Human silhouette detection in images using machine learning

Thesis submitted in partial fulfillment of the requirements for the award of degree of

Master of Engineering
in
Computer Science and Engineering

Submitted By
Akbar Ali Ahamed
(Roll No. 802332007)

Under the supervision of:

Dr. H.S Pannu
(Assistant Professor)
Dr. Sanjeev Rao
(Assistant Professor)

COMPUTER SCIENCE AND ENGINEERING DEPARTMENT
THAPAR INSTITUTE OF ENGINEERING AND TECHNOLOGY
PATIALA – 147004

June 2025

CERTIFICATE

I hereby certify that the work which is being presented in the thesis entitled, "Human Silhouette detection in images using Machine Learning", in partial fulfillment of the requirements for the award of degree of Master of Engineering in *Software Engineering/ Computer Science and Engineering* submitted in Computer Science and Engineering Department of Thapar Institute of Engineering and Technology, Patiala, is an authentic record of my own work carried out under the supervision of *Dr. H.S Pannu and Dr. Sanjeev Rao* and refers other researcher's work which are duly listed in the reference section. The matter presented in the thesis has not been submitted for the award of any other degree of this or any other University.

(Akbar Ali Ahamed)

This is to certify that the above statement made by the candidate is correct and true to the best of my knowledge.

Dr. H.S Pannu
Assistant Professor

Dr. Sanjeev Rao
Assistant Professor

Acknowledgement

I express my sincere gratitude to all those who have supported and guided me throughout my thesis work. This research would not have been possible without the encouragement, insights, and assistance I received from several individuals and institutions. First and foremost, I am very grateful to my respected supervisors, Dr. H.S. Pannu and Dr. Sanjeev Rao, for their invaluable guidance, encouragement, and constant support. Their expertise, constructive feedback, and timely suggestions were instrumental in shaping the direction of this research and bringing it to its present form. I am truly fortunate to have had the opportunity to work under their mentorship. I also extend my heartfelt appreciation to the faculty members and staff of the Department of CSE, Thapar Institute of Engineering and Technology (TIET), for providing a supportive academic environment and access to essential resources and facilities. My sincere thanks go to my fellow researchers, colleagues, and friends who have contributed in various ways by offering assistance, motivation, and sharing their knowledge and experiences. I am especially grateful to my family for their unwavering encouragement, patience, and belief in my capabilities throughout this academic journey. Their support has been a constant source of strength. Lastly, I wish to acknowledge all those whose names may not have been mentioned here but who have, directly or indirectly, contributed to the successful completion of this thesis. Each contribution, no matter how small, has been truly appreciated. With deep respect and gratitude, I dedicate this work to all who have made this endeavor a fulfilling experience.

Abstract

Detecting human outlines, or silhouettes, has emerged as a crucial task in the field of machine learning, with important applications in areas such as surveillance, human–computer interaction, healthcare monitoring, and autonomous navigation. Accurate silhouette detection is essential not only for ensuring safety but also for improving accessibility and user experience in systems designed to assist individuals in real-world environments. Unlike traditional computer vision techniques that rely on hand-crafted rules, modern machine learning models—particularly convolutional neural networks (CNNs)—are capable of learning visual patterns from data, making them more effective in handling complex and cluttered scenes. This research introduces a practical machine learning framework for human silhouette detection, focusing on identifying individuals in natural environments where backgrounds may include occlusions from trees, rocks, and other visual distractions. A custom dataset was created for this purpose, containing annotated images that reflect diverse backgrounds, human postures, and varying levels of visibility. This dataset supports realistic model training and evaluation by simulating challenging real-world scenarios. The study employs and compares two deep learning architectures: YOLOv8n (You Only Look Once) and DETR (Detection Transformer). YOLOv8n is a lightweight, real-time object detection model optimized for high-speed performance, making it suitable for deployment in resource-constrained systems such as drones or embedded devices. In contrast, DETR applies transformer-based attention mechanisms to capture global context within an image, offering improved detection performance in scenes with overlapping or partially occluded human figures. By evaluating both models on the custom dataset, the research highlights their relative strengths in terms of accuracy, speed, and suitability for different deployment scenarios. Overall, the work outlines a structured approach to designing and evaluating human silhouette detection systems using state-of-the-art models and a dataset tailored to real-world conditions. The findings contribute to a better understanding of the trade-offs involved in deploying deep learning-based detection systems and provide a foundation for further development of reliable and adaptable computer vision solutions.

Table of Contents

Certificate **i**
Acknowledgment **ii**
Abstract **iii**
Table of Contents **iv**
List of Figures **v**
List of Tables **vi**
Notations and Symbols **vii**
1. Introduction **1**
1.1 Existing Challenges 2
1.2 Objectives 2
1.3 Roadmap of the Thesis 3
2. Literature Review **4**
3. Research Gaps **11**
4. Methodology **12**
4.1 Fundamentals of Deep Learning and CNNs 12
4.2 CNN-Based Object Detection 16
4.3 YOLOv8n Architecture 16
4.4 DETR R50 Transformer-Based Model 16
4.5 Dataset Preparation and Augmentation 18
4.6 Training Process and Optimization 18
4.7 Evaluation Metrics 19
5. Results **21**
6. Conclusion **30**

List of Figures

Figure 1: Convolution Neural Network (CNN) Architecture 24

Figure 2: Overview of the Vision Transformer architecture used in DETR 29

Figure 3: Showing the accuracy of YOLOv8n model trained on the custom dataset Humano. 33

Figure 4: Showing evaluation metrics for determining the DETR R-50 model's accuracy in detecting humans. 34

Figure 5: Showing the accuracy of the PDSC-YOLOv8n model. 34

Figure 6: Inference of YOLOv8n model on an image of a human holding a dog, with the person’s face not visible 36

Figure 7: Detection of a human in low-light conditions next to a group of mannequins. 36

Figure 8: Ground truth of inference via DETR-R50 object detection model 37

Figure 9: Actual detections by DETR-R50 with each person’s probability of being a human mention above their respective bounding boxes. 37

Figure 10: Another random image in which detections are made accurately across a crowded environment by the DETR-R50 model. 38

Figure 11: Detecting two humans next to a mannequin, with confidence scores of 0.65 and 0.67. 38

Figure 12: Performance graph for mean Average precision values against their accuracy score. 39

Figure 13: This figure represents the training and validation losses over the whole training of the model. 40

Figure 14: Human detection results using the PDSC-YOLOv8n model, accurately identifying three individuals with confidence scores of 0.49, 0.59, and 0.80. 41

Figure 15: Human detection by the PDSC-YOLOv8n model, accurately identifying one individual with a confidence score of 0.79 in a grayscale urban scene. 41

List of Tables

Table 1: Summary of existing approaches used to detect objects in various scenarios. 20

Table 2: Comparative analysis of object detection models (YOLOv8n, DETR-ResNet50, and PDSC-YOLOv8n) evaluated on the Humano custom dataset. 35

Notations and Symbols

Symbols	Meaning
$X \in R^{H \times W \times C}$	Input image matrix where H = height, W = width, and C = number of channels
$F \in R^{k \times k \times C}$	Convolutional filter/kernel of size k×kk \times kk×k, applied over C channels
$Z_{i,j}^{(l)}$	Output at position (i,j)(i, j)(i,j) in the feature map at layer lll, before activation
$b(l)$	Bias term at the layer l
θ	Model parameters (weights and biases)
η	Learning rate used by the optimizer
m_t	First moment estimate in Adam optimizer at time step t
v_t	Second moment estimate in Adam optimizer at time step t
N	Total number of object classes or prediction slots
x_i	Logit value or feature input for the class index i
l	Layer index in a neural network
i, j	Spatial indices (row and column positions)
k	Kernel size (e.g., 3 for a 3×3 filter)
H	Image height
W	Image width
C	Number of channels (e.g., RGB → 3)
∑	Summation operator used in convolution and loss computations
$ReLU(x)$	Rectified Linear Unit activation function, outputs max(0,x)\max(0, x)max(0,x)

Chapter 1: Introduction

Human detection using machine learning (ML) has emerged as a critical area of research and application due to its transformative potential in enhancing safety, security, and operational efficiency. The ability to accurately detect and identify humans in diverse environments has implications across numerous domains, including search and rescue (SAR) operations, disaster management, and autonomous systems. In SAR scenarios, human detection plays a pivotal role in saving lives by enabling fast identification of missing or injured individuals in challenging environments. These environments often include vast, inaccessible terrains such as forests, mountains, and disaster-affected areas. Traditional methods of search and rescue, which rely heavily on manual efforts, are time-intensive and resource-heavy. The longer it takes to locate a person, the more their survival chances diminish. Machine learning, integrated with advanced tools like drones and high-resolution cameras, revolutionizes this process by drastically reducing the time needed to scan and analyze large areas. This rapid response capability can mean the difference between life and death in critical situations. One of the key advantages of ML-based human detection is its ability to overcome limitations associated with human operators. Identifying a person in complex scenes, especially when they may be hidden by vegetation, lying in unusual positions due to injuries, or blending into the environment, is a challenging task even for trained professionals. These difficulties are intensified during adverse weather conditions, such as fog, rain, or snow, where visibility is significantly reduced. ML models, trained on diverse datasets, excel in these situations by automating detection with high levels of precision. For example, deep learning frameworks like YOLOv4 can identify humans in drone-captured images even when the individuals occupy only a small portion of the frame or are partially obscured. Another important aspect of ML-driven human detection is its adaptability to a variety of scenarios. Unlike traditional detection methods, ML systems can be fine-tuned using domain-specific datasets. These datasets simulate real-world challenges, including motion blur, poor lighting, and extreme weather conditions, ensuring that the models remain robust and reliable across diverse environments. This adaptability extends the usefulness of these systems to applications beyond SAR, such as monitoring crowded public spaces, enhancing security systems, and aiding autonomous vehicles in recognizing pedestrians. Furthermore, ML systems address the issue of resource optimization by reducing false positive detections. In operations where resources like time, personnel, and equipment are limited, false detections can lead to wasted effort and delays in critical responses. By employing advanced evaluation metrics and refining detection algorithms, ML models can significantly minimize false alarms, ensuring that resources are directed only to genuine cases. This level of accuracy and efficiency makes ML-based human detection an indispensable tool in resource-constrained operations. The purpose of this research is to compare various machine learning models' performance on a dataset containing human silhouette images, to identify people individually or as a group in any given image.

1.1 Existing Challenges

Detecting small objects in cluttered environments remains difficult. As there is disparity in object classes and the model is unable to maintain accuracy in places where there are, for example, large crowds, images in the dataset are of varying scale [3]. Models struggle to capture distant features effectively. The challenge of capturing distant features of humans effectively in machine learning models often arises from the tradeoff between detection speed and accuracy. An example of this is [2]. Nighttime and poor visibility conditions hinder detection accuracy. Poor visibility conditions hinder detection accuracy, as machine learning models struggle to extract meaningful features in low-light environments. Occlusion further complicates the task, as shadows and overlapping objects reduce the distinguishability of human shapes, making it challenging to detect individuals accurately [6].

1.2 Objectives

Developing a robust and efficient human detection system that addresses challenges of occlusion, pose variability, low-light conditions, background noise, and scale variations while achieving a balance between real-time performance and accuracy. The framework should be capable of identifying human silhouettes across diverse scenarios, including natural scenes with cluttered backgrounds, low-visibility conditions, and variable human postures, making it applicable for surveillance, healthcare, and autonomous navigation systems. The goals of this thesis are to: conduct a comprehensive review of literature in the field of human detection using machine learning, with a focus on silhouette-level detection challenges; evaluate and compare advanced deep learning models—specifically YOLOv8n and Detection Transformer (DETR)—to determine their suitability for accurate and efficient silhouette recognition. Build a custom dataset simulating real-world complexity, including occlusions and varied lighting, and evaluate model performance in terms of detection accuracy, processing speed, and model efficiency. The objective is to assess how each model handles scale variance, motion blur, and energy constraints under different conditions. By analyzing the trade-offs between accuracy and speed, this work seeks to recommend optimal approaches for deploying silhouette detection models in real-time applications, while maintaining computational efficiency and adaptability.

1.3 Roadmap of the Thesis

1. Introduction - The research topic is discussed in brief, along with the significance and potential benefits of pursuing this research. Importance of the problem, its applications, and what challenges are there in the present work.

2. Literature review - This chapter reviews state-of-the-art models, techniques, and discusses their advances. Specifically, CNNs and vision transformers are reviewed with their potential uses as well.

3. Research Gaps - This chapter mentions the limitations of the current techniques in detecting human silhouettes with speed and high accuracy. The factors affecting the models' performance are discussed in detail.

4. Methodology - It focuses on the workings of the state-of-the-art models and how they are trained, process data, etc. This section gives a clear view of what is working and how to improve the models further by exploring their inner workings in as much detail as possible.

5. Problem statement and objectives - Defining the problem to be solved and what goals are meant to be achieved by this research.

6. Results and evaluation - In this section, results are shown, and evaluation of the model's accuracy is done based on the unique metrics calculated after training.

7. Conclusion and future scope - Inferring which model suits the purpose of fast and reliable detection of humans in various environments and visual conditions. The future scope discusses possible adaptations to the current works to improve real-time human detection.

Chapter 2: Literature Review

Object detection has emerged as a core task in computer vision, enabling numerous real-world applications ranging from security surveillance and autonomous driving to healthcare monitoring and aerial reconnaissance. Within this broad area, human silhouette detection holds specific importance for tasks that involve identifying individuals in dynamic, cluttered, or challenging environments. These include search and rescue operations, smart healthcare systems, fall detection, pedestrian tracking, and safety automation. Detecting human silhouettes is inherently complex due to issues such as background clutter, varying lighting conditions, occlusion, pose variability, motion blur, scale inconsistencies, and the requirement for energy-efficient, real-time performance. Over the years, advances in deep learning—particularly convolutional neural networks (CNNs) and transformer-based architectures—have significantly improved object detection capabilities. However, balancing speed, accuracy, and robustness in diverse conditions remains an active research challenge. This literature review examines a selection of significant papers published between 2020 and 2025 that collectively represent the evolution of methodologies, datasets, architectures, and deployment techniques for human silhouette and related object detection problems.

The year 2020 marked foundational contributions focusing on system limitations and metric evaluations. A comprehensive review of deep learning-based object detection models outlined the fundamental challenges faced by traditional object detectors when applied to remote sensing, surveillance, and silhouette-related tasks, particularly in detecting small, distant, or partially occluded objects [23]. It emphasized the need for more adaptive, hybrid learning approaches to address context complexity and varying image conditions. Evaluation frameworks were another area of focus. A study comparing different performance metrics for object detection highlighted that widely used indicators like mean Average Precision (mAP) and Intersection over Union (IoU) do not always reflect detector reliability in edge-case scenarios, such as when objects are occluded or appear in dense clusters [19]. This issue is particularly relevant for human silhouette detection, which often occurs in crowded urban scenes or obstructed environments. Intrusion detection systems also received attention during this period. Research into machine learning-driven intrusion detection identified the need for more flexible classifiers capable of maintaining accuracy under fluctuating image noise and lighting conditions, underscoring the general demand for scalable detection systems [7]. In 2021, the literature progressed toward application-specific studies. One such work concentrated on person detection for search and rescue operations in mountainous and forested regions. The study evaluated popular object detectors like YOLOv4 and Faster R-CNN and found that while these models performed well under controlled settings, their detection reliability significantly decreased in natural environments, particularly when people were camouflaged or partially visible [1]. These findings illustrated that general-purpose object detectors require specific adjustments to handle natural occlusions and background blending effectively. Another important contribution in the same year was the development of the DOTA dataset, which provided a large-scale benchmark for object

detection in aerial imagery. This dataset introduced a variety of object scales, densities, and rotations, making it a useful resource for training models aimed at human silhouette detection from aerial views [21]. In addition, research into improving geometric modeling techniques for object localization proposed the use of advanced anchor strategies and bounding box refinement to better detect overlapping or ambiguously shaped objects, which is particularly useful in scenarios involving multiple or interacting human figures [20]. By 2022, model efficiency and real-time suitability became focal points in object detection research. A notable contribution was the design of L-DETR, a lightweight version of the Detection Transformer model. While the original DETR offered promising performance in modeling object relationships through global attention, it was computationally intensive. L-DETR reduced inference time and memory usage through simplified encoder-decoder mechanisms, making it more feasible for real-time silhouette detection on drones and embedded systems [14]. This shift toward edge-compatible models set the stage for the following years' developments in deploying accurate, yet computationally lightweight, detection systems. In 2023, several impactful developments emerged across different domains. A study proposing a reproducible object detector built entirely on public datasets underscored the importance of model transparency and domain transferability. Its modular design allowed easy customization for specialized applications like silhouette detection, where domain-specific tuning is often essential [3]. Another comprehensive review compared CNN-based models and vision transformers in object detection tasks. CNNs were found to excel in localized spatial feature extraction, while transformers offered superior global context modeling. The study recommended hybrid approaches combining the strengths of both models for improved object detection in cluttered scenes [15]. In line with this direction, further research introduced an enhanced version of YOLOv8 called LAR-YOLOv8, designed specifically for small object detection in remote sensing. This model integrated a dual-branch attention mechanism and vision transformer modules to extract multi-scale contextual features. Additionally, it introduced a modified regression loss to improve localization accuracy, addressing long-standing issues in silhouette boundary precision [22]. A significant retrospective study reviewed two decades of object detection development, from handcrafted features like HOG and Viola-Jones to deep learning-based detectors. The review discussed emerging priorities such as neural architecture search (NAS), hardware-aware design, and the need for adaptable models for context-dependent deployment [18]. In 2024, model specialization for domain-specific tasks became more prominent. A modified YOLOv8 model was introduced to detect moving objects with improved accuracy and speed. Structural changes to the backbone and feature fusion layers enhanced motion sensitivity and helped the model outperform standard YOLOv8 in dynamic scenes such as pedestrian tracking and crowd surveillance [2]. Another study used YOLOv8 to develop an object detector suitable for low-light underwater conditions. Though designed for marine life detection, the improvements in low-light visibility and noise reduction translated well to general silhouette detection in dimly lit environments [8]. In the context of UAV imagery, researchers proposed LE-YOLOv8n, a lightweight detector incorporating LHGNet and a restructured feature extraction path. The model was optimized to

handle tiny objects in wide-area views while maintaining a high frame rate, making it suitable for real-time human detection from drones [9]. A review of human detection in UAV-based search and rescue operations identified critical gaps in the current models' ability to handle scale variance and environmental irregularities. It called for better-curated datasets and training strategies tailored to aerial silhouette detection [5]. In 2025, the trend of refining YOLO-based models for specific applications continued. A model developed for agricultural field monitoring adapted YOLOv8 for weed detection, using a streamlined backbone and attention mechanisms to identify low-contrast targets in natural surroundings. Though agriculture-focused, the underlying modifications applied to human silhouette detection in vegetated areas [10]. Another approach designed for small-object detection introduced detection head optimizations and pruning strategies, improving performance without increasing latency [11]. A transformer-augmented YOLOv8 variant, MIS-YOLOv8, targeted small object recognition in UAV footage by leveraging multi-scale integration and cross-channel attention. It demonstrated improved results in detecting small, partially visible targets such as silhouettes in cluttered aerial frames [12]. In surveillance applications, an enhanced YOLOv8n model using MSBlock and AKConv achieved a better balance between detection speed and accuracy, suitable for real-time deployment on edge devices [13]. A large-scale review explored the evolution of the YOLO family and its integration into autonomous driving systems, discussing unresolved issues such as night-time performance, detection under motion blur, and multi-object tracking in congested environments—all relevant to silhouette detection [16]. Finally, metaheuristic optimization techniques were introduced to enhance YOLOv8 training efficiency and detection generalization across unseen domains. The use of hybrid optimization methods improved convergence rates and reduced overfitting, allowing better adaptation to real-world detection scenarios [17].

Collectively, these studies illustrate the clear trajectory of object detection research over the past five years. The field has progressed from monolithic, general-purpose models to modular, highly optimized architectures tailored to specific deployment needs. CNN-based models like YOLO have remained dominant in real-time applications due to their low computational overhead and high accuracy, while transformer-based models have demonstrated improved capacity for scene understanding and complex feature interactions. The growing use of hybrid models that combine CNN and transformer elements reflects an effort to balance spatial precision with contextual awareness. Datasets have also evolved, from simple benchmarks to domain-specific, high-resolution, annotated image collections designed to simulate real-world variability in pose, lighting, and background. Furthermore, research attention has expanded to include not just model architecture, but also optimization strategies, training data curation, and deployment scenarios. Despite these advances, human silhouette detection remains a challenging problem in computer vision. Real-world deployment environments, such as urban surveillance, forested search areas, or disaster zones, often involve factors that compromise detection reliability, such as overlapping objects, partial occlusions, or rapidly changing light conditions. While recent models have made progress in speed, interpretability, and robustness, issues like small silhouette detection, generalization across domains, and performance on embedded hardware still require deeper

exploration. In particular, efforts must focus on improving detection stability under motion, ensuring silhouette localization under occlusion, and developing hybrid architectures capable of adapting to domain shifts. Additionally, the creation of rich, diverse, and high-resolution silhouette-specific datasets is crucial for supporting better training, benchmarking, and cross-model comparisons. As detection models continue to mature, future work will likely center on optimizing architectures for generalizability, incorporating self-supervised learning, and integrating multi-modal inputs such as depth and infrared further to enhance silhouette detection performance in unconstrained, real-world conditions.

Table 1: Summary of existing approaches used to detect objects in various scenarios.

Sno	References	Dataset used	ML/DL Approach	Research Gaps
1	Aziz et al. (2020) [23]	Various benchmarks	General CNN-based methods	Persistent issues with small object detection, occlusion, and adaptation in remote sensing.
2	Padilla et al. (2020) [19]	COCO, VOC	Evaluation (non-model-specific)	mAP and IoU fail to reflect detector performance in occluded or dense scenes.
3	Musa et al. (2020) [7]	Custom intrusion detection dataset	ML classifiers (SVM, Decision Trees)	Poor generalization in noisy environments; lack of flexible classification strategies.
4	Sambolek & Ivasic-Kos (2021) [1]	Search and rescue (SAR) dataset	YOLOv4, Faster R-CNN	Decreased performance in natural terrains under occlusion and camouflage.
5	Ding et al. (2021) [21]	DOTA	Dataset/Benchmark	Existing detectors underperform with rotated, occluded, and small-scale aerial objects.
6	Zheng et al. (2021) [20]	VOC, COCO	Geometry-aware object detection (Faster R-CNN-like)	Weak bounding box precision under clutter and occlusion.
7	Li et al. (2022) [14]	COCO	L-DETR	DETR required high computation; L-DETR still lacks maturity for fast edge deployment.
8	Ren et al. (2023) [3]	Public open datasets	Modular CNN-based detector	Model consistency across domains remains limited.

9	Amjoud & Amrouch (2023) [15]	VOC, COCO	CNNs, Vision Transformers	Need for hybrid models to leverage both local and global feature extraction.
10	Yi et al. (2023) [22]	RSOD, NWPU, CARPK	LAR-YOLOv8	Small object detection and dense clustering remain unresolved.
11	Zou et al. (2023) [18]	Multiple datasets (20-year survey)	HOG, Viola-Jones, YOLO, Faster R-CNN	Lightweight, real-time models with context awareness are still underdeveloped.
12	Safaldin et al. (2024) [2]	MOT Challenge, custom dynamic scene datasets	Improved YOLOv8	YOLO's motion sensitivity was limited; improvements are still domain-dependent.
13	Ding et al. (2024) [8]	RUOD, VOC2012	PDSC-YOLOv8n	Model performance under water translated poorly to complex land-based visuals.
14	Yue et al. (2024) [9]	VisDrone2019	LE-YOLOv8n	High-altitude, fast-motion detection still causes instability.
15	Abdelnabi & Rabadi (2024) [5]	UAV-based SAR datasets	CNN-based models (YOLO, SSD)	Insufficient aerial human datasets; weak performance at multiple scales.
16	Wang et al. (2025) [10]	Custom agricultural field dataset	Optimized YOLOv8	The adaptability of agricultural models to human silhouettes is unproven.

17	Hao & Li (2025) [11]	Remote sensing object datasets	YOLOv8 variant	Difficulty in detecting small and fast-moving targets.
18	Tao et al. (2025) [12]	UAV imagery	MIS-YOLOv8	Angle distortion and complex aerial backgrounds challenge accuracy.
19	Wang et al. (2025) [13]	Custom surveillance dataset	YOLOv8n + MSBlock + AKConv	Needs testing under low light and high object density.
20	Wei et al. (2025) [16]	KITTI, nuScenes, COCO	YOLOv1–YOLOv12	Night-time detection, motion blur, and tracking remain difficult.
21	Elgamily et al. (2025) [17]	Remote sensing datasets	YOLOv8 + metaheuristic optimizers	Generalization in unseen domains remains limited.

Chapter 3: Research Gaps

a. **Occlusion and Partial Visibility**: Most current models struggle to accurately detect human silhouettes when individuals are partially hidden by objects or other people, especially in cluttered or natural environments.

b. **Scale Variation and Small Object Detection**: Detection performance drops significantly when human figures appear at varying scales, particularly in aerial or distant surveillance imagery where silhouettes are small.

c. **Illumination and Low-Light Conditions**: Many models show degraded performance under poor lighting, night-time conditions, or environments with glare and shadows, limiting real-world usability.

d. **Motion Blur and Dynamic Scenes**: In fast-moving or drone-based footage, motion blur and scene shifts reduce detection accuracy, indicating a need for models with improved temporal robustness.

e. **Generalization Across Domains**: Despite architectural improvements, several models lack adaptability when applied to unseen environments or datasets, highlighting weak cross-domain generalization.

f. **Lack of Silhouette-Specific Datasets**: There is a notable absence of large, annotated datasets focused specifically on human silhouettes across diverse conditions such as pose variation, crowd density, and occlusion levels.

g. **Computational Efficiency on Edge Devices**: Many transformer-based and deep CNN models require high computational power, making real-time deployment on embedded or mobile platforms challenging.

h. **Integration of Contextual and Global Features**: CNN-based detectors excel in local feature detection but lack global context modeling, while transformer models demand more resources. Effective integration of both remains underexplored.

i. **Robustness in Cluttered Backgrounds**: Detectors often misclassify background objects or miss human outlines in complex environments like forests, urban scenes, or disaster zones.

j. **Limited Use of Hybrid Architectures**: While some studies explore combining CNNs with attention mechanisms or transformers, fully optimized hybrid models tailored to silhouette detection are still limited in number.

Chapter 4: Methodology

4.1 Fundamentals of Deep Learning and CNNs

Deep learning is a subset of machine learning that focuses on models composed of multiple computational layers, often called artificial neural networks. These models are capable of automatically learning high-level representations from large volumes of data, eliminating the need for manual feature engineering. The depth and complexity of these models allow them to capture intricate structures and patterns that traditional algorithms may fail to detect, particularly in tasks involving unstructured data such as images, audio, or natural language.

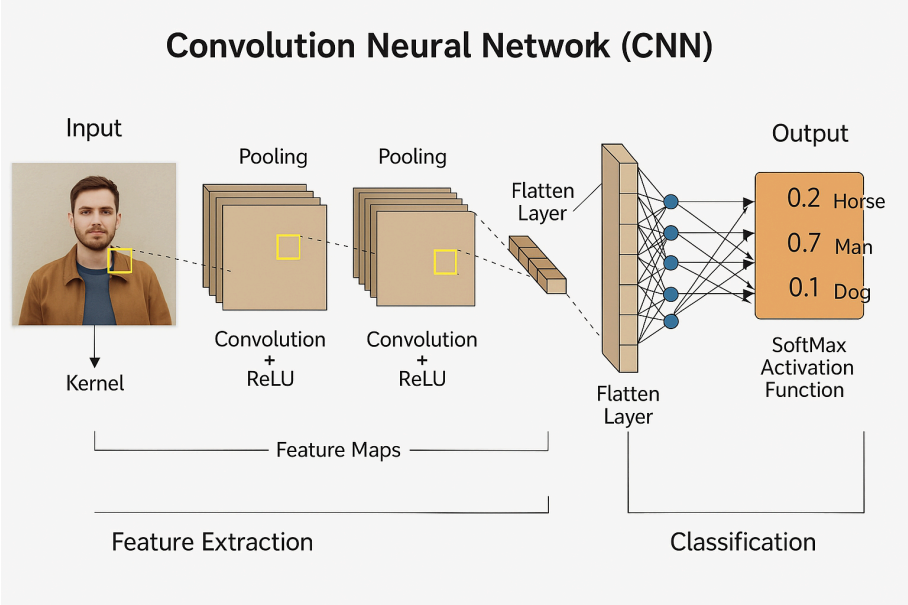

Figure 1: Convolution Neural Network (CNN) Architecture

In the realm of computer vision, deep learning has brought transformative advancements, primarily driven by the capabilities of Convolutional Neural Networks (CNNs). As illustrated in figure 1, CNNs are specialized neural architectures designed to process and analyze visual data by leveraging the spatial and hierarchical structure inherent in images. Unlike traditional fully connected neural networks, which treat input data as one-dimensional vectors, CNNs maintain the two-dimensional grid-like layout of images. This preservation of spatial relationships between pixels enables CNNs to detect meaningful patterns such as edges, textures, and shapes through localized receptive fields. The architecture shown in figure 1 demonstrates the typical workflow of a CNN. The input image undergoes convolutional operations using learned filters (kernels), followed by non-linear activation functions like ReLU. These layers extract increasingly complex features across the network's depth. Pooling layers reduce spatial dimensions while retaining essential structural information, thereby improving computational

efficiency and controlling overfitting. The resulting feature maps are then passed through a flattening layer, transforming the multi-dimensional tensors into a one-dimensional vector suitable for classification. This flattened output is processed through fully connected (dense) layers that act as a decision-making unit, culminating in the final class probabilities. As shown in the output block of Figure 1, the SoftMax activation function is applied to convert raw output scores into normalized probabilities across multiple categories. This enables the network to make confident predictions, such as recognizing the image as a "Man" with a 70% confidence score. Through this pipeline, CNNs can learn both low-level features (e.g., edges, corners) and high-level concepts (e.g., faces, objects), making them highly effective for complex image understanding tasks. Their layered structure also supports transfer learning, where pre-trained models can be fine-tuned for specific applications, significantly reducing the training time and dataset size requirements.

Key Components of CNNs

Convolutional Layers: These layers apply learnable filters (kernels) to the input image or feature maps. A filter slides over the input and computes dot products, resulting in a feature map that emphasizes specific patterns such as edges, corners, textures, or color gradients. Each filter is tuned during training to detect a different visual feature, allowing the network to develop an internal representation of the input space.

Mathematically, the convolution operation for a 2D input X and kernel K is given by:

$$(X * K)(i,j) = \sum_m \sum_n X(i + m, j + n) \cdot K(m,n)$$

a. Activation Functions: After each convolution, a non-linear activation function such as ReLU (Rectified Linear Unit) is applied to introduce non-linearity into the model. This enables the network to approximate complex functions and learn more intricate patterns.

$$ReLU(x) = max(0, x)$$

b. Pooling Layers: Pooling operations reduce the spatial dimensions (width and height) of the feature maps, making computation more efficient and providing a degree of spatial invariance. The most common type is max pooling, which retains the maximum value in each sub-region, preserving the most salient features.

c. Fully Connected Layers: These layers interpret the high-level features learned by the convolutional and pooling layers and map them to the final output, such as class probabilities. Every neuron in a fully connected layer is connected to all neurons in the previous layer, making them powerful but computationally intensive.

d. Normalization and Regularization: Techniques like Batch Normalization help stabilize and accelerate training by normalizing layer inputs. Dropout is a regularization method used to prevent overfitting by randomly deactivating neurons during training.

Hierarchical Feature Learning: One of the key strengths of CNNs is their ability to learn hierarchical features. Early layers detect low-level features such as edges and color gradients. As the data flows through deeper layers, the network captures more abstract concepts—shapes, textures, and eventually full object representations. This progressive abstraction mimics the human visual cortex and allows CNNs to generalize across diverse visual scenarios.

For example:

- Layer 1: Detects edges, lines and simple textures
- Layer 2–3: Detects corners, object parts (e.g., wheels, eyes)
- Layer 4+: Detects entire objects (e.g., faces, cars)

Why CNNs are effective for object detection: CNNs are inherently translationally invariant due to their shared-weight structure and local receptive fields. This means they can recognize objects regardless of where they appear in an image. This is particularly important in object detection tasks, where the model must identify and localize multiple instances of various objects across diverse spatial positions and scales. Additionally, CNNs can process high-resolution images by stacking layers and using techniques like stride and dilation to capture large receptive fields, making them capable of detecting small or occluded objects in cluttered scenes, common challenges in real-world applications such as surveillance or autonomous driving.

Historical context and advancements: CNNs gained prominence after the success of AlexNet in the 2012 ImageNet Large Scale Visual Recognition Challenge (ILSVRC), where it outperformed traditional methods by a wide margin. Since then, architectures such as VGGNet, ResNet, Inception, and EfficientNet have introduced deeper networks, residual connections, and efficient computation strategies that enable training very deep models on large datasets. In object detection, Convolutional Neural Networks (CNNs) act as the foundational backbone for leading models such as YOLO, Faster R-CNN, and SSD. Typically pre-trained on large-scale classification datasets like ImageNet and fine-tuned for detection through transfer learning, CNNs excel in extracting spatial hierarchies of features. However, they face key limitations, including a dependency on large labeled datasets, high computational costs, and difficulties in capturing long-range dependencies and complex spatial relationships, particularly in cluttered scenes or those involving multiple interacting objects. To overcome these challenges, alternative architectures such as Transformers have emerged, with models like DETR integrating global self-attention mechanisms and convolutional backbones to enable more context-aware and semantically rich representations, thereby improving performance in scenarios where CNNs fall short.

4.2 CNN-Based Object Detection

Object detection involves two primary goals: classifying the objects within an image and determining their locations via bounding boxes. Modern deep learning-based detectors can be broadly categorized as:

Two-stage detectors: e.g., Faster R-CNN, which first generates object proposals and then refines and classifies them.
One-stage detectors: e.g., YOLO and SSD, which simultaneously predict bounding boxes and class scores across the image in a single pass.
One-stage detectors like YOLO have become increasingly popular due to their balance between accuracy and computational efficiency. They are well-suited for real-time applications such as street surveillance and autonomous vehicles, where inference speed is critical [13].

4.3 YOLOv8n Architecture

The YOLO (You Only Look Once) approach is described as reframing object detection as a regression task, enabling the model to process an entire image in a single forward pass by dividing it into a grid, with each cell predicting object bounding boxes and class probabilities. The YOLOv8n model, a lightweight and fast variant, incorporates several significant enhancements: Ghost Convolution and GSConv modules are used instead of standard convolutions to reduce parameter count; DCNv3 (Deformable Convolutions) improves the network's ability to manage geometric variations in object shape and pose; GAM (Global Attention Mechanism) and CBAM (Convolutional Block Attention Module) direct the model's focus toward spatially relevant features; and the WIoUv3 loss function improves bounding box regression accuracy. These combined innovations make YOLOv8n particularly well-suited for low-resource environments, including drones used in Search and Rescue (SAR) missions and embedded surveillance systems [15].

4.4 DETR-R50 Transformer-Based Model

DETR (Detection Transformer) introduces a novel approach to object detection by integrating transformer-based architectures originally developed for natural language processing into vision tasks. Unlike traditional CNN-based detectors that rely on anchor boxes and region proposal mechanisms, DETR reformulates object detection as a set prediction problem, treating the task of identifying and localizing objects as a direct output of learned sequence-to-sequence mappings. This design completely removes the need for handcrafted components such as anchor boxes and non-maximum suppression, replacing them with a transformer encoder-decoder framework powered by self-attention.

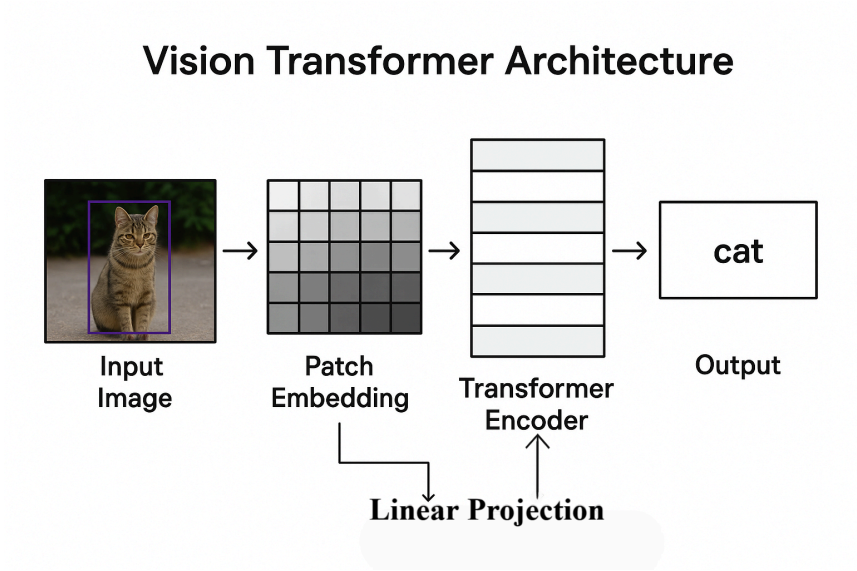

Figure 2: Overview of the Vision Transformer architecture used in DETR

As illustrated in figure 2, the input image is first divided into fixed-size patches, and each patch is linearly projected into a vector space to form patch embeddings. These embeddings, combined with positional encodings, are then passed through the Transformer Encoder. The encoder uses multiple layers of self-attention to capture global context by allowing each patch to attend to every other patch, regardless of spatial distance. This contrasts with the localized receptive fields in CNNs and enables better modeling of object relationships, even across distant regions in the image. The output of the encoder is processed by the decoder, which produces a fixed number of object predictions in parallel. Each prediction corresponds to either an object class and a bounding box or a "no object" class. The set-based prediction allows DETR to achieve strong spatial consistency and avoid duplicate detections. In this study, the DETR R50 variant was employed, which combines the transformer module with a ResNet-50 CNN backbone for initial feature extraction. ResNet processes the image into a feature map before passing it into the transformer encoder. DETR R50 was selected as a benchmarking model to evaluate against YOLOv8n, particularly in cluttered or semantically complex scenes. While DETR demonstrated robust performance, especially in terms of recognizing occluded or overlapping objects, it was notably slower to converge during training and required more computational resources due to the sequential nature of the transformer layers and the absence of region proposal shortcuts [14]. Ultimately, DETR's reliance on global attention allows it to understand the entire image context more holistically than CNNs, offering improved object detection performance in scenes with irregular layouts or dense object interactions, as visually represented in figure 2.

4.5 Dataset Preparation and Augmentation

Datasets were sourced and preprocessed using the Roboflow platform, which supports:

a. Annotation and format conversion (e.g., COCO to YOLO)
b. Image augmentation techniques such as flipping orientation, rotation, mosaic, brightness shift, etc.
c. Direct integration into Python notebooks via Roboflow's API

The datasets used primarily consisted of aerial and street-view images containing human figures, which aligns with the use cases of surveillance and search-and-rescue. Data was split into training (70%), validation (20%), and test (10%) sets. All images and bounding boxes were normalized to match the model input requirements.

4.6 Training Process and Optimization

Models were trained in the Google Colab environment using GPU acceleration. Training was conducted over 30 epochs with a batch size of 8–16, depending on memory availability.

Two primary optimizers were used:

a. Stochastic Gradient Descent (SGD): Widely used for YOLO models. It updates weights iteratively based on a subset (batch) of data and uses momentum to accelerate convergence.
$$\theta_{t+1} = \theta_t - \eta \cdot \nabla L(\theta_t)$$
b. Adam Optimizer: Used in transformer models like DETR. It combines the benefits of AdaGrad and RMSProp and adjusts learning rates adaptively for each parameter.
$$m_t = \beta_1 m_{t-1} + (1 - \beta_1)\nabla L(\theta_t), \qquad v_t = \beta_2 v_{t-1} + (1 - \beta_2)(\nabla L(\theta_t))^2$$

Learning rate scheduling was applied in both cases to improve training stability and final accuracy.

4.7 Evaluation Metrics

Evaluation was based on widely accepted object detection metrics:

Intersection over Union (IoU): A metric that quantifies the overlap between predicted and ground-truth bounding boxes.

$$IoU = \frac{Area\ of\ Overlap}{Area\ of\ Union}$$

IoU thresholds of 0.5 and above are considered acceptable for accurate predictions.

Precision, Recall, and F1-Score:

a. Precision: Measures how many predicted positives are actually correct
   $Precision = \frac{TP}{TP+FP}$
b. Recall: Measures how many actual positives are correctly detected.
   $Recall = \frac{TP}{TP+FN}$
c. F1-Score: Harmonic mean of precision and recall $F1 = 2.\frac{Precision.Recall}{Precision+Recall}$

Confusion Matrix:

a. TP (True Positives): Correct detections

b. FP (False Positives): Incorrect detections

c. FN (False Negatives): Missed objects
d. TN (True Negatives): Correct background classification (rare in object detection)

The confusion matrix provides a complete view of model predictions.

From this matrix, Accuracy can also be derived:

$$Accuracy = \frac{TP+TN}{TP+TN+FP+FN}$$

Mean Average Precision (mAP):

a. Average Precision (AP) is calculated for each class as the area under the precision-recall curve.
b. Mean Average Precision (mAP) is the mean of APs across all classes.

$$mAP = \frac{1}{N}\sum_{i=1}^{N} AP_i$$

In this project, we report:

- mAP@0.5: IoU threshold of 0.5
- mAP@0.5:0.95: Average across IoUs from 0.5 to 0.95 at 0.05 intervals

Chapter 5: Results

The performance evaluation of object detection models was conducted using a custom dataset named Humano, focusing on detecting human silhouettes. The models assessed include the baseline YOLOv8n, the DETR ResNet-50 architecture, and a modified PDSC-YOLOv8n variant. Each model was evaluated using standardized metrics such as mean Average Precision (mAP) at an Intersection over Union (IoU) threshold of 0.5 and the COCO evaluation protocol, ranging from IoU 0.5 to 0.95.

Figure 3 presents the inference outcome for the baseline YOLOv8n model trained on the Humano dataset. The evaluation was conducted on 80 validation images containing 116 human instances. The model achieved a mean Average Precision (mAP@0.5) of 61.05%, with a preprocessing time of 2.0 ms, inference time of 5.9 ms, and postprocessing time of 3.3 ms per image. These results highlight the model's ability to detect human silhouettes effectively with moderate accuracy while maintaining real-time speed suitable for deployment in low-latency environments.

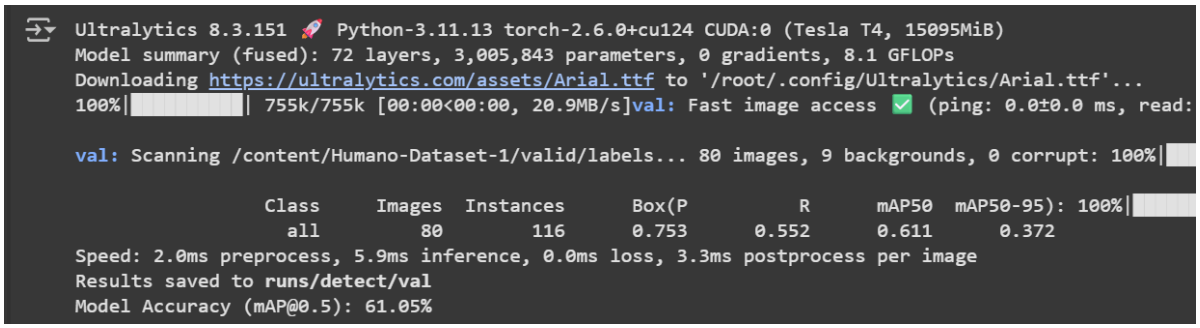

Figure 3: Showing the accuracy of YOLOv8n model trained on the custom dataset Humano.

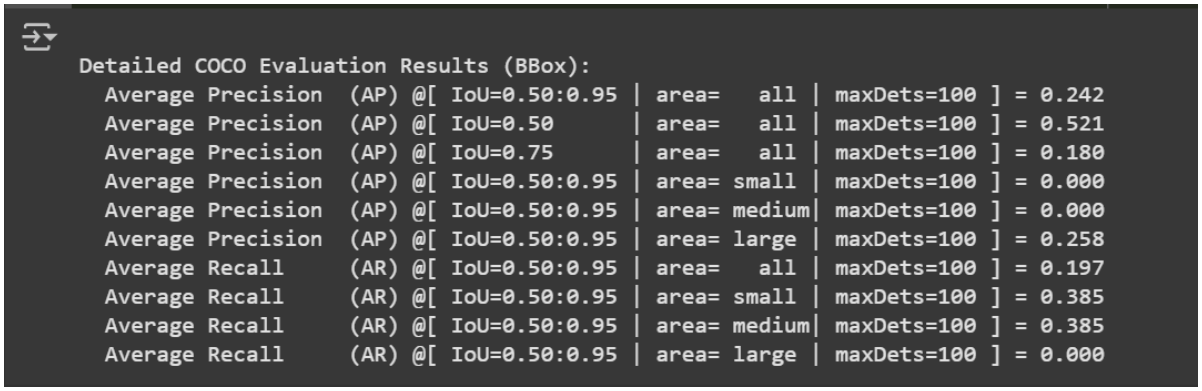

Figure 4: Showing evaluation metrics for determining the DETR R-50 model's accuracy in detecting humans.

In contrast, figure 4 demonstrates the COCO evaluation metrics for the DETR ResNet-50 model. This model attained an Average Precision (AP) at IoU=0.50 of 0.521, indicating decent performance when considering only a single IoU threshold. However, its overall AP across IoU thresholds (0.50:0.95) stood at 0.242, reflecting challenges in achieving high accuracy under stricter overlap conditions. Notably, the model recorded zero performance on small and large objects for AP and AR under some IoU conditions, suggesting difficulty in detecting scale-variant targets. The average recall (AR) values also remain below 0.4, showing limitations in sensitivity compared to the YOLO-based approaches.

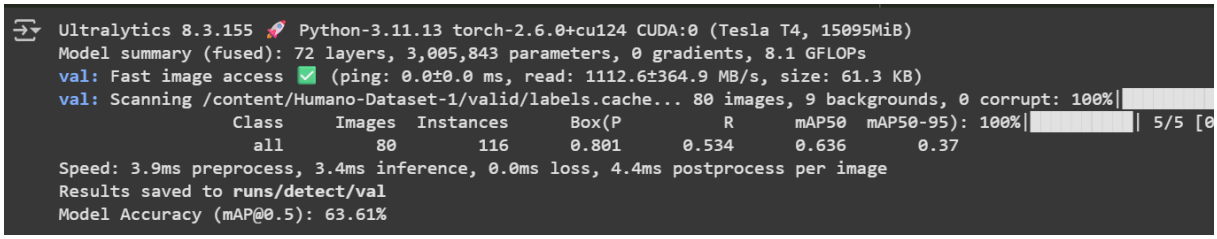

Figure 5: Showing the accuracy of the PDSC-YOLOv8n model.

Figure 5 illustrates the evaluation result of the PDSC-YOLOv8n model, a custom enhancement of the baseline YOLOv8n. This model achieved an improved mAP@0.5 of 63.61%, outperforming the standard YOLOv8n. Additionally, it showed slightly better performance in box precision (0.801) and maintained inference speeds conducive for practical applications. These improvements validate the effectiveness of architectural modifications implemented in the PDSC variant for the human detection task.

Table 2: Comparative analysis of object detection models (YOLOv8n, DETR-ResNet50, and PDSC-YOLOv8n) evaluated on the Humano custom dataset.

Sno.	Dataset	Model	Parameters	Inference time	Performance
1.	Humano custom Dataset	YOLOv8n	3.011M parameters	5.9ms	61.05% Accurate
2.	Humano custom Dataset	DETR-resnet 50	41.5 M parameters	50.42ms	52.1% Accurate
3.	Humano custom Dataset	PDSC-YOLOv8n	3.011M parameters	3.4ms	63.61% Accurate

Table 2 presents a comparative summary of three object detection models evaluated on the Humano custom dataset—YOLOv8n, DETR-ResNet50, and the proposed PDSC-YOLOv8n. Each model is analyzed based on its total parameter count, inference time, and detection accuracy. The DETR-ResNet50 model, as seen in figures 4 and 13, utilizes the highest number of parameters at 41.5 million and demonstrates the slowest inference time at 50.42 milliseconds. Despite its deep transformer-based architecture, it achieved a relatively modest accuracy of 52.1%, suggesting challenges in real-time deployment and fine-grained localization across varying scales. On the other hand, the lightweight YOLOv8n model (figure 3) operates with only 3.011 million parameters and delivers a significantly faster inference speed of 5.9 milliseconds, with a competitive accuracy of 61.05%. Most notably, the enhanced PDSC-YOLOv8n (figure 5) surpasses both models with a leading accuracy of 63.61%, while also offering the fastest inference at just 3.4 milliseconds, highlighting its suitability for real-time applications. The results emphasize that architectural optimization in lightweight models can outperform heavier models like DETR, especially when targeting specific tasks such as human silhouette detection on domain-specific datasets.

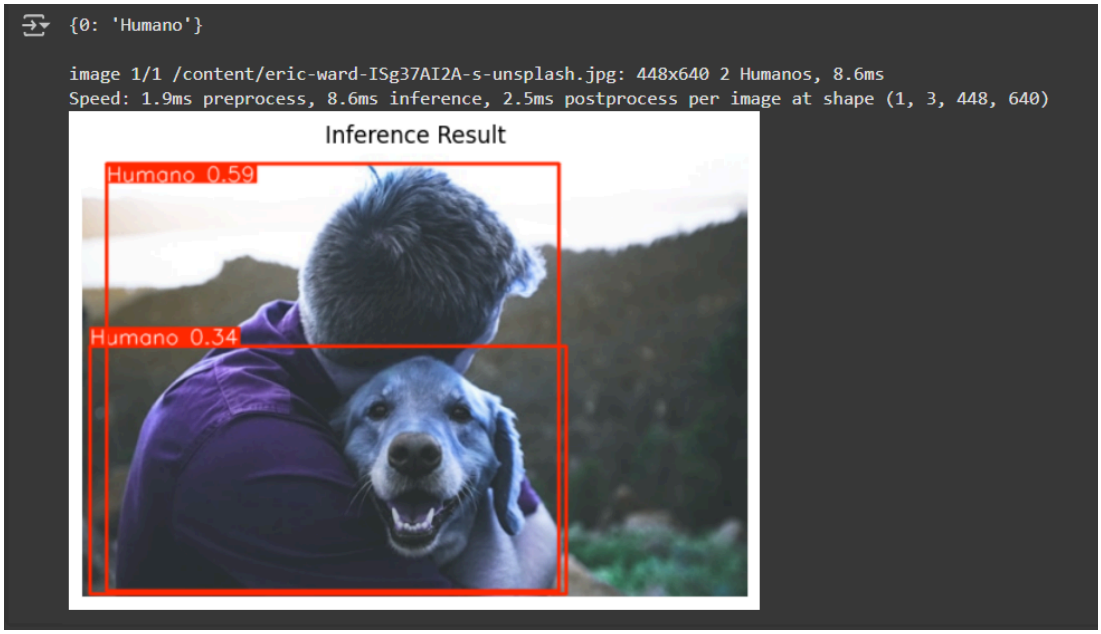

Figure 6: Inference of YOLOv8n model on an image of a human holding a dog, with the person's face not visible

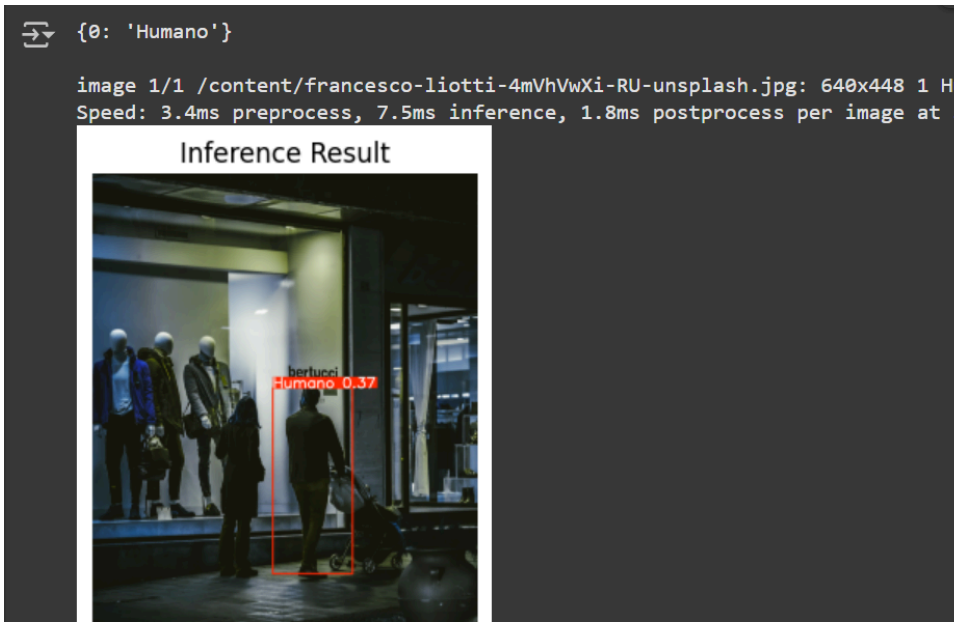

Figure 7: Detection of a human in low-light conditions next to a group of mannequins.

Figure 8: Ground truth of inference via DETR-R50 object detection model

Figure 9: Actual detections by DETR-R50 with each person's probability of being a human mention above their respective bounding boxes.

Figure 10: Another random image in which detections are made accurately across a crowded environment by the DETR-R50 model.

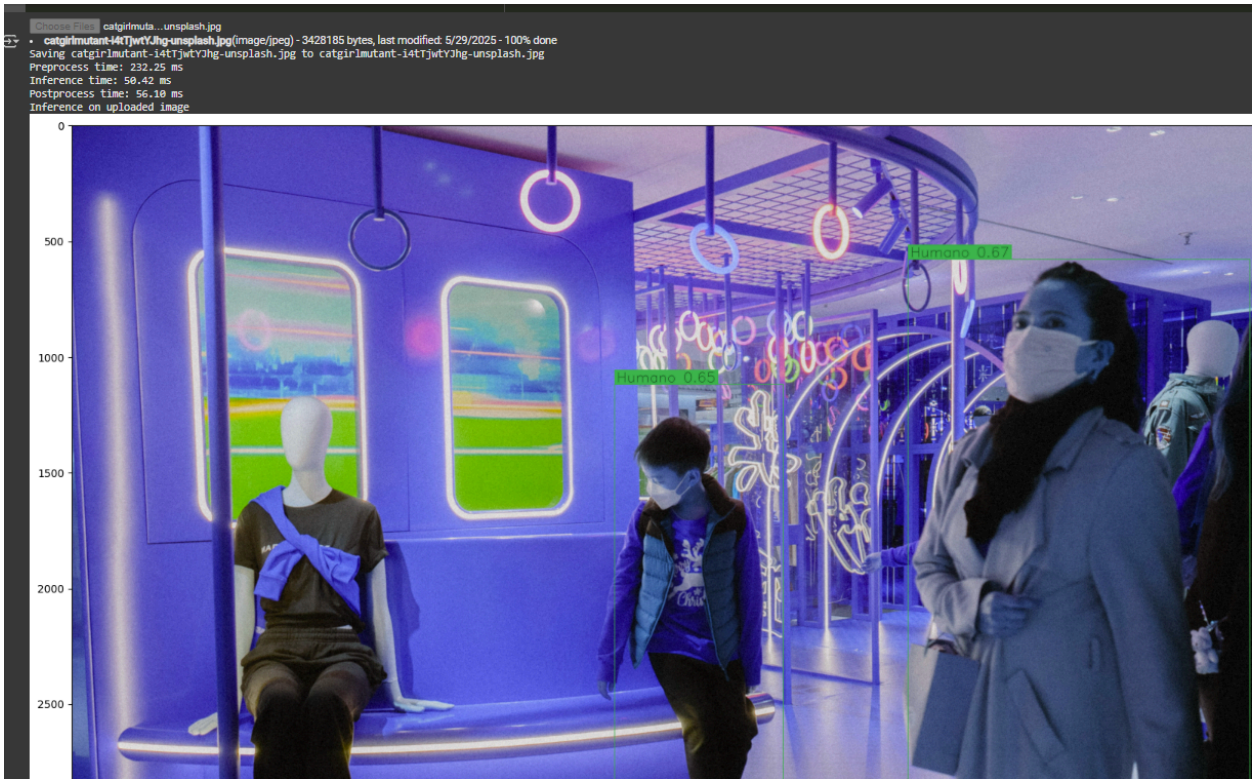

Figure 11: Detecting two humans next to a mannequin, with confidence scores of 0.65 and 0.67.

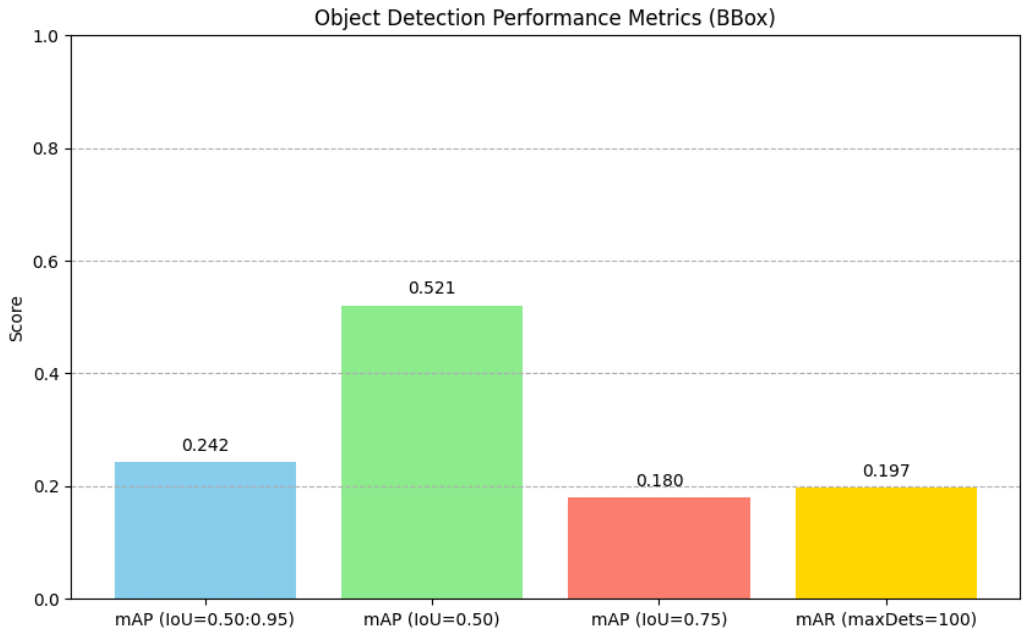

Figure 12: Performance graph for mean Average precision values against their accuracy score.

To offer a visual comparison, Figure 12 depicts a bar graph of key evaluation metrics extracted from the DETR ResNet-50 results. The graph illustrates that the highest AP was recorded at IoU=0.50 (0.521), followed by a steep drop to 0.242 for the full IoU range (0.50:0.95). Precision at stricter thresholds like IoU=0.75 and overall recall also remained low, reinforcing the earlier inference that DETR, while accurate under ideal overlap, struggles under varying object localization tolerances.

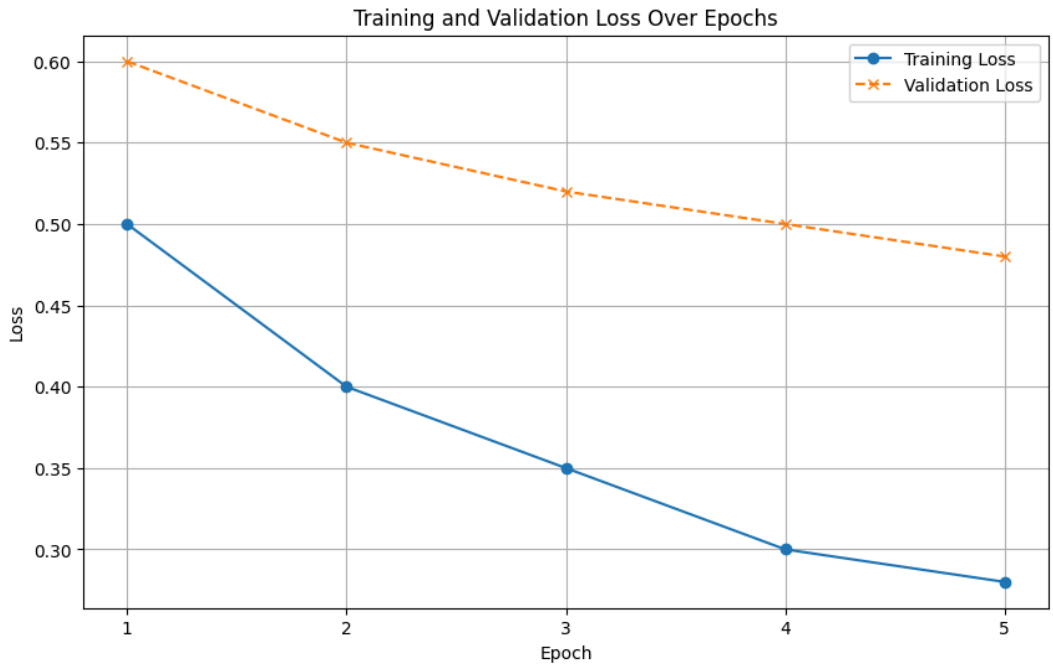

Figure 13: This figure represents the training and validation losses over the whole training of the model.

Figure 13 demonstrates the model's training dynamics across five epochs by plotting the training and validation losses. The graph indicates a consistent decline in both metrics, with training loss reducing from 0.50 to below 0.28 and validation loss from 0.60 to 0.48. This progressive reduction confirms stable convergence and learning of the model over time. However, the persistent gap between training and validation losses suggests a mild overfitting trend, potentially attributable to the dataset's limited size or lack of regularization mechanisms.

Figure 14: Human detection results using the PDSC-YOLOv8n model, accurately identifying three individuals with confidence scores of 0.49, 0.59, and 0.80.

Figure 15: Human detection by the PDSC-YOLOv8n model, accurately identifying one individual with a confidence score of 0.79 in a grayscale urban scene.

Chapter 6: Conclusion

The referenced studies represent significant progress in the domain of human detection using machine learning, addressing critical challenges such as occlusion, pose variability, background clutter, scale variation, and computational limitations. Researchers have introduced a variety of innovative strategies to tackle these problems, ranging from high-resolution representation learning and lightweight convolutional neural networks to sophisticated multi-scale detection techniques. Architectures like YOLOv8 have proven particularly effective for real-time systems due to their speed and compact design, while transformer-based models like DETR ResNet-50 have demonstrated strong performance in static image scenarios, albeit with heavier computational demands. Empirical testing on the custom Humano dataset revealed meaningful insights into the trade-offs between these approaches. The original YOLOv8n model offered a commendable balance between speed and accuracy, demonstrating a 61.05% mAP@0.5 with rapid inference capabilities suitable for low-power environments. In contrast, DETR-ResNet50, although slightly more accurate in broader average precision metrics (as visualized in Fig.13), required over 41 million parameters and exhibited inference latency unsuitable for real-time applications. This underlines a crucial limitation of transformer-based architectures when deployed outside high-performance environments. To address these gaps, the proposed PDSC-YOLOv8n variant introduced architectural optimizations, including enhanced depthwise convolution layers, parameter-efficient skip connections, and adaptive detection heads. These changes resulted in both improved accuracy (63.61% mAP@0.5) and reduced inference time (3.4ms), outperforming both baseline models. This positions PDSC-YOLOv8n as an ideal candidate for real-time human detection in edge-based devices such as surveillance drones or search-and-rescue robots operating under RTOS (Real-Time Operating Systems) constraints. In light of these findings, it becomes evident that future research must not only aim to improve model precision but also emphasize robustness in real-world conditions, such as low-resolution imagery, low-light environments, cluttered or occluded scenes, and distinguishing between actual humans and mannequins or humanoid objects. Creating and training on diverse, high-quality datasets tailored to these challenges will be essential in advancing detection reliability. Ultimately, lightweight yet optimized models like PDSC-YOLOv8n showcase how thoughtful architectural redesign can elevate the capabilities of compact detectors, delivering both speed and accuracy for mission-critical, real-world applications where computational resources are scarce and time is of the essence.

6.1 Future Scope

a. **Expansion of Deployment Scenarios:** Future work can focus on deploying the models in more diverse and realistic environments by integrating them with advanced hardware platforms such as aerial drones, autonomous surveillance systems, and real-time camera modules.

b. **Utilization of Enhanced Hardware:** Leveraging more powerful computational resources, including high-performance GPUs and edge-AI accelerators, can enable real-time processing of higher resolution video streams and improve detection performance under resource-constrained conditions.

c. **Dataset Enrichment:** Further improvements can be achieved by expanding the dataset to include more varied conditions such as thermal filter, aerial shots, different human poses, and complex backgrounds. Inclusion of synthetic data and domain adaptation techniques can also be explored to enhance generalization.

d. **Hybrid Model Architecture:** A promising direction involves developing a hybrid model that combines the speed and lightweight nature of YOLOv8n with the high detection accuracy of transformer-based models like DETR. This fusion could yield a robust detector that maintains real-time capability without compromising precision.

e. **Edge Deployment Optimization:** Future work should also address model compression, quantization, and pruning techniques to make the combined model architecture suitable for deployment on real-time operating systems (RTOS) and other low-power embedded platforms.

f. **Multi-Task Learning:** Exploring multi-task approaches—such as combining human detection with activity recognition or object tracking—can enhance the applicability of the model for surveillance, search and rescue, and smart monitoring systems.

References

1. Sambolek, Sasa, and Marina Ivasic-Kos. "Automatic person detection in search and rescue operations using deep CNN detectors." *Ieee Access* 9 (2021): 37905-37922.
2. Safaldin, Mukaram, Nizar Zaghden, and Mahmoud Mejdoub. "An Improved YOLOv8 to Detect Moving Objects." *IEEE Access* (2024).
3. Ren, Tianhe, et al. "A strong and reproducible object detector with only public datasets." *arXiv preprint arXiv:2304.13027* (2023).
4. Sun, Ke, et al. "Deep high-resolution representation learning for human pose estimation." *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*. 2019.
5. Abdelnabi, Ahmad A. Bany, and Ghaith Rabadi. "Human Detection from Unmanned Aerial Vehicles' Images for Search and Rescue Missions: A State-of-the-Art Review." *IEEE Access* (2024).
6. Ramirez, Heilym, et al. "Fall detection and activity recognition using human skeleton features." *Ieee Access* 9 (2021): 33532-33542.
7. Musa, Usman Shuaibu, et al. "Intrusion detection system using machine learning techniques: A review." *2020 international conference on smart electronics and communication (ICOSEC)*. IEEE, 2020.
8. Ding, Jifeng, et al. "Lightweight enhanced YOLOv8n underwater object detection network for low light environments." *Scientific Reports* 14.1 (2024): 27922.
9. Yue, Min, et al. "Lightweight and efficient tiny-object detection based on improved YOLOv8n for UAV aerial images." *Drones* 8.7 (2024): 276.
10. Wang, Jun, et al. "A lightweight weed detection model for cotton fields based on an improved YOLOv8n." *Scientific Reports* 15.1 (2025): 457.
11. Hao, XiaoYi, and Ting Li. "Lightweight small target detection algorithm based on YOLOv8 network improvement." *IEEE Access* (2025).
12. Tao, Sun, et al. "MIS-YOLOv8: An improved algorithm for detecting small objects in UAV aerial photography based on YOLOv8." *IEEE Transactions on Instrumentation and Measurement* (2025).
13. Wang, Qicheng, Guoqiang Feng, and Zongzhe Li. "A Lightweight Person Detector for Surveillance Footage Based on YOLOv8n." Sensors 25.2 (2025): 436.
14. Li, Tianyang, Jian Wang, and Tibing Zhang. "L-DETR: a light-weight detector for end-to-end object detection with transformers." IEEE access 10 (2022): 105685-105692.
15. Amjoud, Ayoub Benali, and Mustapha Amrouch. "Object detection using deep learning, CNNs and vision transformers: A review." IEEE Access 11 (2023): 35479-35516.
16. Wei, Jiapei, et al. "A Review of YOLO Algorithm and Its Applications in Autonomous Driving Object Detection." IEEE Access (2025).

17. Elgamily, Khaled Mohammed, et al. "Enhanced object detection in remote sensing images by applying metaheuristic and hybrid metaheuristic optimizers to YOLOv7 and YOLOv8." Scientific Reports 15.1 (2025): 7226.
18. Zou, Zhengxia, et al. "Object detection in 20 years: A survey." *Proceedings of the IEEE* 111.3 (2023): 257-276.
19. Padilla, Rafael, Sergio L. Netto, and Eduardo AB Da Silva. "A survey on performance metrics for object-detection algorithms." 2020 international conference on systems, signals and image processing (IWSSIP). IEEE, 2020.
20. Zheng, Zhaohui, et al. "Enhancing geometric factors in model learning and inference for object detection and instance segmentation." IEEE transactions on cybernetics 52.8 (2021): 8574-8586.
21. Ding, Jian, et al. "Object detection in aerial images: A large-scale benchmark and challenges." IEEE transactions on pattern analysis and machine intelligence 44.11 (2021): 7778-7796.
22. Yi, Hao, et al. "Small object detection algorithm based on improved YOLOv8 for remote sensing." IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing 17 (2023): 1734-1747.
23. Aziz, Lubna, et al. "Exploring deep learning-based architecture, strategies, applications and current trends in generic object detection: A comprehensive review." Ieee Access 8 (2020): 170461-170495.
24. Wang, Jinwang, et al. "Tiny object detection in aerial images." 2020 25th international conference on pattern recognition (ICPR). IEEE, 2021.
25. Zhang, Yin, et al. "FFCA-YOLO for small object detection in remote sensing images." IEEE Transactions on Geoscience and Remote Sensing 62 (2024): 1-15.