses. Specifying the attributes which are distinct and also the methods of the system. furthermore, it shows how these different entities of the system interface and collaborate with each other to drive the core capabilities of the system.

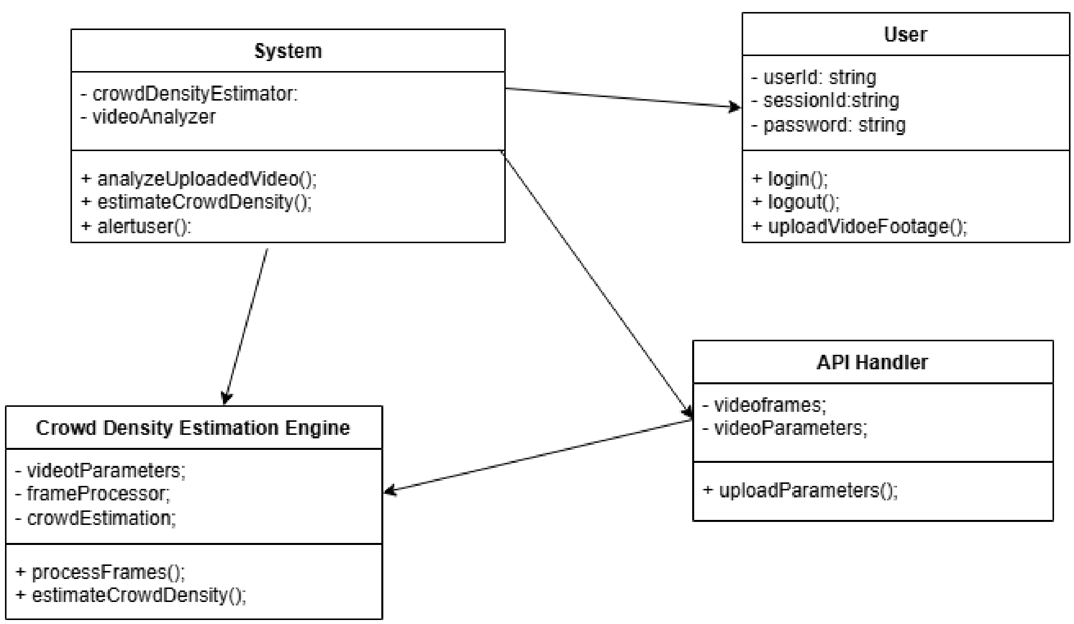


Figure 3.4 The proposed system's class diagram of

The UML class diagram that is presented above outlines the foundation of the proposed system, illustrating how various operations are organized and managed. It therefore provides insights that are critical into the dependencies, interconnections, and relationships that are structural which are established between the diverse components of the system. consequently, this model enables a deeper comprehension of the underlying software architecture and contextual design.

3.4.5 UML SEQUENCE DIAGRAM

The unified Modelling Language (UML) sequence diagram is an illustration that offers a temporal representation of a systems behaviours and which functions to map the discrete interactions of objects in a chronological other within the system. However, this architecture that is depicted by the sequence diagram is made up of lifelines that denote the entities that are participating in the system execution, the messages that represent the communication pathways between them, and the vertical activation bars which functions to quantify the exact duration of the object's lifecycle during the process of execution. Moreover, this sequence diagram captures the chronological dimension of the system and at this, helps the developers and stakeholders that are non-technical to gain an overview of both the sequence and precise timing of the systems operations in a way that is very transparent.

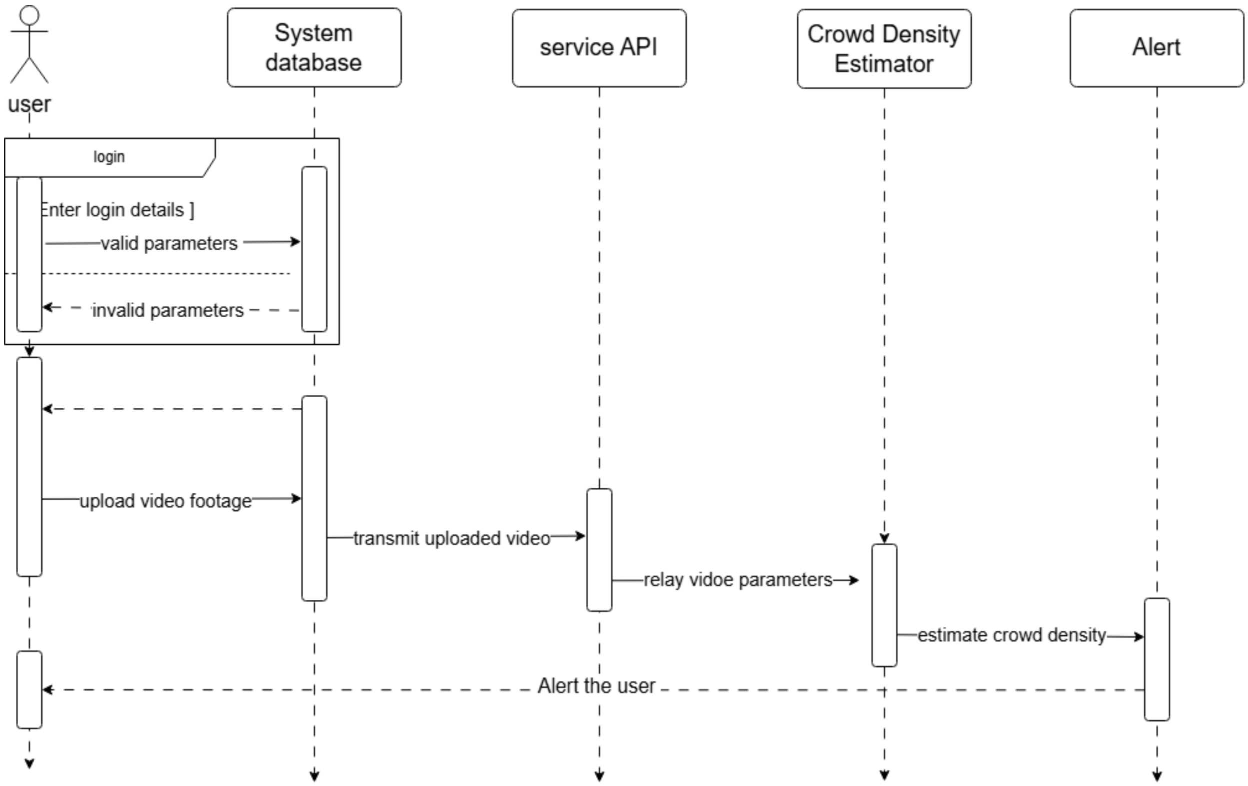


Figure 3.6 The proposed system's Sequence diagram

The illustration that is presented above is used to illustrate the chronological interaction between the various component parts of a system. It is used in this project to illustrate the sequential process of execution of the different processes that goes on in the system that have proposed from the moment that the system is turned on to the point of it alerting the user. It shows the serial order in which it uses to execute and the particular part of the system that is active at any moment in time.

3.5.1 INPUT INTERFACE DESIGNS OF THE PROPEOSED SYSTEM:

This section outlines how the system hopes on satisfying the requirements that have been detailed previously by presenting the design of the user interface of the proposed system for the input module. This user interface is built to ensure that the features of the platform remain highly accessible, this user interface layout therefore provides an interface that is straightforward where users can input the metrics of students easily. This interface sketch is displayed below:

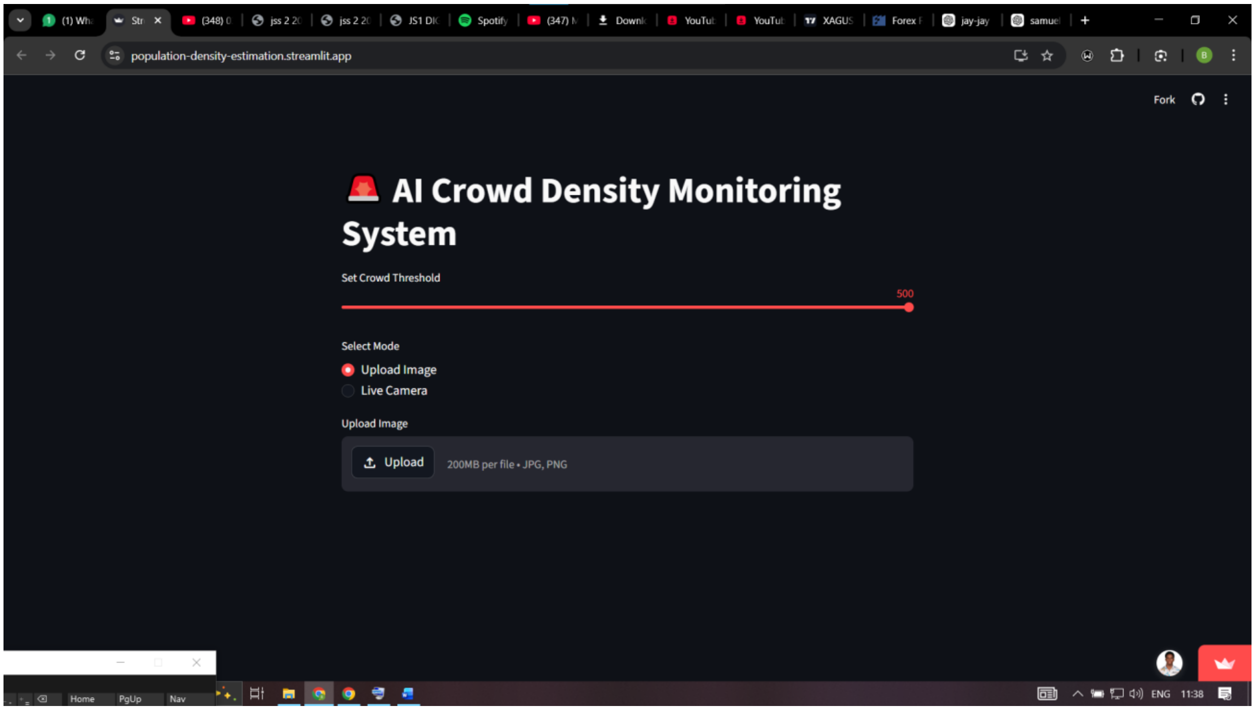


Figure 3.6 Input designs for the proposed system

The diagrammatic illustration that is present above stands to illustrate the design of the user interface for the proposed system from which the user gets to supply the prerecorded video to the system. This involves the user supplying the data that is required by the system for it to estimate the crowd density in the video.

3.5.2 OUTPUT INTERFACE DESIGN OF THE OF THE PROPOSED SYSTEM:

This section of this project work demonstrates how the system tends to fulfils the requirements of the proposed system that is structural by presenting the layout for the module for output. The module is therefore designed to make the insights of the platform to be easily accessible, and therefore the layout displays a screenshot of the output interfaces in other to show exactly how the proposed system intents to present the result of its computations after estimating the crowd density of a particular video footage. The design of the output interface part of the system that have been proposed in this project is outlined below

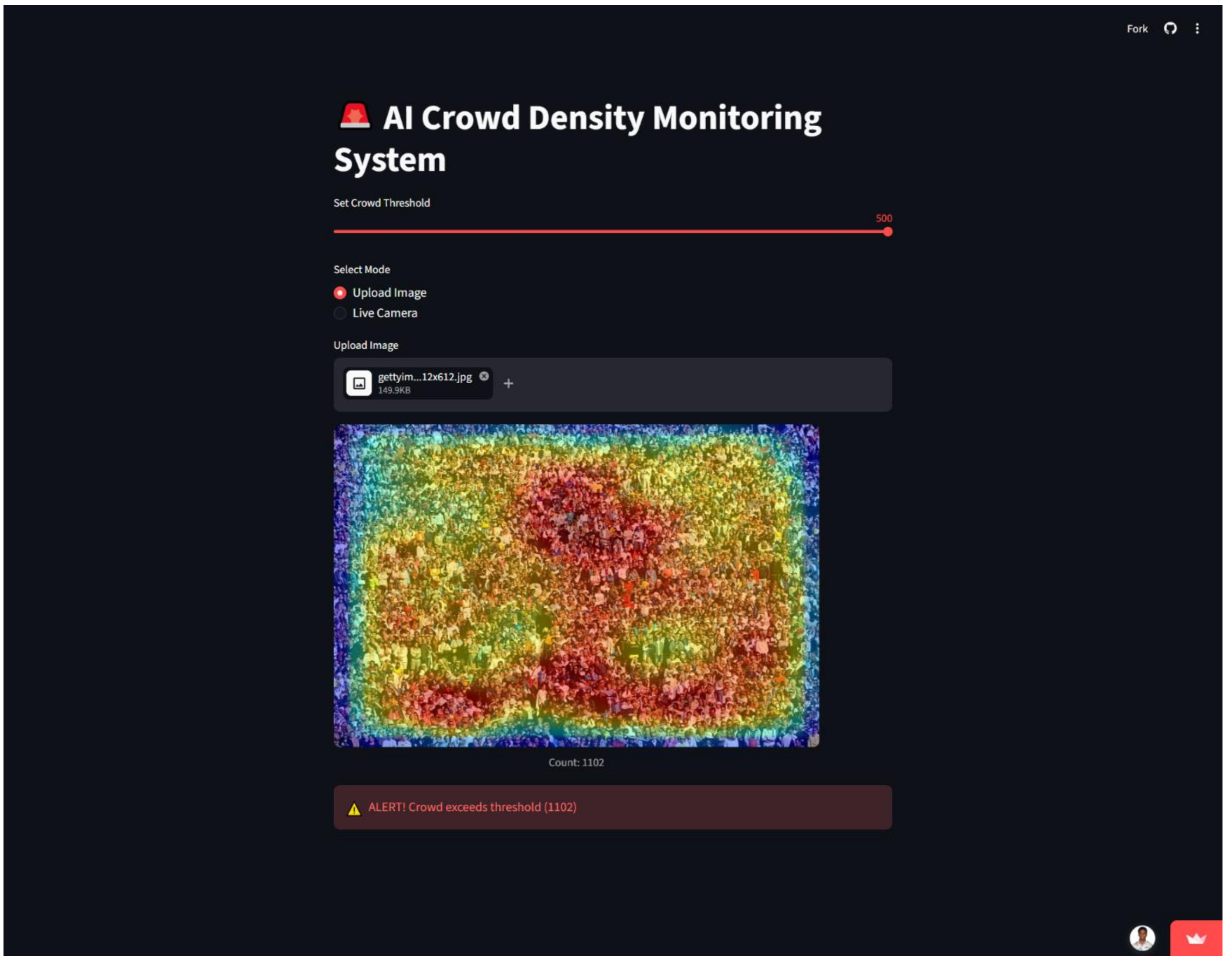


Figure 3.7 Output designs for the proposed system

The figure that is presented above shows the output interface that have been designed for the proposed system. specifically, these user interface mockups demonstrate the way that the results that are generated after calculating the density of a crowd is formatted and arranged . overall, this section of this project work provides a representation of how the processed data is rendered on the screen for the end user to see graphically.

3.7 DATABASE DESIGN

This section of this project discusses the architectural schema which provides a blueprint of the target database in a way that is abstract by detailing the structural tables that are designated for a continuous storage of data. However, this database design maps out the necessary system entities alongside their corresponding attributes to establish the schema of the proposed system application that is robust by being implemented as model that is relational, leveraging the querying framework of MySQL.

Database schema for the proposed system

Table 3.1 User table for the proposed system

Attribute	**Data Type**	**Description**
Id	Integer	primary key
Username	varchar	The username of the user
Email	varchar	The email of the user
Password	varchar	The password of the user
Date_joined	timestamp	The date a user signed up to the platform.

Table 3.2 Crowd estimation table of the proposed system

Attribute	**Data Type**	**Description**
Id (pk)	Integer	Primary key
Crowd_Estimation	varchar	The estimated crowd density
User	varchar	The user that made use of the system.
Session	timestamp	The session of the user.

Table 3.3 Alert table of the proposed system

Attribute	**Data Type**	**Description**
Id (pk)	int	Primary key
Crowd_Estimation	varchar	The Estimated crowd density
Alert	varchar	Crowd density alert
Session	timestamp	The session of the prediction.

CHAPTER FOUR

SYSTEM IMPLEMENTATION

4.1 introduction

This chapter of this research exists to help explain in detail how the AI-Based Crowd Density Estimation System that was proposed at the prior section of this work was developed and implemented based on the different design interfaces that were presented in the previous chapters. The chapter also describes how the conceptual system was successfully transformed into a fully functional solution that is capable of analysing the density of a crowd from a video data. The chapter also highlights the tools, processes and technologies that were employed during the development process. However, each of the component of the system was selected carefully, tested and also integrated into the system to ensure that the final system is efficient, reliable and suitable for the analysis of crowd monitoring.

4.2 TOOLS USED FOR IMPLEMENTATION (TECH STACK OF THE PROPOSED SYSTEM)

The implementation of the proposed system done with the use of modern frameworks and tools that are suitable for computer vision and web-based applications that work in real-time. These tools and frameworks that were employed when developing implementing the proposed system is discussed in detail below.

4.2.1 Frontend

- **Python:** This is one of the most versatile programming languages and it was used in this project work for backend development, AI model training, and also in building simple test graphical user interfaces.
- **JavaScript**: This is the most popular scripting languages and was adopted in this project work to add interactivity to the frontend of the system.
- **HTML:** This is a simple markup language that was used in the system development process to build the elemental structure of the system's user interface.
- **CSS**: This is a cascading stylesheet that was used in this project work to structure and add styles to the interface of the system.

4.2.2 Frameworks and Libraries

- **OpenCV**: This is a computer vision library that is open source and which contains over 2,500 algorithms that are optimized for real-time computer vision, image processing and machine learning. It was used in the development process of this system for video processing and frame extraction
- **NumPy:** These are two open-source python libraries that are fundamental and essential for data science and numerical computing. They were adopted in the development of this system for the purpose of data manipulation and preprocessing.
- **Streamlit:** This is a Python framework that is open-source, which is used for building web applications that are interactive for data science and machine learning purposes.
- **Torch (pyTorch):** This is a machine learning library that is open-source and is used for applications of computer vision and natural language processing.

- **Torchvision:** This is one of the packages that are specialized which belongs to the ecosystem which provided access to popular datasets, architectures of models and image transformations for computer vision that are common.
- **Pillow:** This a modern fork of the Python Imaging Library (PIL) which is actively maintained. It is very meticulous as it adds robust image processing capabilities to the Python interpreter in other to be able to support extensive file formats, internal representations that are efficient and a pixel manipulation tool that is direct.

4.3 DEVELOPMENT PALTFORM AND HARDWARE REQUIREMENTS

The process of developing and implementing this system was done using the dedicated Jupiter Notebook extension that is integrated into the visual studio code Integrated Development Environment (VS Code IDE). Also The python programming language was employed alongside other frameworks as it offers huge support for data analysis and machine learning. Moreover, the hardware specifications of the personal computer that was used in the development of the proposed system includes the following:

- **Processor:** intel Core i5 or equivalent
- **RAM:** 8GB minimum
- **Storage:** At least 20 GB available disk space
- **GPU;** Was not really required as the CPU-based acceleration in Google Collab was sufficient

Note: The above listed are the minimum requirements that are required for the system to function, but in other for the system to reach its full potential, the following are the recommended requirements:

- **Processor:** intel Core i7 or equivalent AMD Ryzen 7
- **RAM:** 16GB or higher
- **Storage:** At least 20 GB available disk space
- **GPU;** NVIDIA GPU (e.g., GTX 1650 or higher) for faster and smooth detection
- **VRAM:** At least 4GB

4.4 DATASET DESCRIPTION AND PREPROCESSING

4.4.1 Dataset Description

The researcher trained the model and evaluated it using the ShanghaiTech Crowd Counting Dataset, which was made up two parts that are distinct.

- **Part A (Dense Crowd Dataset – Cross Evaluation):** This part of the dataset consisted of 300 training images and 182 test images. However, this first part of the dataset was characterised by crowds that are highly dense, an occlusion that is severe and a scale variation that is significant.
- **Part B (Sparse Crowd Dataset – Training):** This part of the dataset was made up of 400 training images and 316 test images. However, this part was characterised by Low to medium crowd density, an occlusion that is minimal and the heads in the images were clear and visible. However, it is important to note that this was the dataset that was used in training the model.

4.4.2 Data Processing

Before the process of model training started, the dataset that was obtained from online repositories was transformed in other to prepare it and make it suitable for machine learning. However, the different steps that were involved in preparing the dataset to make it suitable for machine learning include the following:

- **Resizing:** The different images from the dataset were resized to a fixed dimensions of 640 x 640
- **Normalization:** The values of the pixels in the images were scaled for better convergence.
- **Data Augmentations:** The images from the dataset was augmented by doing the following:
  - Horizontal flipping:
  - Rotation:
  - Brightness adjustment:
  - Scaling:

4.4.3 Training and Validation Split

After the dataset was pre-processed, the second Part B of the dataset was used for training the model while the Part A of the dataset was used for evaluating the generalization of the developed model on crowds that is dense.

4.5 MODEL ARCHITECTURE AND TRAINING

The system made use of deep learning-based crowd density estimation model was built using the convolutional neural networks. However, a detailed description of the model's architecture is as follows:

- **Backbone:** This was created using layers of Convolutional Neural Network (CNN) for the extraction of features.
- **Intermediate Layers:** The part of the model is used for feature refinement and scaling.
- **Output Layer:** This layer of the model is used for generating the density maps.

4.5.1 Training Parameters

Different parameters were adopted during the process of training the models of the system. These parameters include the following:

- **Epochs:** 50 – 100
- **Batch Size**: 16 (adjusted based on GPU memory)
- **Optimizer**: Stochastic Gradient Descent (SGD) / Adam
- **Learning Rate:** 0.001
- **Loss Function:**
    - Mean Squared Error (MSE) for the estimation of density

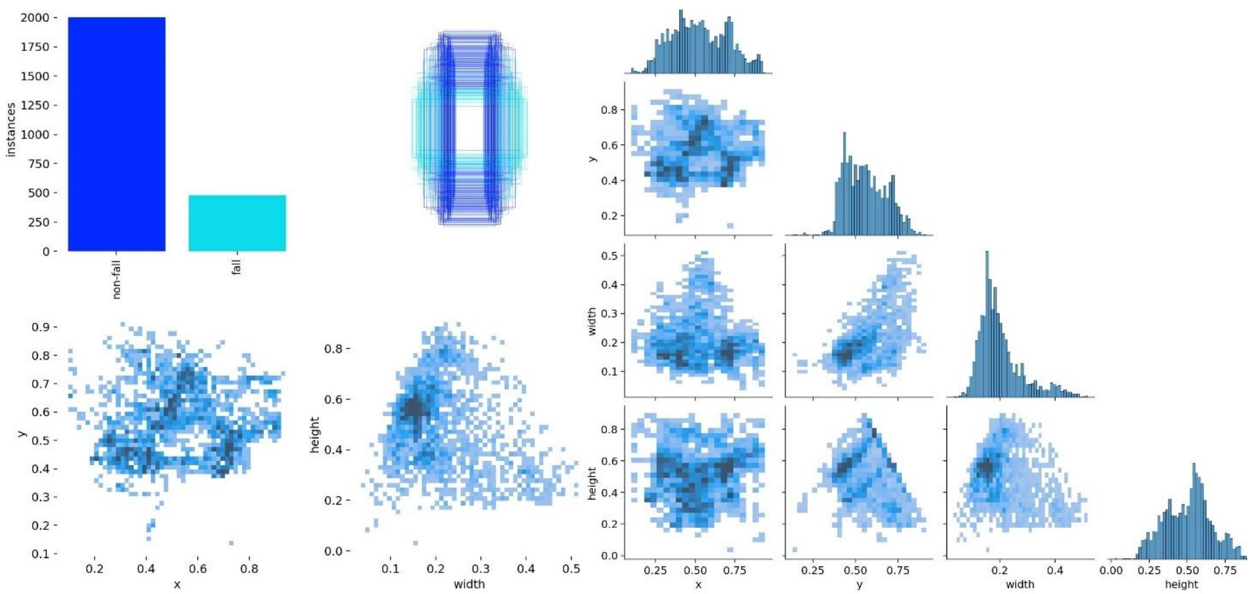


Figure 4.1 diagram of the model's labels and labels correlogram.

The labels however represent the bounding boxes, class names, or segmentations that were used in the identification of dense crowds in real time as the model is running. But at the end of the labelling, the system was trained using images from the dataset.

Figure 4.2 sample images used in training the model

4.6 SYSTEM IMPLEMENTATION AND MODULES

The system was designed to work in two ways: using real-time video feed and making use of uploaded images. However, system is made up of different modules which work together hand in hand to enable the system achieve its full potential. The different modules that are involved in the system include the following:

- **Video input Module:** This module enables the system to capture images from the camera or video stream and sends frames from the video to the backend for processing.
- **Frame Processing Module:** This module of the system works by extracting frames from images that have been captured by the system's camera and however, prepares these frames for model inference.
- **Density Estimation Module (Core AI Engine):** This module generates density maps and crowd counts from the images.
- **Analysis Module:** This module is responsible for classifying the crowd levels into any of the four classes (Low, Medium, High and Critical)
- **Visualization Module:** This is one of the most important modules in the system which functions by displaying heatmaps and crowd statistics.
- **Alert Module:** This module is responsible for telling the user when the density of the crowd in a captured or uploaded image exceeds the expected threshold.
- **Frontend Dashboard Module:** This module of the system houses the interface visual of the system which grants the user the opportunity to effectively use the system. It however functions by displaying the live video feed and alerts.

However, even as the system is made up of a collection of modules, these modules still coexist depending on each other in other to enable the system reach its full potential and perform the actions to which it was developed.

4.7 TESTING, EVALUATIONAND RESULTS

4.7.1. Evaluation of the proposed system

In other to determine the effectiveness of the model's prediction, several machine learning evaluation metrics were used. These metrics include the Accuracy, Latency, Frames Per Second (FPS), Mean Absolute Error (MAE), Mean Squared Error (MSE) and Root Mean Square Error (RMSE). The system was tested under two conditions namely: Sparse Crowd and Dense Crowd.

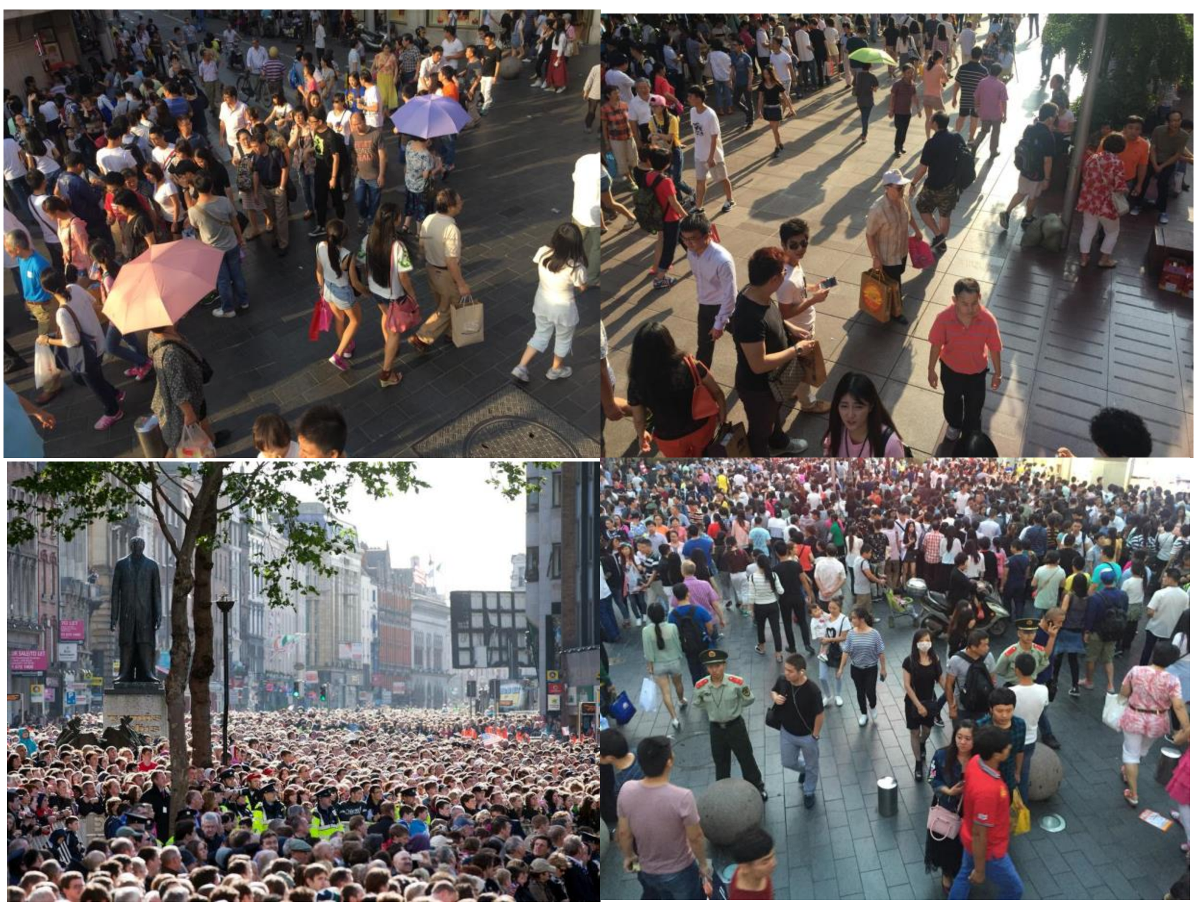


Figure 4.7 image samples used for model evaluation.

4.7.2. User interface Testing of the proposed system

This section represents a component of the system evaluation phase that is critical, specifically focussing on the testing of the interfaces of the application that have been proposed. The process that is detailed in this section ensures that all the visual and interactive elements align with the core technical requirements of this project by validating how each use interface functions and responds, However, the system's interface was cross evaluated using the first part (part A) of the dataset. This testing was therefore done to ensure that the system is capable of satisfying all the objectives that were stated at the earlier section of this project work. Moreover, the model was ale to detect and catch errors and as well prevent the activation of false alarms when necessary.

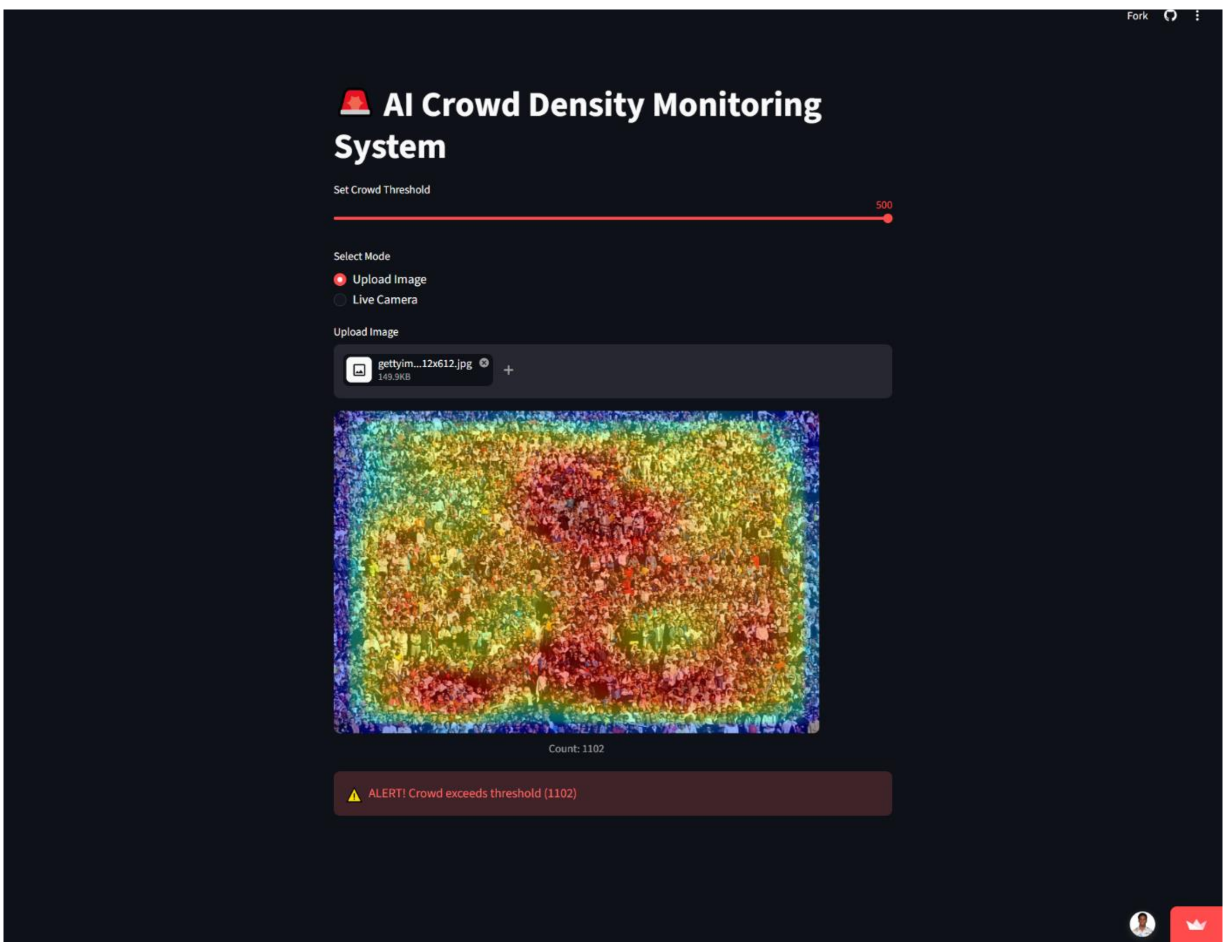


Figure 4.8 The proposed system's input screen of the

4.7.3 Result Presentation and analysis:

This section of this project presents in detail the outcome of the testing and analysis phase of the system that have been proposed in this project. It displays how the system tends to present the results that was gotten while testing the input interfaces of the proposed system. The result section however will display the values of the testing parameters that were gotten at the end of the testing phase.

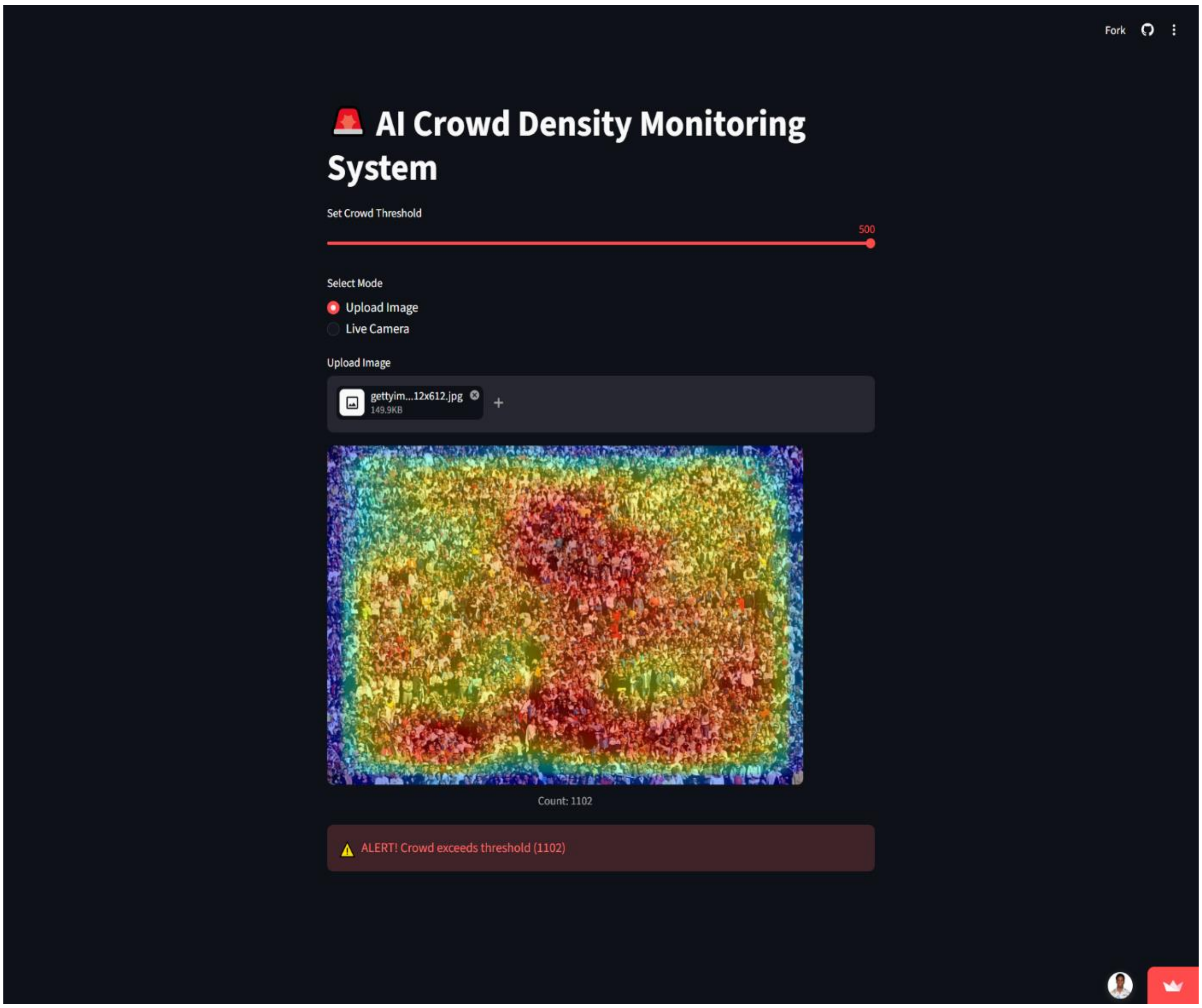
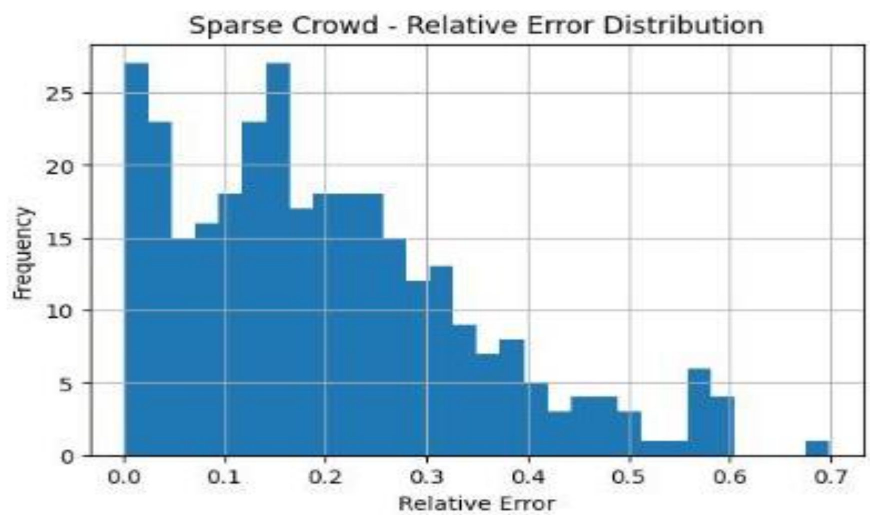


Figure 4.9 The in result of computations proposed system.

4.8 Discussion and Results

At the end of the development phase of the system, the system was further tested based on the parameters and evaluation metrices which have been outlined in the previous section of this chapter using live data from class halls and seminar rooms. However, at the end of the testing phase, the results that were gotten signified that the system was very effective and was able to satisfy all the underlying reasons to which it was conceptualized and built at the first place. Moreover, the results gotten while testing the system are shown below:



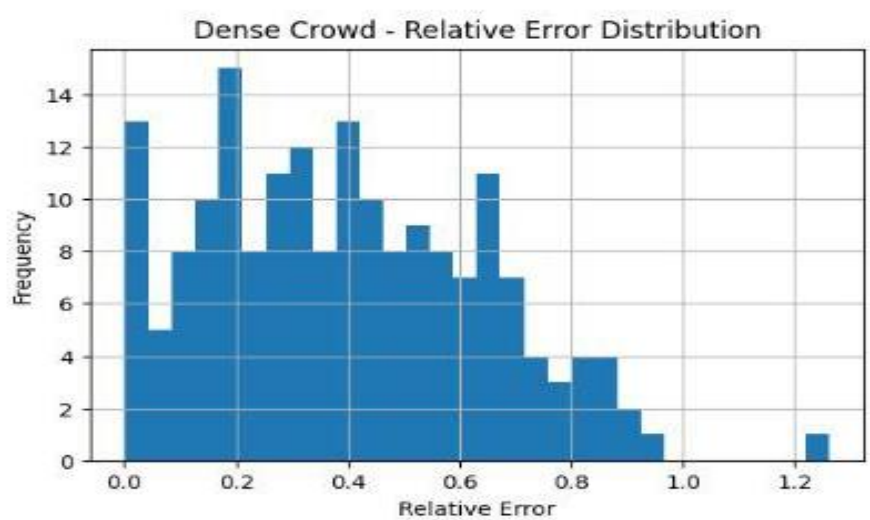


Figure 4.10 The Dense and Sparse Crowd Relative Error Distribution (RED) graph

From the graph above, in sparse environments, Error distribution is concentrated around lower values, indicating that the model was able to make stable predictions in environments that are less crowded. While in dense environments, error distribution is widely spread, indicating that the model had higher variance and made frequent large prediction errors in environments that had crowd that is dense.

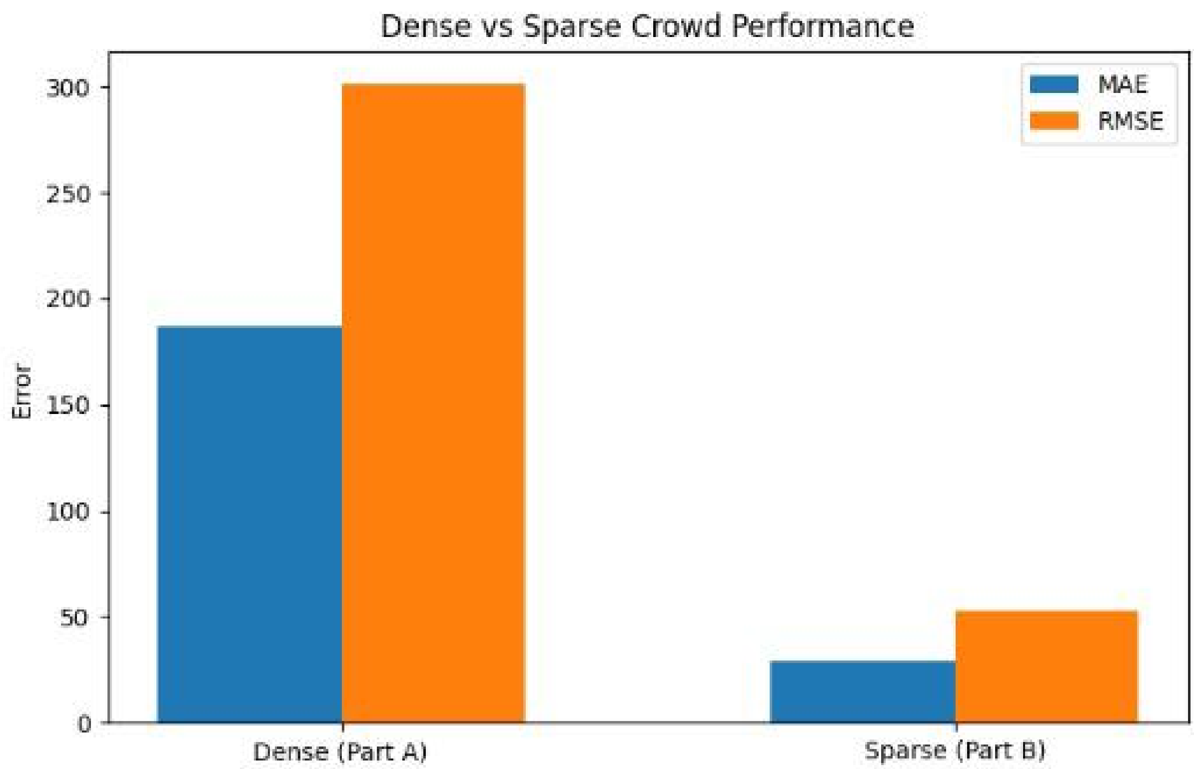


Figure 4.11 The comparison of error metrics between sparse and dense crowd datasets.

However, as part of the testing process, the performance of the model was compared in two environments namely the sparse and dense environments. In scenarios that are dense, the model shows significantly higher error, occlusion and crowd complexity than in sparse environments. Moreover this difference is expected due to the fact that there is severe occlusion in dense crowd, a lot of scale variation as people appear smaller and kind of overlap with each other, and most especially, there is a lot of difficulty in distinguishing the regions of individual heads. But despite the difference in the accuracy of the system, the model maintains its capabilities of ~15–19 FPS on NVIDIA T4 GPU ~0.05 sec latency per frame in real time.

In addition to the above, thee model demonstrates significantly better performance on sparse crowd scenes (Part B) compared to dense crowd scenarios (Part A). where in sparse environments, the model achieves:

- Lower MAE (29.13)
- Lower RMSE (52.87)
- Higher accuracy (26.9%)

While in dense environments, performance degrades:

- MAE increases to 186.58
- RMSE rises sharply to 301.26
- Accuracy drops to 12.64%

In summary, the result gotten at the end of the testing process is displayed in the table below:

Metric	Sparse Crowd (Part B)	Dense Crowd (Part A)
MAE	29.13	186.58
MSE	2794.97	90,758.08
RMSE	52.87	301.26
Accuracy(%)	26.90%	12.64%
FPS	18.76	15.62
Latency(sec)	0.0516	0.0549

CHAPTER FIVE

SUMMARY, RECOMMENDATIONS AND CONCLUSION

5.1 OVERVIEW

This section of this project work of research exists as one of the most significant chapters of the project, as it is the chapter that presents a synopsis of this project work of research in a detailed manner along side the recommendations for further improvements of the model that have been developed in this project in the future as well as the final conclusion of the project. The chapter however highlights the key contributions and findings that was achieved in the study.

5.2 SUMMARY OF THE PROJECT

This work of research is geared mainly towards the modelling and implementation of an AI application that is intelligent for analysis of a crowd's density in public and private areas. This project therefore started by the identification of the limitations that is evident in the traditional methods of detecting and monitoring the density of crowd in public and private gatherings and the need for an automated solution.

However, moving forward on the aim that have been specified, the system was implemented on the later sections of the project with the application of the different techniques of both machine learning and computer vision. The system that was developed was therefore constructed in a way that it integrated the real-time monitoring of crowd density and alerting the user when the density of a crowd goes above the specified threshold. In addition, the researcher also adopted the UML diagrams during the design phase of the software development life cycle of the application, describing all of the internal architecture and interactivity of the application in detail.

Moreover, when the development of the system was complete and the system was tested, the results showed that the system can effectively detect the density of crowd and provide timely alerts about the density of the crowd in a gathering in other to tackle stampede and problems that are related to congestion.

5.3 RECOMMENDATIONS

This system that has developed as a result of this research, can be improved further by integrating deep learning models that are much more advanced to increase the accuracy of detection. However, the works that are coming after this one can also improve a smart mobile application so as to give support for remote monitoring by site administrators and officials. But in addition to this, the system that have been completed in this study can also be extended to support more safety features to help prevent crowd related hazards and accidents.

5.4 CONCLUSION

At last, at this point, we have gotten to the end project as this part of this project concludes the on all the objectives that were stated in the earlier section of the project. However, this research has successfully used modern technologies to develop an AI crowd estimation system. The system therefore addresses key challenges in monitoring crowd density level by providing the detection and alert mechanisms automatically.

Although the current system has been developed to meet all the objects of this project research, there is still more room for further improvements and expansion and with continued update and development, the system can become a standard tool for ensuring safety in areas of public gatherings.

REFERENCES

- Akhtar, R., & Malhotra, D. (2022). Intelligent techniques for crowd detection and people counting—A systematic study. In *Computer Vision and Robotics: Proceedings of CVR 2021* (pp. 119-130). Singapore: Springer Singapore.
- Alrowais, F., Alotaibi, S. S., Al-Wesabi, F. N., Negm, N., Alabdan, R., Marzouk, R., ... & Al Duhayyim, M. (2022). Deep transfer learning enabled intelligent object detection for crowd density analysis on video surveillance systems. *Applied Sciences*, *12*(13), 6665.
- Alsubai, S., Dutta, A. K., Alghayadh, F., Alamer, B. H., Pattanayak, R. M., Ramesh, J. V. N., & Mohanty, S. N. (2024). Design of artificial intelligence driven crowd density analysis for sustainable smart cities. *IEEE Access*, *12*, 121983-121993.
- Ashwini, K. B., & Yakapur, B. (2025, November). AI-Driven Surveillance System for Crowd Safety and Monitoring. In *2025 9th International Conference on Computational System and Information Technology for Sustainable Solutions (CSITSS)* (pp. 1-7). IEEE.
- Basthikodi, M., Vidya, B., Pinto, E. M., Basith, M., & Rao, S. A. (2024, July). AI Based Automated Framework for Crime Detection and Crowd Management. In *2024 Second International Conference on Advances in Information Technology (ICAIT)* (Vol. 1, pp. 1-6). IEEE.
- Basunduwah, E., Mohzary, M., Song, S., & Choi, B. Y. (2024, December). AIM-CP: AI-Based Multi-Visual Components Analysis for Crowd Dynamics Prediction. In *2024 39th International Conference on Image and Vision Computing New Zealand (IVCNZ)* (pp. 1-6). IEEE.

- Cheong, K. H., Poeschmann, S., Lai, J. W., Koh, J. M., Acharya, U. R., Yu, S. C. M., & Tang, K. J. W. (2019). Practical automated video analytics for crowd monitoring and counting. *IEEE access*, *7*, 183252-183261.
- Dheepak, G., Vishwa, I., Alex David, S., & Sakthi Karthi Durai, B. (2025, August). Smart Crowd: AI-Driven Crowd Density Monitoring and Management in Indian Public Hotspots. In *2025 3rd International Conference on Sustainable Computing and Data Communication Systems (ICSCDS)* (pp. 651-657). IEEE.
- Fan, Z., Zhang, H., Zhang, Z., Lu, G., Zhang, Y., & Wang, Y. (2022). A survey of crowd counting and density estimation based on convolutional neural network. *Neurocomputing*, *472*, 224-251.
- Gao, G., Gao, J., Liu, Q., Wang, Q., & Wang, Y. (2025). A survey of deep learning methods for density estimation and crowd counting. *Vicinagearth*, *2*(1), 1-37.
- Hassen, K. B. A., Machado, J. J., & Tavares, J. M. R. (2022). Convolutional neural networks and heuristic methods for crowd counting: A systematic review. *Sensors*, *22*(14), 5286.
- Ilyas, A., & Bawany, N. (2025). Crowd dynamics analysis and behavior recognition in surveillance videos based on deep learning. *Multimedia Tools and Applications*, *84*(23), 26609-26643.
- Jayasudha, M., Chalappuram, J. J., & Edla, B. R. (2025, March). Real-Time Crowd Management and Crime Prevention System Using AI. In *International Conference on Artificial Intelligence, Communication Technologies & Smart Cities* (pp. 557-570). Singapore: Springer Nature Singapore.
- Khan, M. A., Menouar, H., & Hamila, R. (2023). Revisiting crowd counting: State-of-the-art, trends, and future perspectives. *Image and Vision Computing*, *129*, 104597.

- Kiruthika, S., & Suchitra, S. (2023, December). Estimation of Crowd Density in Surveillance Cenes Based on Deep Learning Techniques. In *2023 Intelligent Computing and Control for Engineering and Business Systems (ICCEBS)* (pp. 1-9). IEEE. Patwal, A., Diwakar, M., Tripathi, V., & Singh, P. (2023). Crowd counting analysis using deep learning: A critical review. *Procedia Computer Science*, *218*, 2448-2458.
- Kushwaha, H., & Gupta, S. K. A Comparative Study and Analysis on Crowd Density Estimation Approaches. *Available at SSRN 3985903*.
- Leisha, R., Medows, K. J., Thiruthuvanathan, M. M., Babu, S. R., Divakaran, P., & Chaturvedi, V. M. (2025). Advanced Cyber-Physical Systems Utilizing Deep Learning for Crowd Density Detection and Public Safety. In *Navigating Cyber-Physical Systems With Cutting-Edge Technologies* (pp. 83-122). IGI Global Scientific Publishing.
- Macriga, G. A., Bavyatha, S., Abhinidhi, S. H., & Jagadeesh, N. (2024, October). Crowd Management Using AI & ML. In *2024 International Conference on Power, Energy, Control and Transmission Systems (ICPECTS)* (pp. 1-5). IEEE.
- Maharajpet, S. S., & Hegde, A. V. (2025). Intelligent Real-Time Crowd Density Estimation for Proactive Event Safety: A Machine Learning Approach. *Recent Research Reviews Journal*, *4*, 264-80.
- Mannickathan, G. P., & Shamsu, A. M. (2025, August). Advanced AI-Driven Predictive Framework for Dynamic Crowd Flow Management and Congestion Mitigation in High-Density Environments. In *2025 IEEE 2nd International Conference on Information Technology, Electronics and Intelligent Communication Systems (ICITEICS)* (pp. 1-15). IEEE.

- Padmalal, S., John, K. M., Joseph, T., Dinesh, V., & Vinod, V. (2025, July). A real-time design for dense crowd enumeration via tracking using AI system. In *AIP Conference Proceedings* (Vol. 3260, No. 1, p. 020035). AIP Publishing LLC.
- Sujihelen, L., Subhadra, S., Kota, S. P., & Vignesh, S. (2025, July). Video based crowd analysis using deep learning. In *AIP Conference Proceedings* (Vol. 3257, No. 1, p. 020064). AIP Publishing LLC.
- Thangamani, R., Vimaladevi, M., & KM, S. (2025, June). Deep Learning Based Crowd Counting for Campus Safety Monitoring. In *2025 6th International Conference on Inventive Research in Computing Applications (ICIRCA)* (pp. 1626-1633). IEEE.
- Tyagi, B., Nigam, S., & Singh, R. (2022). A review of deep learning techniques for crowd behavior analysis. Archives of Computational Methods in Engineering, 29(7), 5427-5455.
- Wang, M., Zhou, X., & Chen, Y. (2025). A comprehensive survey of crowd density estimation and counting. *IET Image Processing*, *19*(1), e13328.
- Wang, Mingtao, Xin Zhou, and Yuanyuan Chen. "A comprehensive survey of crowd density estimation and counting." *IET Image Processing* 19.1 (2025): e13328.
- Wang, S., Pu, Z., Li, Q., & Wang, Y. (2022). Estimating crowd density with edge intelligence based on lightweight convolutional neural networks. *Expert Systems with Applications*, *206*, 117823.
- Zhang, Y., Zong, R., Shang, L., Kou, Z., Zeng, H., & Wang, D. (2022). Crowdoptim: A crowd-driven neural network hyperparameter optimization approach to ai-based smart urban sensing. *Proceedings of the ACM on Human-Computer Interaction*, *6*(CSCW2), 1-27.
- Zhu, Y., Ni, K., Li, X., Zaman, A., Liu, X., & Bai, Y. (2024). Artificial intelligence aided crowd analytics in rail transit station. *Transportation research record*, *2678*(2), 481-492.

APPENDIXES

```
import streamlit as st
import torch
import cv2
import numpy as np
import time
from PIL import Image
# import your model
from model import CSRNet
# ---------------------------
# CONFIG
# ---------------------------
device = "cuda" if torch.cuda.is_available() else "cpu"

@st.cache_resource
def load_model():
    model = CSRNet().to(device)
    model.load_state_dict(torch.load("best_model.pth", map_location=device))
    model.eval()
    return model
model = load_model()
# ---------------------------
# PREPROCESS
# ---------------------------
def preprocess(frame):
    import cv2
    import torch
    import numpy as np
    # grayscale → RGB
    if len(frame.shape) == 2:
        frame = cv2.cvtColor(frame, cv2.COLOR_GRAY2RGB)
```

```
    # RGBA → RGB
    elif frame.shape[-1] == 4:
        frame = cv2.cvtColor(frame, cv2.COLOR_BGRA2BGR)
    # BGR → RGB
    else:
        frame = cv2.cvtColor(frame, cv2.COLOR_BGR2RGB)
    frame = cv2.resize(frame, (512, 512))
    frame = frame.transpose(2, 0, 1)
    frame = torch.tensor(frame / 255.0, dtype=torch.float32)
    return frame.unsqueeze(0)
# ---------------------------
# PREDICT
# ---------------------------
def predict(frame):
    img = preprocess(frame).to(device)
    with torch.no_grad():
        pred = model(img)
    density = pred.squeeze().cpu().numpy()
    density = np.maximum(density, 0)  # 🔥 remove negatives
    count = density.sum()
    return density, count
# ---------------------------
# HEATMAP
# ---------------------------
def create_heatmap(frame, density):
    h, w = frame.shape[:2]  # get frame dimensions explicitly
    density_norm = density / (density.max() + 1e-6)
    heatmap = cv2.applyColorMap((density_norm * 255).astype(np.uint8), cv2.COLORMAP_JET)
    heatmap = cv2.resize(heatmap, (w, h))  # ✅ cv2.resize expects (width, height)
    # Ensure frame is BGR for addWeighted (convert from RGB if needed)
    frame_bgr = cv2.cvtColor(frame, cv2.COLOR_RGB2BGR)

    overlay = cv2.addWeighted(frame_bgr, 0.6, heatmap, 0.4, 0)
```

```
    # Convert back to RGB for Streamlit display
    return cv2.cvtColor(overlay, cv2.COLOR_BGR2RGB)
# --------------------------
# UI
# --------------------------
st.title("🚨 AI Crowd Density Monitoring System")
threshold = st.slider("Set Crowd Threshold", 10, 500, 100)
mode = st.radio("Select Mode", ["Upload Image", "Live Camera"])
# --------------------------
# IMAGE UPLOAD MODE
# --------------------------
if mode == "Upload Image":
    file = st.file_uploader("Upload Image", type=["jpg", "png"])

    if file:
        image = Image.open(file)
        frame = np.array(image)
        density, count = predict(frame)
        overlay = create_heatmap(frame, density)
        st.image(overlay, caption=f"Count: {int(count)}")
        if count > threshold:
            st.error(f"⚠️ ALERT! Crowd exceeds threshold ({int(count)})")
        else:
            st.success("✅ Safe Crowd Level")
# --------------------------
# LIVE CAMERA MODE
# --------------------------
if mode == "Live Camera":
    run = st.checkbox("Start Camera")
    FRAME_WINDOW = st.image([])
    cap = cv2.VideoCapture(0)
    while run:
        ret, frame = cap.read()
```

```
    if not ret:
        st.warning("Camera not accessible")
        break
    frame = cv2.cvtColor(frame, cv2.COLOR_BGR2RGB)
    # every 5 seconds inference
    density, count = predict(frame)
    overlay = create_heatmap(frame, density)
    FRAME_WINDOW.image(overlay)
    st.write(f"👥 Count: {int(count)}")
    if count > threshold:
        st.error(f"🚨 ALERT! Threshold exceeded ({int(count)})")
    else:
        st.success("Safe")
    time.sleep(5)
cap.release()
```