Object Detection and Person Re-identification for Surveillance System

by

Reema Parikh
202211066

A Thesis Submitted in Partial Fulfilment of the Requirements for the Degree of

MASTER OF TECHNOLOGY
in
INFORMATION AND COMMUNICATION TECHNOLOGY
to

DHIRUBHAI AMBANI INSTITUTE OF INFORMATION AND COMMUNICATION TECHNOLOGY

May, 2024

Declaration

I hereby declare that

i) the thesis comprises of my original work towards the degree of Master of Technology in Information and Communication Technology at Dhirubhai Ambani Institute of Information and Communication Technology and has not been submitted elsewhere for a degree,

ii) due acknowledgment has been made in the text to all the reference material used.

Reema Parikh

Certificate

This is to certify that the thesis work entitled A**" Object Detection and Person Re-identification for Surveillance System "** has been carried out by Reema Parikh for the degree of Master of Technology in Information and Communication Technology at *Dhirubhai Ambani Institute of Information and Communication Technology* under my supervision.

Dr. Manish Khare
Thesis Supervisor

Dr. Amit Mankodi
Thesis Co-Supervisor

Acknowledgments

I would like to express my sincere gratitude to Dr. Manish Khare and Dr. Amit Mankodi for their invaluable guidance and support throughout the duration of this thesis work. Dr. Khare's expertise, encouragement, and unwavering commitment have been instrumental in shaping this research endeavor. His insightful feedback, constructive criticism, and continuous encouragement have significantly enriched the quality of this work. I am profoundly grateful to the esteemed faculty members at DAIICT for their invaluable guidance and unwavering support during my master's program. Their commitment to academic excellence and dedication to nurturing the intellectual development of students have profoundly impacted my academic journey.I would like to acknowledge my classmate Arjun for being my project partner and for his valuable contributions throughout the project.I extend my sincere thanks to Dhruv, Harshit, and Naisargi for their invaluable assistance in collecting the dataset for my project. Their dedication and support greatly contributed to the success of this endeavor. Furthermore, I am deeply thankful to my friends and classmates for their camaraderie, encouragement, and support throughout my master's journey. Their friendship and collaborative spirit have made the learning experience more enriching and enjoyable.

Contents

Abstract

Surveillance systems play a crucial role in ensuring public safety by monitoring and identifying potential threats. At the heart of these systems lie three essential tasks: object detection, person re-identification, and segmentation. Object detection involves accurately locating and classifying objects within surveillance videos, while person re-identification focuses on matching individuals across multiple camera feeds. Complementing these tasks, segmentation delineates regions of interest within the video frame, enabling more nuanced analysis and interpretation.The successful implementation of these tasks holds immense importance as it can significantly enhance the effectiveness of surveillance systems and improve response times to potential security threats. To address the challenges associated with object detection, person re-identification, and segmentation, this thesis suggests a novel approach that combines advanced techniques from computer vision and deep learningThe proposed system integrates state-of-the-art object detection methodologies, such as YOLO v5 [50], with innovative person re-identification techniques, including DeepSort [55], FaceNET [42], and Omni-Scale Feature [60]. Additionally, advanced segmentation algorithms are incorporated to delineate objects and individuals more accurately within surveillance footage. Promising results have been achieved using YOLO v5 [50] for object detection, while the integration of FaceNet [42] addresses issues related to identical outfits in person re-identification tasks. Furthermore, the incorporation of DeepSORT [55] and Omni-Scale Feature Learning [60] enhances the robustness and accuracy of the proposed system.

List of Tables

List of Figures

CHAPTER 1
Introduction

In the realm of public safety and security, surveillance systems are critical tools for monitoring and identifying potential threats. Central to these systems are tasks like object detection, segmentation, and person re-identification, all essential for enhancing surveillance capabilities.Object detection and segmentation use advanced computer vision and deep learning techniques to find and track objects in surveillance footage. Object detection pinpoints and categorizes objects, while segmentation goes further by dividing images or video frames into regions and labeling each area with a specific item. This helps in precise identification and categorization of objects.These techniques are vital across various fields, including video surveillance, robotics, and medical treatment. In surveillance, they help in detecting and tracking people, vehicles, and other objects of interest, aiding in behavior analysis and threat detection. Real-time object detection is crucial, and the YOLO (You Only Look Once) [23] framework stands out for its speed and accuracy. It has evolved over time, improving object recognition algorithms from YOLOv1 to YOLOv8 [49]. However, person re-identification poses unique challenges. Matching a person's appearance across different camera views requires overcoming obstacles like varying viewpoints and occlusion. Deep learning techniques, particularly convolutional neural networks (CNNs), are key in addressing these challenges. This thesis aims to contribute to surveillance technology by reviewing methodologies for person re-identification. It explores advanced methods like Saliency Features and Mask-guided Contrastive Attention Models and proposes a new network leveraging YOLOv5 [50] for pedestrian detection and Deep-SORT [55] for tracking. Through rigorous testing on various datasets, the proposed system's accuracy and speed in real-world scenarios will be evaluated. By enhancing existing surveillance systems, this research aims to improve public safety and security infrastructure. It seeks to advance surveillance techniques in diverse settings, from markets to public spaces, through its exploration of object detection and person re-identification methods.

This thesis is structured into two main parts, each addressing key aspects of surveillance technology.

1. Object Detection and Segmentation

2. Person Re-identification

Object detection and segmentation are fundamental tasks in surveillance systems, utilizing advanced techniques from computer vision and deep learning to locate and track objects in surveillance footage. Object detection identifies and categorizes objects, while segmentation further divides images or video frames into labeled regions. These techniques are vital across various fields, including video surveillance, robotics, and medical imaging. This section of the thesis will offer a detailed examination of methods and progressions in object detection and segmentation. It will explore cutting-edge approaches like the YOLO (You Only Look Once) framework [23], which has evolved significantly to achieve real-time and precise object recognition. Additionally, the thesis will explore recent research and developments in instance segmentation, aiming to identify individual object instances within images or videos.In this section of the thesis, we will delve into methodologies for person re-identification, exploring advanced techniques like Saliency Features, Mask-guided Contrastive Attention Models, and others. We will also propose a novel network architecture leveraging the YOLO v5 [50] framework for pedestrian detection and the Deep-SORT [55] algorithm for tracking. Through thorough evaluation on various datasets, our aim is to assess the accuracy, speed, and reliability of the proposed system in real-world surveillance scenarios.

By dividing the thesis into these two parts, we aim to provide an in-depth exploration of critical techniques and advancements in surveillance technology, ultimately contributing to the enhancement of public safety and security infrastructure.

1.1 Objective

- The objective is to develop and propose a new method that leverages state-of-the-art techniques for object detection and person re-identification, aiming for improved accuracy, speed, and robustness.

- Conduct a comprehensive review of existing methodologies for object detection, segmentation, and person re-identification in surveillance systems.

- Investigate advanced techniques such as Saliency Features, Mask-guided

Contrastive Attention Models, and YOLO v5 [50] for pedestrian detection to enhance object detection and person re-identification capabilities.

- Rigorously evaluate the proposed system on various datasets to measure its accuracy, speed, and reliability in real-world surveillance scenarios.
- Contribute to the advancement of surveillance technology by enhancing object detection, segmentation, and person re-identification methodologies, ultimately improving public safety and security infrastructure.

1.2 Organization

Futhermore, here's an organized structure for the rest of thesis.

- Chapter 2 is the background knowledge and literature survey of the Object Detection , Segmentation and Person Re-Identification methods.
- Chapter 3 outlines the methodology for object detection and segmentation, elucidating the implementation process and resultant findings.
- In Chapter 4, the focus shifts to person re-identification, where a fusion approach is introduced to enhance the existing techniques.It details the implementation and outcomes of this approach.
- Chapter 5 concludes the thesis work by summarizing the potential issues with the proposed approach and key findings.

CHAPTER 2
Background and Related Work

In this section, an overview of fundamental concepts is provided, and a literature review is conducted, encompassing object detection, segmentation, and person re-identification.

2.1 Object Detection and Segmentation

Numerous studies have investigated object identification and segmentation in surveillance videos to enhance public safety and security. Techniques such as deep learning-based object detection and segmentation algorithms have been employed to locate, categorize, and track objects of interest in real-time surveillance footage.

YOLO Framework: The YOLO [23] [33] [49] framework, known for its speed and accuracy in object detection, has undergone several iterations to improve performance and address limitations. Researchers have proposed enhancements to the YOLO [23] architecture, such as optimizing network structures, incorporating attention mechanisms, and integrating contextual information, to achieve better object detection results in surveillance scenarios.

Evolution of YOLO [23]: The evolution of the YOLO framework, from its initial version (YOLOv1) [33] to the latest iterations (e.g., YOLOv8) [49], reflects the continuous efforts to advance object detection techniques. Each version of YOLO [23] [33] [49] has introduced novel features, optimizations, and improvements to overcome challenges and adapt to diverse surveillance environments.

Comparative Studies: Comparative studies have evaluated the performance of different object detection and segmentation approaches, including YOLO [23] [33] [49] variants, against benchmark datasets and real-world surveillance scenarios. These studies provide insights into the strengths and weaknesses of various techniques and help identify the most suitable methods for specific surveillance applications.

Challenges and Future Directions: Despite significant progress, challenges remain in object identification and segmentation in surveillance videos, such

as occlusions, varying lighting conditions, and complex backgrounds. Future research directions may focus on developing robust algorithms, integrating multi-modal sensor data, and exploring novel architectures to address these challenges and further improve surveillance capabilities.

By examining related work in object detection and segmentation, particularly within the context of surveillance videos, researchers can gain valuable insights into the current state-of-the-art techniques, challenges, and future research directions in the field.

2.1.1 Methods of object detection

Object detection is a crucial task in computer vision, with numerous methods developed to accurately locate and classify objects within images or videos. Here, we outline some of the prominent methods used for object detection:

YOLO (You Only Look Once):

YOLO [37] is a real-time object detection system that processes images in a single pass through a neural network, enabling rapid detection of objects. YOLO [37] divides the input image into a grid and predicts bounding boxes and class probabilities for each grid cell. It achieves high speed and accuracy by directly optimizing detection performance during training.

Faster R-CNN (Region-based Convolutional Neural Network):

Faster R-CNN [14] is a two-stage object detection framework that uses a Region Proposal Network (RPN) to generate candidate object regions, which are then classified and refined in a subsequent stage. By decoupling region proposal and classification, Faster R-CNN [14] achieves improved accuracy and efficiency compared to earlier methods.

Single Shot MultiBox Detector:

SSD [30] is a single-shot object detection algorithm that simultaneously predicts bounding boxes and class probabilities for multiple objects in an image. SSD [30] achieves real-time performance by incorporating feature maps of multiple scales to capture objects of different sizes and aspect ratios in a single network pass.

Mask R-CNN:

Mask R-CNN [15] extends Faster R-CNN [14] by adding a branch for predicting segmentation masks in addition to bounding boxes and class labels. This enables precise instance segmentation, where each object instance is segmented and classified separately. Mask R-CNN [15] is widely used for tasks requiring accurate object delineation, such as medical image analysis and autonomous driving.

RetinaNet:

RetinaNet [27] is a single-stage object detection model designed to address the problem of class imbalance in dense object detection. By introducing a novel focal loss function, RetinaNet [27] assigns higher weights to hard examples during training, improving the model's ability to detect rare objects while maintaining high accuracy on common objects.

EfficientDet:

EfficientDet [48] is an efficient object detection model that achieves state-of-the-art performance with significantly fewer parameters compared to previous methods. By optimizing model architecture, feature extraction, and network scaling, EfficientDet [48] achieves a good balance between accuracy and computational efficiency, making it suitable for deployment on resource-constrained devices.

CenterNet:

CenterNet [10] is a keypoint-based object detection approach that directly predicts object center points and bounding boxes from image features. By regressing object centers instead of bounding box corners, CenterNet [10] simplifies the detection process and achieves competitive performance with fewer computational resources.

Cascade R-CNN:

Cascade R-CNN [4] improves object detection accuracy by refining bounding box proposals in a cascade of stages, each focusing on different aspects of object localization. By progressively filtering out false positives and refining bounding box coordinates, Cascade R-CNN [4] achieves superior performance on challenging datasets.

These methods represent a diverse range of approaches to object detection, each with its strengths and limitations. Researchers continue to explore

novel architectures, loss functions, and training strategies to further advance the field of object detection and address emerging challenges in real-world applications.

2.1.2 Methods of segmentation

Image segmentation is a fundamental task in computer vision, involving partitioning an image into meaningful regions to simplify its representation and facilitate further analysis. Here are some commonly used methods for image segmentation:

Thresholding:

Thresholding [53] is a simple and widely used technique for image segmentation, where pixels are classified as foreground or background based on their intensity values relative to a predefined threshold. Thresholding [53] methods include global thresholding [53], where a single threshold is applied to the entire image, and adaptive thresholding [53], where thresholds are computed locally based on image characteristics.

Edge Detection:

Edge detection methods [43] aim to identify boundaries between different regions in an image by detecting abrupt changes in pixel intensity. Techniques such as the Canny edge detector, Sobel operator, and Prewitt operator are commonly used to highlight edges in images, which can then be used as cues for segmentation.

Region Growing:

Region growing methods [11] start with seed points and iteratively grow regions by merging neighboring pixels that satisfy certain criteria, such as similarity in intensity or texture. This process continues until no more pixels can be added to the regions, resulting in segmentation based on homogeneous regions.

Clustering:

Clustering [8] techniques group pixels into clusters based on their feature similarity, such as color, intensity, or texture. K-means clustering [8] is a popular method for image segmentation, where pixels are partitioned into K clusters

based on their Euclidean distance to cluster centroids. Other clustering [8] algorithms, such as hierarchical clustering [8] and spectral clustering [8] , can also be used for image segmentation.

Watershed Transform:

The watershed transform treats [56] pixel intensities as topographic elevations and simulates the flooding of a landscape to partition the image into catchment basins. Watershed segmentation [56] is particularly useful for segmenting objects with well-defined boundaries but can suffer from over-segmentation in noisy or textured regions.

Graph-based Segmentation:

Graph-based segmentation methods [13] represent an image as a graph, where pixels are nodes and edges represent relationships between pixels. Segmentation is then formulated as a graph partitioning problem, where the goal is to partition the graph into disjoint regions that minimize a cost function. Graph cuts and normalized cuts are examples of graph-based segmentation techniques [13].

Deep Learning-based Segmentation:

Deep learning-based segmentation methods [32] leverage convolutional neural networks (CNNs) to learn hierarchical features directly from image data. Fully convolutional networks (FCNs) [31], U-Net [41], and SegNet are popular architectures for semantic segmentation, where each pixel is classified into predefined classes. Instance segmentation methods, such as Mask R-CNN [15], extend semantic segmentation by predicting segmentation masks for each object instance separately.
Several deep convolutional neural network (CNN) models are capable of both object detection and segmentation tasks. These models leverage their ability to extract hierarchical features from images to perform both tasks simultaneously or sequentially. Here are some notable deep CNN models capable of object detection and segmentation:

- **Mask R-CNN:** Mask R-CNN [15] is an extension of the Faster R-CNN [14] object detection framework that adds a branch for predicting segmentation masks alongside bounding box predictions. It enables precise instance segmentation by generating segmentation masks for each detected object in an image. Mask R-CNN [15] achieves state-of-the-art performance in both object detection and segmentation tasks.

- **Cascade R-CNN:** Cascade R-CNN [4] is a variant of Faster R-CNN [14] that improves object detection performance by employing a cascade of detectors with increasing levels of difficulty. In addition to object detection, Cascade R-CNN [4] can be adapted to perform instance segmentation by incorporating a mask prediction branch similar to Mask R-CNN.

- **YOLO (You Only Look Once) with Segmentation Head**: YOLO is a real-time object detection model that divides an image into a grid and predicts bounding boxes and class probabilities directly from the grid cells. YOLOv3 [38] and YOLOv4 [2] have been extended to include segmentation heads that predict segmentation masks alongside object detections, enabling simultaneous object detection and segmentation.

- **EfficientDet:** EfficientDet [48] is a family of efficient object detection models that achieve high accuracy with relatively few parameters. By integrating lightweight segmentation heads, EfficientDet [48] models can perform object detection and segmentation tasks concurrently. These models offer a good trade-off between speed and accuracy for joint object detection and segmentation.

- **DETR (DEtection TRansformer):** DETR [5] is a transformer-based object detection model that uses a set-based prediction approach rather than anchor boxes. While not originally designed for segmentation, DETR [5] can be adapted to perform segmentation by adding a segmentation head that generates masks for detected objects, similar to Mask R-CNN.

- **DeepLabv3+:** DeepLabv3+ [7] is a deep CNN model specifically designed for semantic image segmentation. While its primary focus is segmentation, DeepLabv3+ [7] can be combined with object detection models such as Faster R-CNN [14] or YOLO to perform joint object detection and segmentation tasks. The segmentation masks generated by DeepLabv3+ [7] can provide additional context for detected objects.

These deep CNN models demonstrate the capability to perform both object detection and segmentation tasks effectively. By integrating segmentation heads or adapting existing architectures, these models can generate precise object masks while detecting objects in images, facilitating various applications such as scene understanding, autonomous driving, and medical imaging analysis.

2.2 Person Re-identification

Person re-identification (Re-ID) in surveillance videos is a vital task for public safety and security, involving the identification of individuals across different video sequences. Deep learning has emerged as a powerful tool to tackle the challenges inherent in accurate and efficient person re-identification, such as changes in viewpoint, occlusion, and pose variation.

In a recent survey conducted by Khawar Islam [22], an in-depth overview of advancements in deep learning methodologies for video re-ID was presented, covering major milestones, technical challenges, and comparative performance analysis across various datasets. Swathi Jamjala Narayanan et al. [34] proposed a novel approach to person re-identification by integrating deep learning-based human body part segmentation and Gaussian filtering-based smooth mask generation.

Nico Klingler [24] discussed the efficacy of deep learning in person reidentification, shedding light on its adaptation to diverse challenges encountered in surveillance videos. Additionally, Fatih AKSU et al. [1] introduced a comprehensive person re-identification system leveraging deep convolutional neural networks for feature extraction and similarity calculation, aiming to improve accuracy and efficiency in identifying individuals across video sequences.

2.2.1 Deep Learning Methodologies for Feature Extraction

A. DenseNet 121:

DenseNet121 [19], a convolutional neural network (CNN) architecture, is characterized by dense connections between layers, forming dense blocks. Each layer within a dense block receives inputs from all preceding layers and passes its own feature maps to all subsequent layers. This design mitigates the vanishing gradient problem and promotes feature reuse, leading to more compact models and improved performance. DenseNet [19]-121 architecture comprises various layers, including a 7x7 convolution layer, 58 3x3 convolution layers, 61 1x1 convolution layers, 4 average pooling layers, and 1 fully connected layer. Additionally, transition layers between dense blocks incorporate down-sampling on feature maps through a 1x1 convolution and a 2x2 average pooling layer. DenseNet-121 [19] has been successfully applied in diverse applications, including person re-identification, where it effectively addresses scalability challenges, particularly in resource-intensive settings.

Output:

Observations: Upon utilizing DenseNet121 [19], it was observed that 50% of individuals consistently maintained their identities throughout the imple-

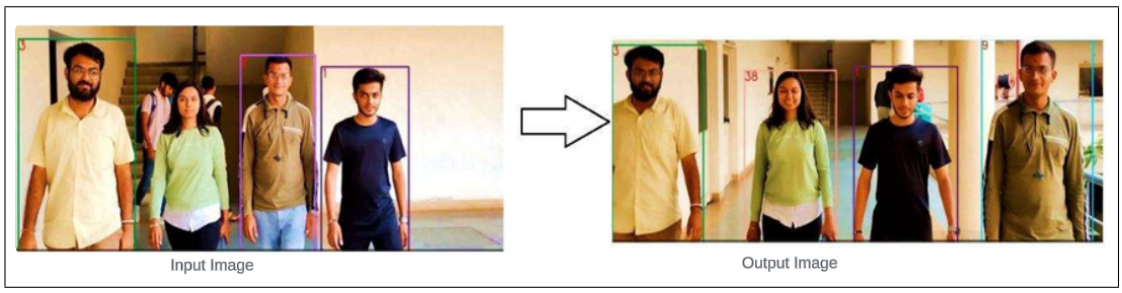

Figure 2.1: DenseNet 121 [19] on DECDC Dataset

mentation. However, the remaining 50% of individuals experienced multiple identity changes within the videos.

B. ResNet 18 [44] [9]

ResNet18 [44] [9], renowned for its convolutional neural network architecture, has proven its efficacy across diverse computer vision tasks, notably in image classification. Extensive applications of ResNet18 [44] [9] include cardiac auscultation classification, clothing recognition, remote sensing image classification, and medical image classification, where it has demonstrated exceptional performance. While explicit mentions of ResNet18's [44] [9] application in person re-identification were not found in available literature, its success in image classification tasks suggests its suitability for person re-identification, a related task involving identifying individuals across various images or video frames. Thus, ResNet18 [44] [9] stands as a well-established architecture with a robust track record in computer vision tasks, positioning it as a promising candidate for person re-identification.

Output:

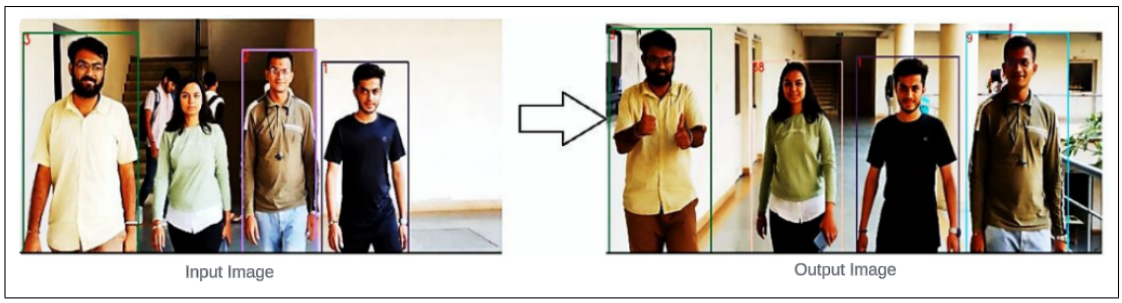

Figure 2.2: Resnet 18 on DECDC Dataset

Observations: Upon employing ResNet18 [44] [9], it was observed that 50% of individuals consistently maintained their identities, while the remaining 50% underwent multiple identity changes between videos. These results parallel those obtained with DenseNet [19], indicating comparable performance between the two architectures and highlighting their potential in person re-identification scenarios.

C. SE-ResNet50

SE-ResNet50 [18] represents an evolution of the ResNet50 architecture by integrating Squeeze-and-Excitation (SE) blocks. These SE blocks enhance feature recalibration by dynamically adjusting the importance of different channels, thereby improving model attention and overall performance. SE blocks incorporate a global average pooling operation, followed by two small fully connected layers and a sigmoid activation function. The output of these layers is then multiplied by the input feature map, facilitating a channel-wise scaling operation. Demonstrating its effectiveness, the SE-ResNet50 [18] model achieves a top-1 error rate of 22.28% on the ImageNet dataset, surpassing the original ResNet50 [18] model by 1.3%.

Output:

Figure 2.3: SE-ResNet50 on DECDC Datase

Observations: Upon implementing SE-ResNet50 [18], it was observed that 25% of individuals consistently maintained their identities, while 75% underwent multiple identity changes between videos.

D. NasNetMobile

NasNetMobile [61] is a convolutional neural network tailored for mobile and resource-constrained devices, developed by Google. It utilizes reinforcement learning to autonomously discover optimal neural network architectures. Trained on over a million images from the ImageNet database, NasNetMobile [61] excels in classifying images into 1000 object categories with an input size of 224-by-224. Specifically tuned for mobile phone CPUs, it stands as one of the prominent architectures for image recognition and computer vision tasks.

Output:

Observations: Upon deploying NasNetMobile [61], it was observed that its architecture consists of a global average pooling operation, followed by two small fully connected layers and a sigmoid activation function. This design enables efficient scaling operations on channel-wise features. Furthermore, NasNetMobile [61] has demonstrated promising results with a top-1 error rate

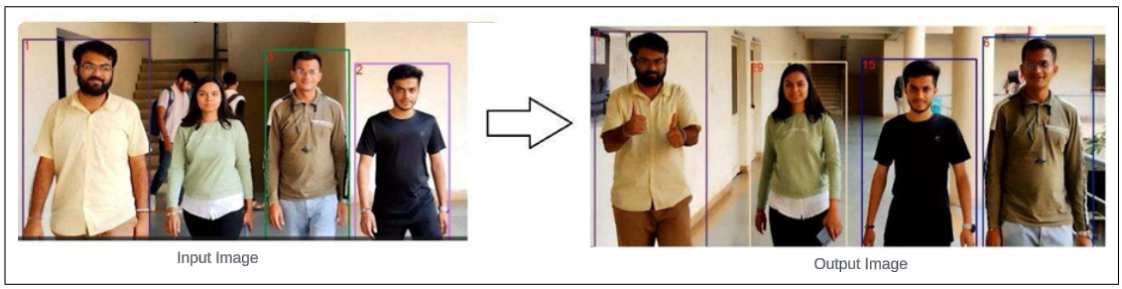

Figure 2.4: NasNetMobile on DECDC Dataset

of 22.28% on the ImageNet dataset, representing a notable improvement over the original ResNet50 model.

E. Inception

Inception [47], also known as GoogleNet, represents a groundbreaking deep convolutional neural network architecture introduced by Google in 2014. Distinguished by its inception [47] modules, this architecture revolutionizes feature extraction by employing multiple filter sizes within each layer, enabling the capture of features at various scales simultaneously. Inception [47] modules integrate a combination of 1x1, 3x3, and 5x5 convolutions to learn diverse feature maps across different scales.

Introduced to address computational inefficiencies in deeper networks, the Inception [47] architecture has significantly influenced subsequent convolutional neural network designs. Inception v1 [47], the original version, comprises nine inception [47] modules and achieved state-of-the-art performance on the ImageNet dataset upon release. Subsequent iterations, such as Inception v2 [47] introduced in 2015, further enhance CNN performance.

Key features of Inception [47] modules include multi-level feature extraction, reduced overfitting, improved performance, and dimensionality reduction through 1x1 convolutions. Despite its complexity and the need for additional hyperparameters, the Inception [47] architecture stands as a pivotal milestone in the evolution of deep learning architectures.

Output:

Figure 2.5: Inception model on DECDC Dataset

Observations: Upon utilization, the Inception [47] architecture revealed that 25% of individuals consistently maintained their identities, while 75% experienced multiple identity changes across videos.

F. ShuffleNet

ShuffleNet [57], a convolutional neural network architecture tailored for mobile and edge devices, introduces innovative operations like pointwise group convolution and channel shuffle. These operations effectively reduce computational complexity while preserving competitive accuracy levels. Introduced in 2017, ShuffleNet [57] has demonstrated superiority over other architectures like MobileNet within similar computational budgets. For instance, it achieved a top-1 error rate of 7.8% on the ImageNet classification task, surpassing MobileNet's 9.6% error rate under the same 40 MFLOPS computation budget.

The architecture of ShuffleNet [57] comprises stacked ShuffleNet [57] units organized into three stages, with bottleneck channels set to 1/4 of the output channels for each unit. A scale factor adjusts the channel count to tailor the network's complexity. ShuffleNet v2 [57], an optimized version, prioritizes direct speed improvements over indirect metrics like FLOPs. It introduces new operations like channel split and relocates the channel shuffle operation for enhanced efficiency. ShuffleNet [57] v2 has demonstrated competitive performance across various tasks such as image classification, object detection, and semantic segmentation.

Due to its simple design and efficient operations, ShuffleNet [57] has gained popularity for mobile and edge applications where computational resources are limited.

Output:

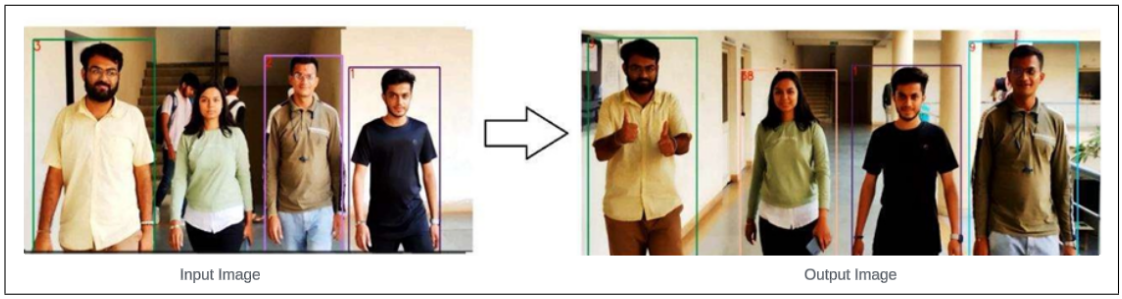

Figure 2.6: ShuffleNet on DECDC Dataset

Observations: Upon employing ShuffleNet [57], it was observed that 50% of individuals consistently maintained their identities, while the remaining 50% underwent multiple identity changes across videos.

G. SqueezeNet

SqueezeNet [21] is a neural network architecture specifically engineered for efficient model inference and deployment on devices with limited computational resources. It was devised to address the escalating demand for smaller models with reduced parameters. The architecture's compactness stems from its unique design, incorporating 1x1 convolutions, a fire module, and aggressive downsampling techniques.

Central to SqueezeNet [21] is the fire module, comprising a squeeze convolutional layer followed by an expand convolutional layer. While the squeeze layer diminishes input spatial dimensions, the expand layer augments channel numbers. This strategic design enables SqueezeNet [21] to maintain an optimal performance-to-parameters ratio. The architecture is organized into 12 convolutional blocks, each featuring either a 1x1 or 3x3 convolution with 32 filters and a stride of 1. Batch normalization and Leaky ReLU activation with a = 0.1 follow suit. This constrained approach yields significantly improved results while ensuring parameter efficiency.

SqueezeNet's [21] efficacy extends across diverse domains, such as gesture recognition, where it has exhibited superior effectiveness compared to contemporaneous alternatives, all while boasting a markedly reduced parameter count. Furthermore, it seamlessly adapts to process images of various sizes, such as those in the CIFAR-10 dataset, without compromising classification performance.

Output:

Figure 2.7: SqueezeNet on DECDC Dataset

Observations: Upon leveraging SqueezeNet [21], it was observed that 50% of individuals consistently maintained their identities, while the remaining 50% underwent multiple identity changes across videos. Notably, SqueezeNet [21] and ShuffleNet [57] demonstrated analogous performance in this aspect.

H.Multi-Level Factorization Net

The Multi-Level Factorization Net (MLFN) [6] represents a pioneering network architecture tailored for person re-identification (Re-ID), aimed at over-

coming the hurdles associated with modeling discriminative and view-invariant factors across varying semantic levels without the need for laborious human annotation. Comprising multiple stacked blocks, MLFN [6]integrates factor modules crafted for specific semantic levels, alongside factor selection modules dynamically selecting pertinent factor modules to decipher input image content.

This architecture has demonstrated state-of-the-art performance across three Re-ID datasets and delivered compelling outcomes on the general object categorization CIFAR-100 dataset. MLFN's [6] groundbreaking aspect lies in its capacity to autonomously unearth latent discriminative factors sans manual annotation, marking a promising stride in unsupervised person Re-ID.

Output:

Figure 2.8: MLFN on DECDC Dataset

Observations: Upon employing MLFN [6], it was noted that 25% of individuals consistently maintained identical identities, while 75% underwent multiple identity changes across videos.

I. Harmonious Attention CNN (HA-CNN)

The Harmonious Attention CNN (HA-CNN) [26] model represents a sophisticated deep learning framework engineered to overcome the constraints of existing person re-identification (re-id) methodologies, especially in scenarios characterized by arbitrary alignments, significant pose variations, and uncontrolled auto-detection errors. HA-CNN [26] integrates soft pixel attention, hard regional attention, and feature representation learning within a Convolutional Neural Network (CNN), strategically maximizing the utilization of complementary information to bolster discriminative capabilities.

Extensively evaluated across prominent benchmarks like CUHK03, Market-1501, and DukeMTMC-ReID, HA-CN [26]N has consistently outperformed state-of-the-art methods, showcasing its efficacy in challenging real-world scenarios. Its key advantage lies in its ability to concurrently learn attention selection and feature representation, optimizing person re-id even in instances of

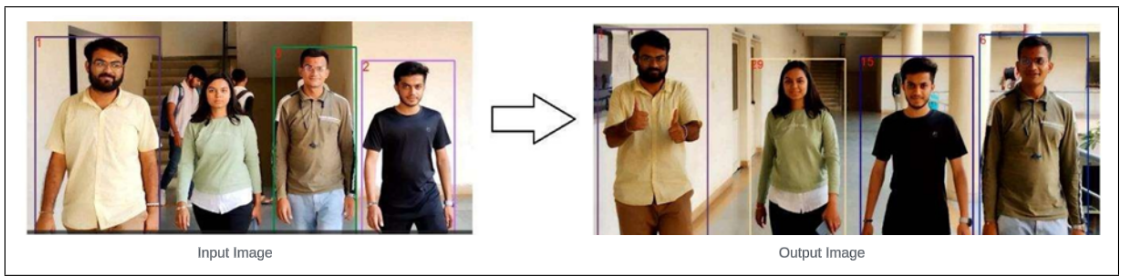

Figure 2.9: HA-CNN on DECDC Dataset

image misalignment. Moreover, HA-CNN's [26] emphasis on contour information contributes significantly to its success in person re-identification tasks.
Highlighted Contributions of HA-CNN [26]:
1. Joint learning of soft pixel attention and hard regional attention.
2. Simultaneous optimization of feature representations.
3. Enhanced discriminative capabilities for person re-id in uncontrolled (misaligned) images.
4. Superior performance demonstrated on large-scale benchmarks.

The model's efficacy has been validated through extensive comparative evaluations, solidifying its position as a leading solution in the realm of person re-identification.
Output:
Observations: Upon implementing HA-CNN [26], it was observed that 25% of individuals maintained consistent identities, while 75% exhibited multiple identity changes across videos.

J. MuDeep

MuDeep [36], a multi-scale deep learning architecture tailored for person re-identification tasks, is designed to match individuals across disparate camera views in public environments. The model excels in learning discriminative feature representations across multiple scales, automatically determining the most effective scales for accurate matching. MuDeep [36] surpasses state-of-the-art models on various benchmarks, underscoring its effectiveness in addressing the complexities of person re-identification in surveillance settings.

Comprising five key components including tied convolutional layers, multi-scale stream layers, a saliency-based learning fusion layer, and sub-nets for person re-identification, MuDeep [36] delivers robust performance in matching individuals across different camera views.
Output:
Observations: Upon deploying MuDeep [36], it was observed that 50% of individuals consistently maintained their identities, while the remaining 50%

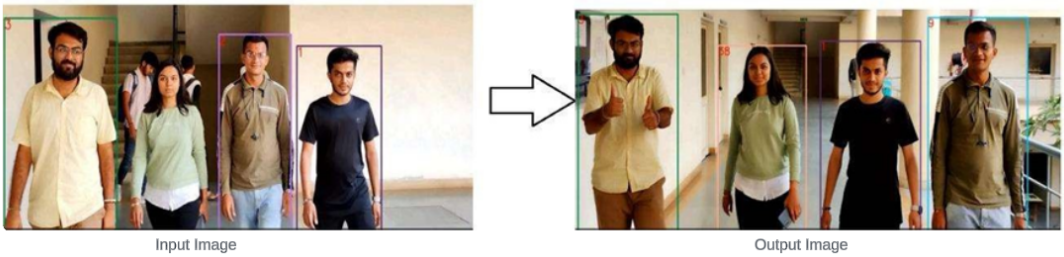

Figure 2.10: MuDeep on DECDC Dataset

underwent multiple identity changes across videos.

K. Part-based Convolutional Baseline (PCB)

The Part-based Convolutional Baseline (PCB) [46] network adopts a unique approach to construct a convolutional descriptor by extracting diverse part-level features using a consistent partitioning strategy. This methodology yields competitive results when compared to existing state-of-the-art methods, establishing PCB [46] as a dependable convolutional baseline for person retrieval tasks. Additionally, the introduction of the refined part pooling (RPP) technique aims to address outliers within uniformly partitioned parts. This refinement significantly improves within-part consistency, leading to notable performance enhancements for PCB [46]. Notably, the implementation of RPP surpasses current state-of-the-art results on the Market-1501 dataset.
Output:

Figure 2.11: PCB on DECDC Dataset

Observations: Observations: Upon utilizing PCB [46], it was observed that 50% of individuals consistently maintained their identities, while the remaining 50% experienced multiple identity changes across videos.

K. Omni-Scale Network (OSNet)

The Omni-Scale Network (OSNet) [60] is an innovative deep learning architecture tailored for person re-identification (ReID) tasks. It features a residual block with multiple convolutional streams, each adept at detecting fea-

tures across different scales. OSNet [60] introduces a unified aggregation gate, dynamically fusing multi-scale features with input-dependent channel-wise weights. By employing pointwise and depthwise convolutions, it enhances spatial-channel correlations while curbing overfitting, resulting in an exceptionally lightweight model. Across six person ReID datasets, OSNet [60] achieves state-of-the-art performance, surpassing larger models by a significant margin.
Output:

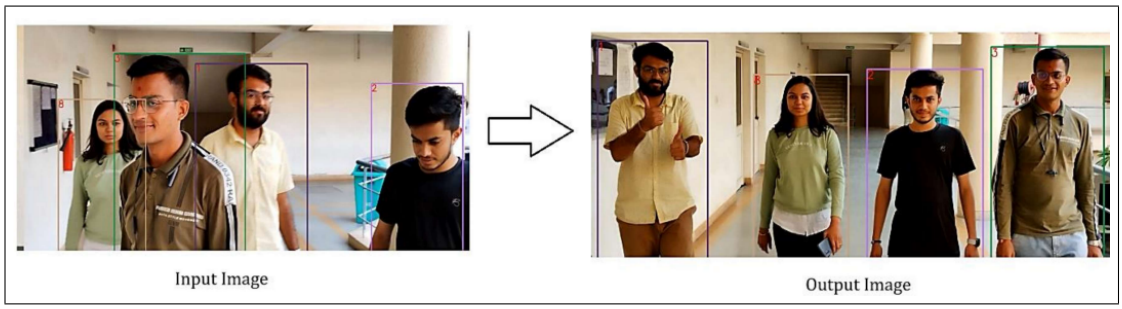

Figure 2.12: OSNet on DECDC Dataset

Observations: In our investigation, OSNe [60]t exhibited promising results on the DECDC dataset, with 75% of individuals maintaining consistent identities and 25% undergoing multiple identity changes between videos. This underscores the effectiveness of OSNet in handling person re-identification tasks, especially in scenarios with varied scales and challenging conditions.

These methodologies collectively provide a comprehensive exploration of deep learning approaches for person re-identification, offering valuable insights into their applicability and performance across diverse scenarios encountered in surveillance video analysis.

CHAPTER 3
Object detection and Segmentation

In this section, we delve into the methodologies, datasets, implementation details, and results pertaining to object detection and segmentation. We start by exploring various versions of the You Only Look Once (YOLO) architecture, ranging from YOLOv1 to YOLOv8 [49], highlighting their key features and advancements. Subsequently, we discuss the datasets utilized for training and evaluation, including the PASCAL VOC 2012 [12] and MS COCO datasets [28]. Following this, we provide insights into the implementation specifics, focusing particularly on the YOLO architecture [33]. Finally, we present the results obtained from our experiments, with a special emphasis on analyzing the trade-off between speed and accuracy in object detection and segmentation tasks.

Object identification and segmentation methods for surveillance videos represent a crucial aspect of modern computer vision and deep learning research. These techniques aim to detect, track, and categorize objects within surveillance footage, enabling a wide range of applications in public safety, security, and beyond. Identifying objects in surveillance videos involves locating and categorizing objects within a frame or video sequence. On the other hand, segmentation techniques go a step further by dividing images or video frames into distinct regions and associating each area with a specific object or class. Instance segmentation, a more advanced form of segmentation, delineates individual object boundaries and assigns labels according to their classes, providing detailed information beyond mere object identification.

Importance and Applications: Object identification and segmentation play a pivotal role in various fields, including video surveillance, robotics, medical imaging, and human-computer interaction. In the context of video surveillance, these techniques enable the detection and tracking of people, vehicles, and other objects of interest. This information can be utilized to analyze behavior patterns, monitor traffic flow, and identify potential security threats, thereby enhancing public safety and security.

Challenges and Solutions: Semantic segmentation and instance segmentation are two primary approaches to the segmentation task, each with its unique

challenges and applications. While semantic segmentation labels each pixel in an image with a matching class, instance segmentation aims to identify individual instances of each class with precise boundaries
. **The YOLO Framework** Real-time object detection has become increasingly important in surveillance and other applications. The You Only Look Once (YOLO) [23] [33] [49] framework has emerged as a prominent solution due to its exceptional balance of speed and accuracy. The YOLO family of algorithms has evolved over time, with each iteration building upon the strengths of its predecessors to address challenges and improve performance.
Research Objectives This thesis aims to provide a comprehensive review of the YOLO framework's development, from the original YOLOv1 to the latest YOLOv8 [51]. By elucidating key innovations, differences, and improvements across different versions, this research seeks to contribute to the understanding and advancement of object detection and segmentation methods for surveillance videos.

3.1 Methodology

3.1.1 YOLOv1

YOLOv1 [33] revolutionized real-time object detection by introducing a unified approach to predict bounding boxes and class probabilities within a single neural network. Its architecture divided the input image into a grid of cells and generated predictions for each cell, allowing for efficient inference. With 24 convolutional layers followed by 2 fully connected layers, YOLOv1 [33] achieved impressive performance while maintaining real-time processing speeds. However, its fixed grid cell size limited its ability to detect small or densely packed objects effectively. Despite this limitation, YOLOv1 [33] laid the groundwork for subsequent advancements in object detection.

3.1.2 YOLOv2

Building upon the success of YOLOv1 [33], YOLOv2 introduced significant improvements to object detection. By incorporating the Darknet neural network framework with 53 convolutional layers and 4 detection layers, YOLOv2 [20] achieved better accuracy and robustness. One of its key innovations was the introduction of anchor boxes, enabling the model to detect objects with varying aspect ratios and sizes more effectively. Additionally, YOLOv2 [20] employed batch normalization and a novel loss function, enhancing training stability and convergence speed. These advancements propelled YOLOv2 [20] to the fore-

front of real-time object detection systems.
By combining anchor boxes, YOLOv2 [20] improved upon YOLOv1 [33] by removing its limitations on the network's ability to identify objects with various aspect ratios and sizes. To improve training accuracy and stability, batch normalization is also used. In addition, YOLOv2 [20] uses a novel loss function that penalizes classification errors differently depending on the size of the object and the confidence level. YOLOv2 [20] filters out duplicate bounding boxes during inference using NMS and a confidence threshold.

3.1.3 YOLOv3

A deep convolutional neural network (CNN) is used by the top-of-the object recognition system YOLOv3 [38] to identify and locate objects in pictures. Three sections make up its architecture. The Backbone uses a modified version of DarkNet with 53 convolutional layers and residual connections to extract features from the input picture, allowing for feature extraction at different dimensions.

The Neck uses a feature pyramid network (FPN) to identify objects of various sizes and aspect ratios by combining information from various scales to build a semantically rich feature map.

Using anchor boxes of various sizes and aspect ratios, The Head predicts items location, shape, objectness scores, and class probabilities.
In comparison to YOLOv2 [20], YOLOv3 [38] offers a number of improvements, including SPP for creating feature maps with multiple scales, various scales detection for improved precision, softmax loss function for improved classification, and a higher IOU threshold throughout training to decrease false positives and improve detection accuracy.

3.1.4 YOLOv4

YOLOv4 [2] marked a significant leap forward in object detection capabilities with its innovative architecture and advanced techniques. By leveraging features like CSPNet, a modified Mish activation function, and Self-Adversarial Training, YOLOv4 [2]achieved superior accuracy while reducing the number of parameters. This resulted in faster inference speeds and reduced false positives. Additionally, YOLOv4 [2] introduced advanced data augmentation techniques like CutMix and Mosaic, further improving detection accuracy and generalization. With its state-of-the-art performance and efficiency, YOLOv4 [2] solidified its position as a leading object detection system.

3.1.5 YOLO v5

YOLO v5 [50], developed by Ultralytics, represented a paradigm shift in object detection with its transition to the PyTorch framework and introduction of a modified CSP architecture. Offering various model sizes (e.g., YOLOv5s, YOLOv5m, YOLOv5l, YOLOv5x), YOLOv5 achieved superior accuracy and efficiency across a wide range of tasks. By starting with a backbone network (efficient net or CSPDarknet) for feature extraction, followed by additional convolutional layers for abstract feature representation, YOLO v5 [50] demonstrated remarkable performance improvements over previous versions. Its ease of use, flexibility, and impressive results propelled YOLO v5 [50] to become the go-to choice for many object detection applications.

3.1.6 YOLOv6

EfficientRep, a new backbone used by YOLOv6 [25], is based on RepVGG and takes advantage of more parallelism than earlier YOLO backbones. Additionally, the model incorporates additional classification and regression losses, such as a classification VariFocal loss and a SIoU/GIoU regression loss, as well as label assignment utilizing the Task alignment learning technique. In order to create a more rapidly detector, YOLOv6 [25] additionally uses a self-distillation technique for the regression and classification tasks as well as a quantization scheme for detection utilizing RepOptimizer and channel-wise distillation.

3.1.7 YOLOv7

The design modifications and new features added in YOLOv7 [52] are the primary distinctions between YOLOv6 [25] and YOLOv7 [52]. The Extended efficient layer aggregation network (E-ELAN) design, which improves network learning without eliminating the initial gradient route, was proposed by YOLOv7 [52]. To retain the ideal structure of the model, YOLOv7 [52] also proposed a novel scaling technique for concatenation-based models. These modifications, coupled with a number of bonus items, improved the accuracy of YOLOv7 [52] without slowing down inference speed; only training time was affected.

3.1.8 YOLOV8

Ultralytics introduced YOLOv8 [39] as an anchor-free object identification model, representing a significant advancement in the YOLO series. This iteration aims to streamline the object detection process by speeding up Non-Maximum Suppression (NMS) and reducing the number of box predictions, thereby improving efficiency and accuracy. By eliminating the need for anchor boxes, YOLOv8 [39] simplifies the detection pipeline while maintaining high performance.

One notable feature of YOLOv8 [39] is its utilization of mosaic augmentation throughout training. Mosaic augmentation involves combining multiple images into a single training batch, allowing the model to learn from diverse perspectives and backgrounds. However, to mitigate potential downsides, mosaic augmentation is turned off for the final 10 epochs of training, ensuring optimal performance during the later stages of training.

In addition to its innovative architecture, YOLOv8 [39] offers several integrations for labeling, training, and deployment, making it accessible and user-friendly. Users can leverage YOLOv8 [39] from the Command Line Interface (CLI) or deploy it as a PIP package, simplifying the workflow for both researchers and practitioners. This flexibility and ease of use contribute to YOLOv8 [39]'s appeal as a powerful and versatile object detection solution.

Overall, YOLOv8 [39] represents the culmination of years of research and development in the field of object detection. With its anchor-free approach, efficient NMS optimization, and robust training pipeline, YOLOv8 [39] offers state-of-the-art performance and usability, making it a valuable asset for a wide range of computer vision applications.

3.2 DataSet Used

3.2.1 PASCAL VOC 2012

The PASCAL Visual Object Classes(VOC) 2012 [12] dataset includes 20 diffrent object categories, covering home items, domestic animals, and other: an aeroplane, a bicycle, a boat, a bus, a car, a motorcycle, a train, a bottle, a chair, a dining table, a potted plant, a couch, a TV/monitor, a bird, a cat, a cow, a dog, a sheep, and a human.

The PASCAL VOC 2012 dataset [12] is a widely recognized benchmark in the field of computer vision, particularly for tasks like object detection, classification, and segmentation. It comprises a diverse collection of images, each annotated with bounding boxes delineating objects of interest and corresponding

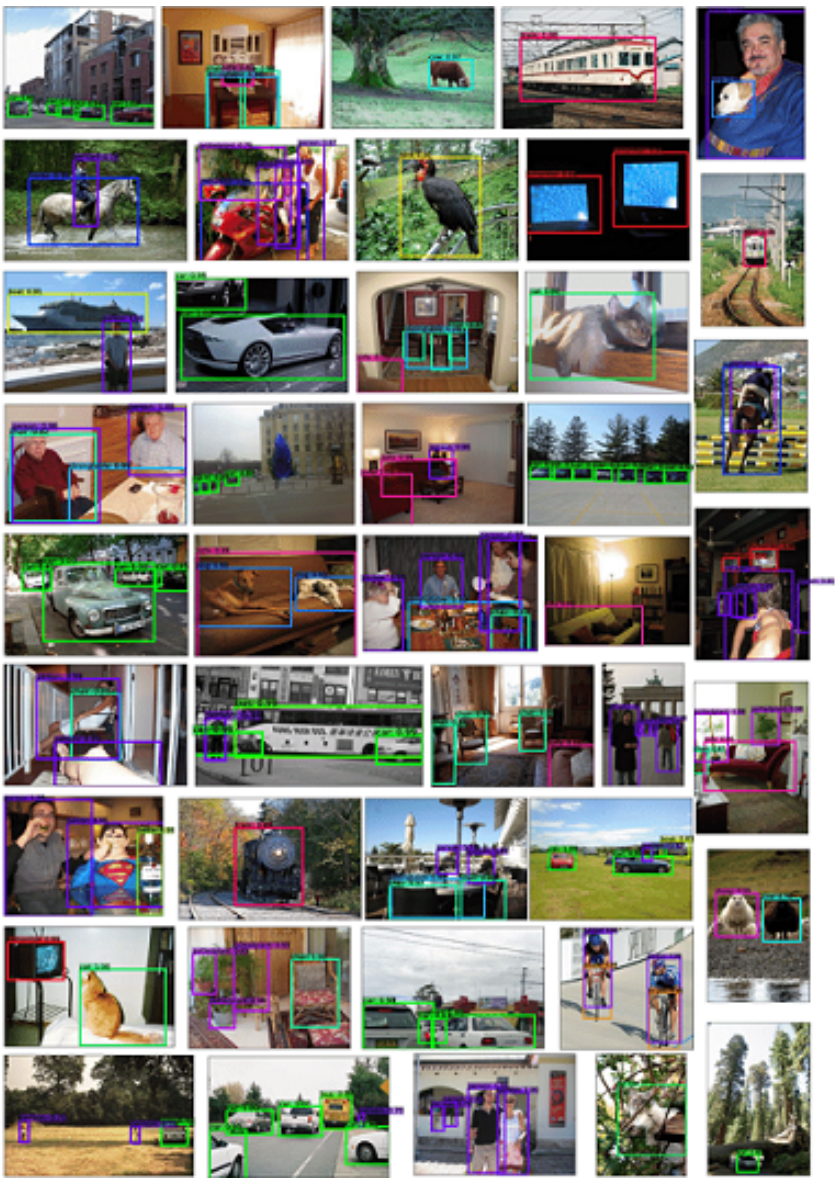

Figure 3.1: PASCAL VOC Dataset [12]

class labels. Specifically, the dataset covers twenty object categories, including common entities such as people, vehicles, animals, and household items.

One notable aspect of the PASCAL VOC 2012 dataset [12] is its rich annotations, which provide detailed information essential for training and evaluating object detection algorithms. These annotations not only include bounding boxes around objects but also assign class labels to each object instance, enabling precise classification. Additionally, the dataset offers segmentation masks, allowing for pixel-level delineation of object boundaries, which is crucial for tasks like semantic segmentation.

Moreover, the PASCAL VOC 2012 [12] dataset incorporates a diverse range of scenes and image compositions, ensuring the inclusion of various real-world scenarios. This diversity enhances the robustness and generalizability of models trained on this dataset, as they are exposed to a wide array of visual con-

texts and object configurations.

Researchers and practitioners often leverage the PASCAL VOC 2012 dataset [12] to develop, benchmark, and compare state-of-the-art object detection algorithms. Its widespread adoption and comprehensive annotations make it a valuable resource for advancing the field of computer vision, facilitating progress in tasks related to object detection and beyond.

3.2.2 MS COCO

The MS COCO dataset [28], or Microsoft Common Objects in Context, comprises an extensive collection of 328,000 images capturing everyday items and individuals. This dataset serves as a valuable resource for training machine learning models to detect, classify, and characterize objects, leveraging its rich annotations. Offering annotations for object labels, segmentation masks, and captions, MS COCO [28] is widely recognized as a benchmark dataset for tasks like object detection, segmentation, and captioning within the realm of computer vision. Its vast array of images covers a diverse spectrum of object categories and scenes encountered in real-life scenarios, making it an ideal choice for both training and assessing the performance of object detection algorithms.

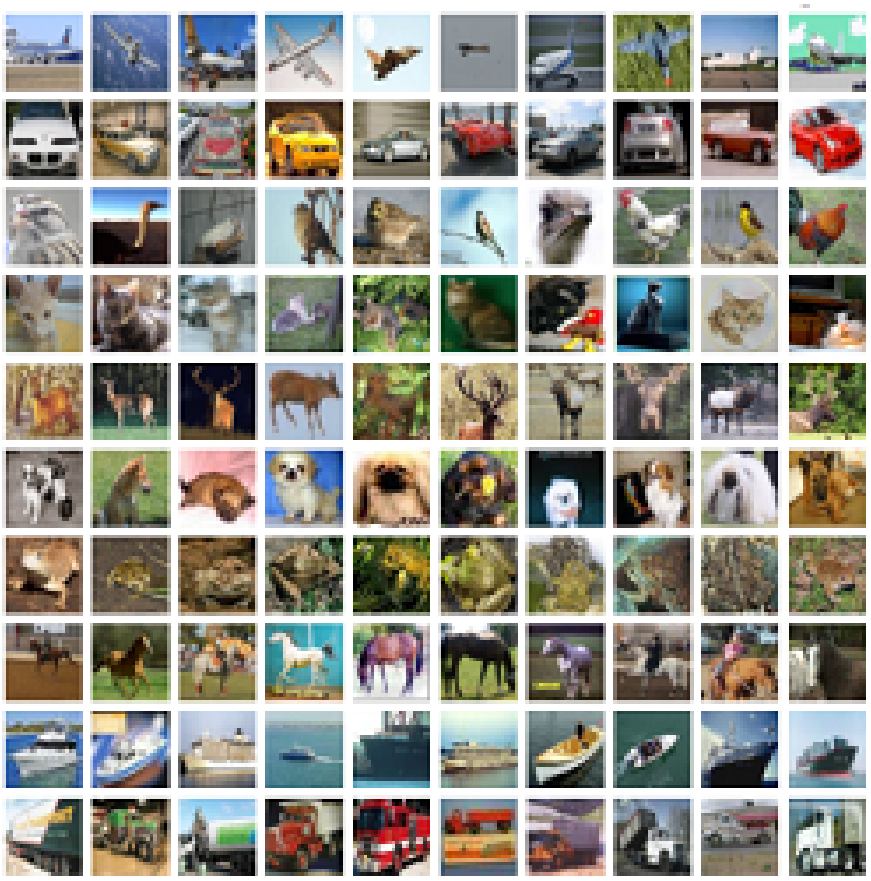

Figure 3.2: MS COCO Dataset [28]

The MS COCO dataset [28], short for Microsoft Common Objects in Context dataset, is a widely used benchmark dataset for object detection, segmentation, and captioning tasks in computer vision. It contains a large collection of images, each annotated with object labels, segmentation masks, and captions. The dataset is designed to cover a diverse range of object categories and scenes commonly encountered in everyday life, making it suitable for training and evaluating object detection algorithms.

Key features of the MS COCO dataset [28] for object detection include:

- **Large Scale:** The dataset consists of over 200,000 images, each containing multiple objects across 80 different categories, making it one of the largest datasets available for object detection.
- **Rich Annotations:** Each image in the dataset is annotated with bounding boxes around object instances, providing precise localization information for training object detection models.
- **Segmentation Masks:** In addition to bounding boxes, the dataset also includes pixel-level segmentation masks for object instances, enabling more detailed analysis and evaluation of segmentation-based approaches.
- **Diverse Object Categories:** The dataset covers a wide range of object categories, including common objects such as people, animals, vehicles, and household items, as well as more specific categories like sports equipment and electronic devices.
- **Complex Scenes:** Images in the dataset capture various real-world scenarios with multiple objects, occlusions, and cluttered backgrounds, challenging object detection algorithms to accurately identify and localize objects in complex scenes.
- **Standard Evaluation**: MS COCO [28] provides a standardized evaluation protocol for object detection algorithms, including metrics such as Average Precision (AP) and Average Recall (AR), allowing researchers to compare the performance of different models objectively.

Overall, the MS COCO [28] dataset serves as a valuable resource for advancing the state-of-the-art in object detection research, providing a comprehensive benchmark for training and evaluating algorithms on real-world, diverse imagery.

3.3 Implementation Details

3.3.1 YOLO (You Only Look Once)

The YOLO (You Only Look Once) algorithm represents a groundbreaking approach to object detection, capable of detecting objects in real-time with impressive accuracy. Here, we outline the implementation steps for various versions of YOLO in object detection tasks.

Preprocessing the Dataset

Before processing the dataset, itś essential to initialize certain constants. These constants typically include the fixed width and height to which the model will resize the input images, as well as any normalization factors needed to scale pixel values appropriately.

```
# Constants initialization
IMAGE_WIDTH = 416
IMAGE_HEIGHT = 416
```

Generating YOLO Model

The YOLO model architecture comprises convolutional layers followed by detection layers, which output bounding boxes and class predictions. Let's see how each version of YOLO differs in its implementation:

Loss Function: Categorical cross-entropy
Optimizer: Adam optimizer
Metrics: Accuracy
Epochs: The number of training epochs, usually a large number to allow for sufficient training time
Steps per Epoch: The number of steps (batches) to be processed in each epoch

YOLO v5 [50] (YOLO)

YOLO v5 [50] represented a departure from the traditional YOLO architecture, as it was developed by a different research group. It introduced a streamlined architecture and focused on simplicity, achieving competitive performance with significantly fewer parameters.

Training Hyperparameters for YOLOv5 [50]:

```
# Training hyperparameters for YOLO v5 loss_YOLO5 = "categorical_crossentropy"
optimizer_YOLO5 = "Adam"
metrics_YOLO5 = ["accuracy"]
epochs_YOLO5 = 5000
steps_per_epoch_YOLO5 = 50
.
```

Implementing object detection using YOLO v5 [50] involves several steps, including dataset preparation, model configuration, training, and evaluation. Let's delve into each of these aspects:

- **Dataset Preparation:**

- Data Collection: Gather a diverse dataset containing images annotated with bounding boxes around objects of interest. This dataset should represent the target objects and scenarios the model will encounter in real-world applications.

- Data Annotation: Annotate the dataset by marking bounding boxes around objects in the images. Each bounding box should specify the object class and its coordinates.

- Data Augmentation: Augment the dataset to increase its diversity and robustness. Techniques such as rotation, scaling, flipping, and color jittering can help expose the model to various object orientations, sizes, and lighting conditions.

- **Model Configuration:**

- Choose YOLO v5 [50] Variant: Select the YOLO v5 [50] variant (e.g., YOLOv5s, YOLOv5m, YOLOv5l, YOLOv5x) based on your computational resources and accuracy requirements. Larger variants offer higher accuracy but require more computational resources.

- Model Configuration: Configure the YOLO v5 [50] model architecture, including parameters such as input image size, anchor box settings, and number of classes to detect. This configuration ensures that the model is tailored to your specific object detection task.

- **Training:**

- Data Loading: Load the annotated dataset and preprocess the images and annotations as per the YOLOv5 [50] input requirements. This may involve resizing images, normalizing pixel values, and encoding bounding box annotations.

- Model Initialization: Initialize the YOLOv5 [33] model architecture with the chosen variant and configuration. This involves setting up the neural network layers, loss functions, and optimization algorithms.
- Training Procedure: Train the YOLOv5 [33] model on the annotated dataset using techniques such as stochastic gradient descent (SGD) or Adam optimization. During training, the model learns to predict bounding boxes and object classes while minimizing a chosen loss function (e.g., cross-entropy loss).
- **Hyperparameter Tuning:** Fine-tune hyperparameters such as learning rate, batch size, and regularization strength to optimize model performance and convergence speed.
- **Evaluation:**
- Validation Set: Split the annotated dataset into training and validation sets. Use the validation set to monitor the model's performance during training and adjust hyperparameters accordingly.
- Metrics Calculation: Evaluate the trained YOLO v5 [50] model on a separate test set using metrics such as precision, recall, and mean average precision (mAP). These metrics quantify the model's ability to accurately detect and localize objects in unseen images.
- **Visualization**: Visualize the model's predictions on sample images from the test set, overlaying predicted bounding boxes and class labels to assess its performance qualitatively.
- **Deployment:**
- Inference: Deploy the trained YOLO v5 [50] model for inference on new, unseen images or video streams. This involves feeding input images through the model and processing the output predictions (i.e., detected bounding boxes and class probabilities).
- Integration: Integrate the YOLOv5 [50] model into your application or system architecture, ensuring compatibility with the target deployment environment (e.g., edge devices, cloud servers).
- Performance Monitoring: Continuously monitor the model's performance in real-world scenarios, collecting feedback data to iteratively improve its accuracy and reliability.

  By following these steps, practitioners can effectively implement object detection using YOLO v5 [50] and deploy robust, accurate models for a variety of applications.

3.4 Results

Figure 3.3: Test images for different YOLO versions

The result represents the difference between YOLO versions. We can see that YOLOv5 [50] performs very well on object detection; also, YOLOv8 [39] try to work in different task like segmentation, pose estimation, etc, with the command line interface.

While the rest of the metrics were reported on COCO [28] dataset, the metrics for YOLO and YOLOv2 [20], YOLOv5, YOLOv8 [39] were reported on VOC20012 [12].

Version	Date	Anchor	Framework	Backbone	COCO AP (%)	VOC AP (%)
YOLO	2015	No	Darknet	Darknet24	63.4	-
YOLOV2	2016	Yes	Darknet	Darknet24	48.2	63.4
YOLOV3	2018	Yes	Darknet	Darknet53	36.2	-
YOLOV4	2020	Yes	Darknet	CSPDarknet53	43.5	-
YOLOv5	2020	Yes	Pytorch	Modified CSP v7	64.0	73.0
YOLOV6	2022	No	Pytorch	EfficientRep	52.5	-
YOLOV7	2022	No	Pytorch	RepConvN	56.8	-
YOLOV8	2023	No	Pytorch	YOLO v8	58	68.0

Table 3.1: Summary of YOLO versions

We can see a general trend of increasing object detection performance over time when comparing the various YOLO (You Only Look Once) versions. The official YOLO versions have increased Average Precision (AP) values on the COCO [28] and VOC datasets [12] from YOLOv1 [33] to YOLOv8 [39]. This illustrates how the YOLO algorithms are continuously improved for real-time object identification applications in order to increase accuracy and speed.

It is crucial to take into account the specific advantages of YOLO algorithms in terms of real-time object recognition capabilities when comparing them to other state-of-the-art (SOTA) object detection techniques, such as Faster R-CNN [14], RetinaNet [27], and EfficientDet [48]. YOLO algorithms efficiently

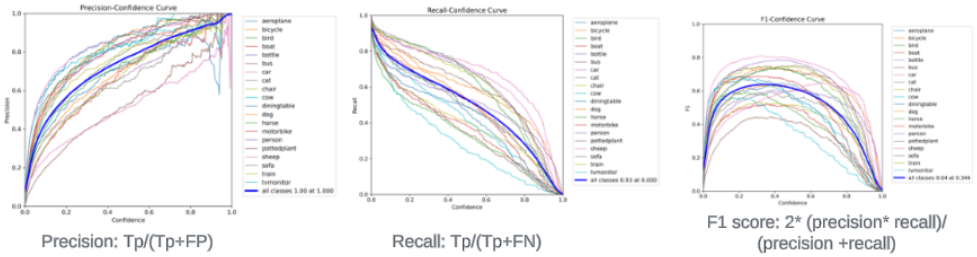

Figure 3.4: Performance Measure

balance accuracy and speed, making them particularly ideal for applications where real-time performance is critical, even if their AP values might not necessarily be higher than those of other SOTA approaches.

Here is a brief summary of each version compared to its predecessor:

YOLO Version	Backbone	Key Features
YOLO	Darknet24	First real-time object detection model, unified detection framework [33].
YOLOv2	Darknet24	Introduced anchor boxes, batch normalization, improved localization and performance [20].
YOLOv3	Darknet53	Added multi-scale predictions, improved feature extractor, better speed-accuracy trade-off [38].
YOLOv4	CSPDarknet53	Implemented CSPNet backbone, PANet, mish activation, cross mini-batch normalization [2].
YOLOv5	Modified CSP v7	PyTorch implementation, automated anchor box calculation, smaller and faster model [50].
YOLOv6	EfficientRep	High efficiency, designed for real-time performance on edge devices [25].
YOLOv7	RepConvN	Introduced E-ELAN design, novel scaling techniques, better handling of small objects [52].
YOLOv8	CSPDarknetBackbone	Enhanced architecture, faster NMS, mosaic augmentation [39].

Table 3.2: Summary of YOLO Versions and Their Key Features

3.4.1 Tradeoff between speed and accuracy

In order to predict item places and classes immediately from the input image, the original YOLO model prioritized high-speed object recognition using a single CNN. This focus on speed, meanwhile, resulted in decreased accuracy, particularly for small objects or overlapping bounding boxes.

Later YOLO versions fixed these issues by improving the foundation while retaining real-time functionality. In order to enhance object localisation and increase accuracy, YOLOv2 [20] included anchor boxes and passthrough layers. Through multi-scale feature extraction, YOLOv3 [38] considerably improved performance and made it possible for better object recognition at various sizes. The speed-accuracy tradeoff got more complex as the YOLO system devel-

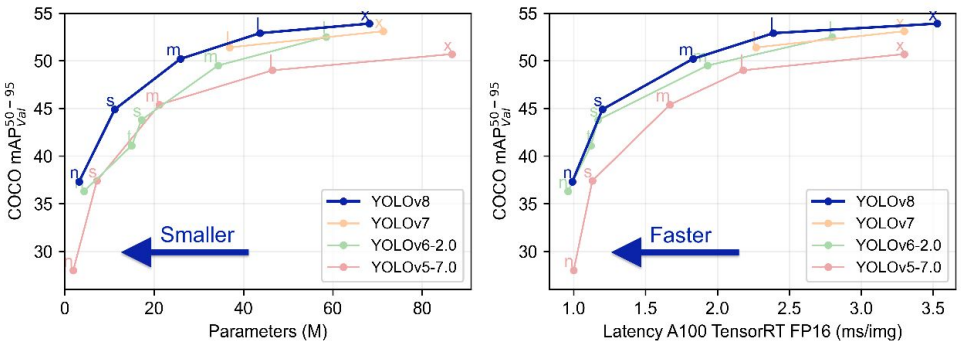

Figure 3.5: YOLOv5 to YOLOv8 [39] Tradeoff [51]

oped. Innovations include new network backbones, cutting-edge data augmentation methods, and optimized training methodologies were integrated in YOLOv4 [2] and YOLO v5 [50], resulting in considerable accuracy advances without compromising real-time performance.

Since Scaled YOLOv4 [2] was released, all official YOLO models have improved the speed-accuracy tradeoff by providing several model sizes; Fig. 6 shows the many YOLO models and their use cases, including big, medium, and tiny, each of which is tailored to a particular application's hardware needs. These updated versions provide lightweight variants that are designed for edge devices, sacrificing accuracy for simplified computing and quicker processing.

CHAPTER 4
Person Re- identifiaction

In this section, we present our proposed approach for person re-identification, outlining the various blocks involved in our methodology. Firstly, we divide the input video into individual images to facilitate processing. Next, we employ a person detection and tracking model to identify and track individuals across frames. Subsequently, we extract features using the OMNI feature extraction method [60], followed by further feature extraction using FACENET [42]. Finally, we perform person re-identification using the extracted features. We discuss the datasets used for evaluation, including existing datasets and proposed video datasets. Furthermore, we provide details on the implementation of our approach and present the results obtained from our experiments, demonstrating the effectiveness of our proposed methodology.

4.1 Proposed Approach

Our proposed methodology for Person Re-identification involves several essential steps, each contributing to accurate individual identification and tracking. Initially, person detection within frames is achieved using the YOLO v5 [50] object detection model [33], followed by tracking individuals using SORT [55] with Deep Association Metric Learning and models like ResNet [17], VGG [29], and Inception [47].

Next, a threshold is established to determine the preferred feature extraction approach, which includes facial features detected by the FaceNet model [42] or Omni feature learning [60]. These extracted features are then utilized for person re-identification.

At the core of our model lies the Omni Scale [60] and FaceNet [42] approach, integrating cutting-edge computer vision and deep learning technologies. Utilizing YOLO v5 [50] for pedestrian detection and SORT with Deep Association Metric for tracking [55], our model addresses challenges such as dynamic camera perspectives and occlusion.

Architecturally, our model employs ResNet with multiple blocks and an innovative aggregation gate, dynamically fusing multi-scale feature maps for

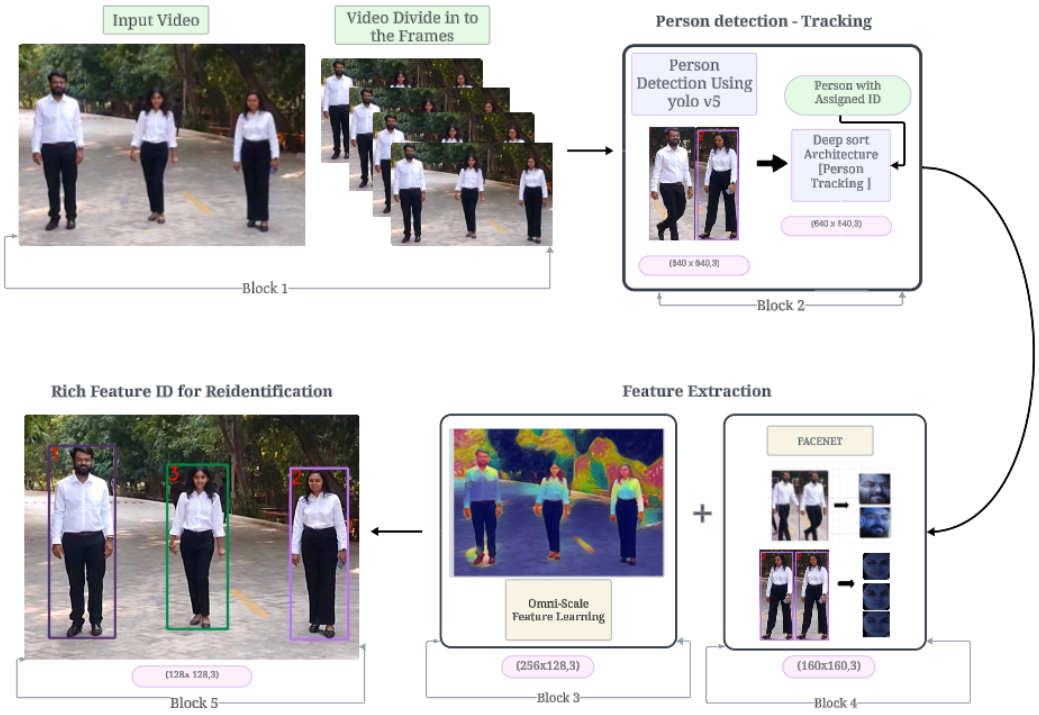

Figure 4.1: Proposed Method

effective feature extraction. This choice enables superior performance in person re-identification, even in challenging scenarios.

A key aspect of our approach is the division of videos into frames, processed through YOLO v5 [50] for bounding box annotation. These boxes are then tracked using SORT with Deep Association Metric [55], followed by feature extraction using various models like ResNet, DenseNet [19], ShuffleNet [57], MLFN [6], and Mudeep [36]. Omni-scale feature learning [60] combines features such as clothing attributes to establish identity, further refined by the FaceNet [42] model.

These stages of our proposed approach are elaborated through five distinct blocks, as depicted in Figure 4.1.

4.1.1 Block 1: Video divided into Images

As shown in Fig. 4.2 ,To process the provided input video, it needs to be partitioned into frames, achieving a rate of 60 frames per second. This can be accomplished using OpenCV or other computer vision techniques.

4.1.2 Block 2: Person Detection and Tracking model

In the second stage, person detection takes center stage. ,Once persons are identified within each frame, the process advances to tracking.

YOLO v5 [50] for Person Detection: YOLO v5 [50] [33] stands out among

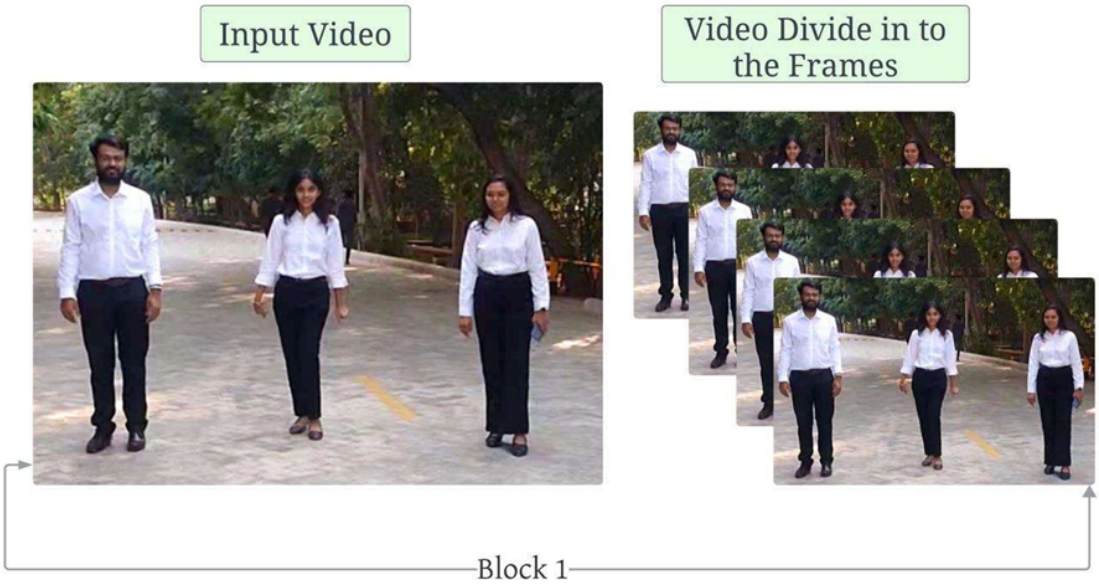

Figure 4.2: Converting Video to frames

YOLO versions due to key advancements:
Architecture: YOLO v5 [50] employs the intricate EfficientDet [48] architecture, derived from EfficientNet, enabling greater accuracy and generalization, especially for persons [33].
Dynamic Anchor Boxes: YOLO v5 [50] introduces dynamic anchor boxes generated by clustering ground truth boxes, aligning them more closely with detected objects' size and shape [33].
Spatial Pyramid Pooling (SPP): Incorporating SPP, YOLO v5 [50] reduces feature map spatial resolution, improving detection for small objects across multiple scales [33].
CIoU Loss: YOLO v5 [50] introduces CIoU loss, enhancing model performance on imbalanced datasets common in person detection tasks [33].
Training Procedure: YOLOv5 employs advanced data augmentation and training procedures, contributing to its overall performance [33].
Ease of Use: Designed to be user-friendly, YOLO v5 [50] offers easy installation, fast training, and intuitive data file system structure [33]. While YOLO v5 [50] shows improved accuracy, YOLOv3 [38] remains a strong contender due to its balanced performance [3].
SORT with Deep Association Metric Algorithm for Person Tracking [55]: Tracking is executed using the robust SORT with Deep Association Metric [55] backbone architecture [5], offering flexibility with options like ResNet, VGG, and Inception [47].
Detection: Utilizing YOLO v5 [50] for object identification and tracking in images or videos. Difference Detector: Identifying changes between consecutive frames, useful in motion detection or triggering further analysis for significant changes.

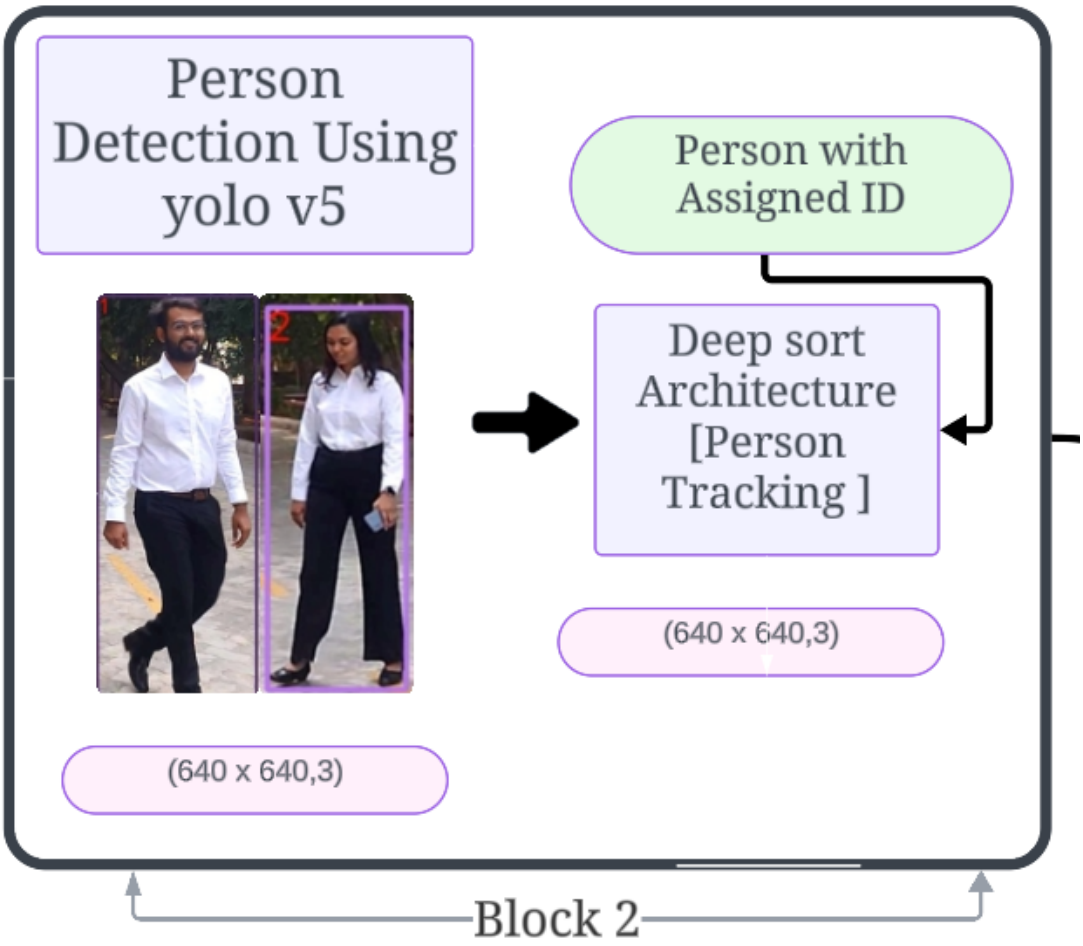

Figure 4.3: Person Detection and Tracking model

4.1.3 Block 3: Feature Extraction with OMNI feature

we employ the Omni-Scale Feature Learning method [60] to extract discriminative features. This technique enhances feature representation across various semantic levels, utilizing a network architecture with concatenated channel blocks and an aggregation gate for dynamic fusion of multi-scale feature maps. This ensures effective feature extraction, even in challenging scenarios like occlusion and varying appearances.

Next, we focus on local/global features extracted through the Omni-Scale method [60], particularly emphasizing dressing style. However, a limitation arises when dealing with individuals wearing the same clothing.

Figure 4.4: Snaps of Person Detection and Tracking

4.1.4 Block 4: Feature Extraction using FACENET

To overcome further processing through Omni-Scale Feature Learning method's [60] limitation, we incorporate facial features extracted by the FaceNet model . Renowned for its robust face recognition and feature embedding capabilities, FaceNet [42] utilizes a triplet loss function to generate condensed representations of faces, enabling precise face verification and clustering. This integration significantly improves the accuracy of person re-identification compared to alternative feature extraction methods.

4.1.5 Block 5: Person Re-Identification

The final block involves the classification and assignment of re-identity to each individual. This is accomplished using a variety of techniques, including Resnet, VGG, OSNET [60], or the enriched features obtained through Omni-Scale Feature Learning [60]. The output from this stage provides a comprehensive understanding of the identified individuals, ensuring accurate re-identification across frames.

Figure 4.5: Feature Extraction using Local, Global and Face Features

4.2 Dataset

4.2.1 Existing Datasets:

MSMT17 [54]: A multi-scene multi-time dataset with 180 hours of videos captured by 12 outdoor and 3 indoor cameras, featuring 4,101 identities and 126,441 bounding boxes.

Market-1501 [59] [45]: A large-scale benchmark dataset with 1501 identities and 32,668 pedestrian image bounding-boxes, divided into training and testing sets.

PRID 2011 [16] [35]: A dataset providing multiple person trajectories recorded from static monitoring crosswalks and sidewalks, with clean backgrounds and minimal occlusion.

LIDS-VID [45]: Involves 300 different pedestrians observed across two cameras.

DukeMTMC-reID [45] [40]: A subset of the DukeMTMC dataset, consisting of 16,522 training images, 2,228 query images, and 17,661 gallery images.

MARS [59] [45]: An extension of the Market-1501 dataset, capturing 1,261 pedestrians by at least 2 cameras.

SenseReID [58]: A dataset for evaluating ReID models captured from real surveillance cameras.

COCO [28]: A large-scale image recognition dataset with 328K images, 80 object categories, and 91 stuff categories, aiding in object detection, segmentation, and captioning tasks.

VOC 2007: [12] A collection of images used for object recognition tasks, with

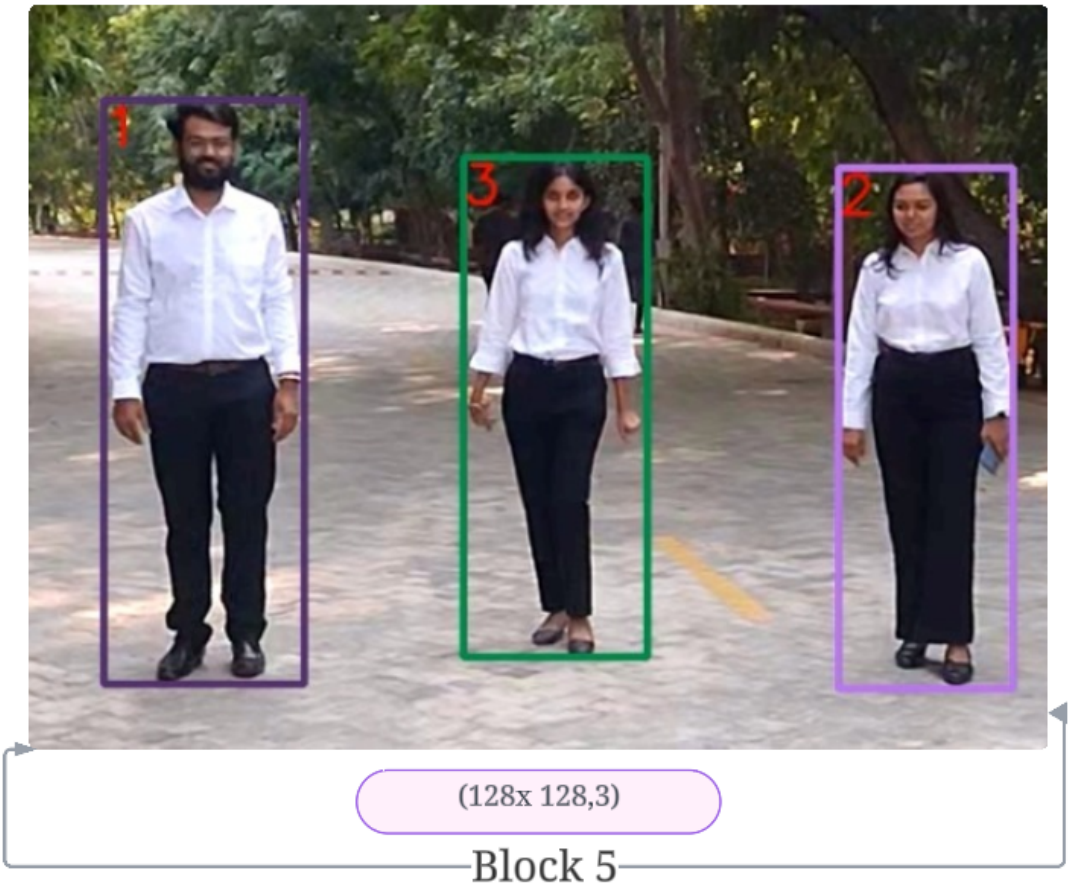

Figure 4.6: Result of proposed approch for Person Re-identification

9,963 images across 20 different classes.

4.2.2 Proposed Video Datasets:

a. Dynamic Environment Change with Different clothes (DECDC):

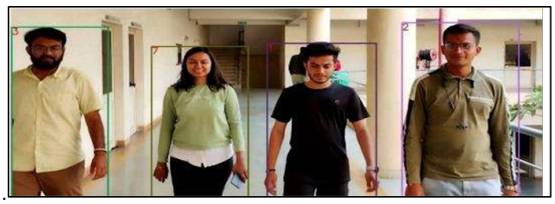

Figure 4.7: Dynamic Environment Change with Different clothes Dataset

Curated to reflect diverse environments and lighting conditions, DECDC comprises three videos featuring indoor and outdoor settings. Each video introduces challenges like changing directions and additional individuals to disrupt person re-identification.

Figure 4.8: Dynamic Environment Change with Same clothes Dataset

b. Dynamic Environment Change with Same clothes (DECSC): This dataset presents three dynamic environments with varying lighting conditions, each featuring three individuals dressed identically. Challenges include background complexity and intentional distractions, making it a valuable resource for refining person re-identification algorithms in real-world scenarios.

4.3 Implementation

Our computational journey in computer vision began with discovering YOLO v5 [50] on GitHub as our primary model. We customized it to suit our needs and integrated additional models such as FaceNet [42] and OmniScale [60]. Harnessing Python's capabilities and leveraging frameworks like TensorFlow, Keras, and PyTorch, alongside essential libraries like NumPy and OpenCV, we implemented our customized models. This approach allowed us to adapt the models for various tasks, including image processing and diverse feature extraction. Furthermore, we utilized Google Colab's computational power and ISRL Lab's high-performance GPU infrastructure to execute our implementations effectively. This comprehensive approach underscores our commitment to employing cutting-edge methodologies in advancing computer vision solutions.

4.4 Result

Result of Proposed Model on DECSC Dataset with Same clothes images:

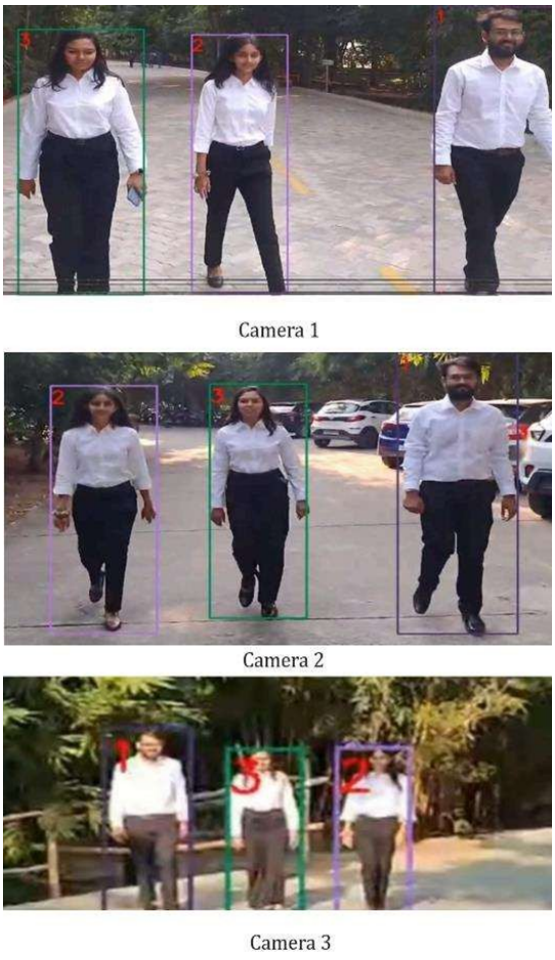

Figure 4.9: Result Snaps on DECSC dataset with Same clothes images

Table 4.1: Result Snaps on DECSC dataset with Same clothes images

Videos	Video Frames	Accuracy (%)
camera 1	840	100
camera 2	780	100
camera 3	660	72.72
Total	2280	92.10

In DECSC, all the 3 videos are setup at 3 different outdoor environments in different light conditions, where almost we achieve 92.10% accuracy in same style cloth condition.

Result of Proposed Model on DECDC Dataset with Diffrent clothes images:

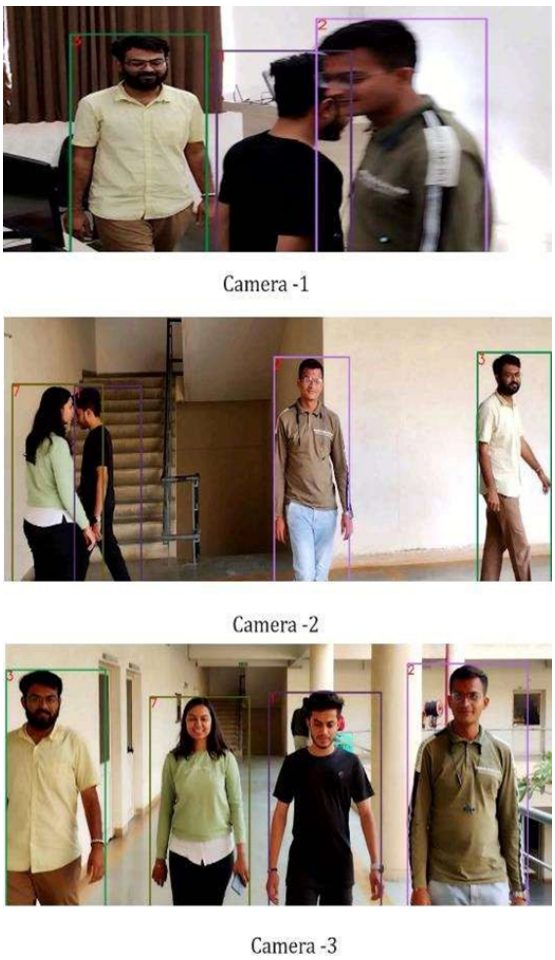

Figure 4.10: Result Snaps on DECDC dataset with Same clothes images

Table 4.2: Result with Diffrent clothes images on DECDC dataset

Videos	**Video Frames**	**Accuracy (%)**
camera 1	840	85.71
camera 2	1140	94.73
camera 3	1080	100
Total	3060	94.11

In DECDC, all the 3 videos are of different environments, one is indoor and two is outdoor environment with different light conditions, where almost we achieve 94.11% accuracy in different cloth condition.

CHAPTER 5
Conclusion

In conclusion, YOLOv8 and YOLO v5 [51] demonstrate competitive performance in real-time object recognition, balancing accuracy and speed effectively. While their AP values may not always surpass those of other techniques, their quick and accurate object identification remains unmatched. Future YOLO iterations are expected to enhance both accuracy and speed, pushing the boundaries of real-time object identification.Transitioning to person re-identification, it represents a dynamic and promising field with significant potential for enhancing public safety and security in various real-world scenarios. Our approach integrates advanced neural network architectures with object detection and tracking algorithms to form a robust foundation. By breaking down input videos into frames and employing YOLO v5 [50] for person detection, we ensure accurate identification in each frame. The Sort [55] tracking algorithm maintains consistent tracking across frames, while a dual-feature approach captures both facial and body features, enhancing accuracy and reliability.

Our comprehensive pipeline seamlessly integrates object detection, tracking, and feature extraction, resulting in a powerful person re-identification system capable of linking individuals across diverse camera views and scenarios. Validation on proposed datasets, DECDC and DECSC, demonstrates superior performance, with an accuracy of 92.10% in DECSC and 94.11% in DECDC. These results underscore the potential applications of our person re-identification model in real-world scenarios, further advancing public safety and security.

CHAPTER 6
Future work

Future work for this project could focus on several aspects to further enhance its capabilities and address potential limitations: **Integration of Advanced Features:** Explore the integration of additional features beyond facial and body features, such as gait analysis or clothing attributes, to improve the robustness of person re-identification across different scenarios and lighting conditions.

Enhancement of Tracking Algorithms: Investigate advanced tracking algorithms or multi-object tracking frameworks to improve the consistency and accuracy of person tracking across frames, especially in scenarios with occlusions or crowded environments.

Fine-tuning YOLO Models: Continuously monitor and incorporate updates or improvements to YOLO models (e.g., YOLOv9) to further optimize the balance between accuracy and speed in object detection, ensuring the system remains at the forefront of real-time object recognition.

Data Augmentation and Generalization: Augment existing datasets or collect additional data from diverse environments to improve the generalization capability of the person re-identification model. This can help ensure reliable performance in real-world scenarios with varying camera viewpoints, lighting conditions, and occlusion levels.

Deployment and Integration: Explore the deployment of the person re-identification system in real-world surveillance or security applications, considering factors such as hardware constraints, network latency, and scalability. Integration with existing security systems or platforms could also be explored to enhance overall efficiency and effectiveness.

Ethical Considerations: Conduct an ethical evaluation of the deployed system to ensure privacy protection and mitigate potential biases or discriminatory outcomes. This may involve implementing privacy-preserving tech-

niques, developing transparent and accountable decision-making processes, and conducting regular audits or evaluations of the system's performance and impact. By addressing these areas in future research and development efforts, the project can continue to advance the field of person re-identification and contribute to enhancing public safety and security in various real-world scenarios.

References

[1] F. Aksu and C. Direkoğlu. Person re-identification in surveillance videos using deep learning based body part partition and gaussian filtering. *European Journal of Science and Technology*, 2(November):291–296, 2020.

[2] A. Bochkovskiy, C.-Y. Wang, and H.-Y. M. Liao. Yolov4: Optimal speed and accuracy of object detection. *arXiv preprint*, arXiv:2004.10934, 2020.

[3] N. Buhl. YOLO models for object detection explained [YOLOv8 updated], 2023. Accessed Nov. 12, 2023.

[4] Z. Cai and N. Vasconcelos. Cascade r-cnn: Delving into high quality object detection. *arXiv preprint arXiv:1712.00726*, 2017.

[5] N. Carion, F. Massa, G. Synnaeve, N. Usunier, A. Kirillov, and S. Zagoruyko. End-to-end object detection with transformers. *arXiv preprint arXiv:2005.12872*, 2020.

[6] X. Chang, T. M. Hospedales, and T. Xiang. Multi-level factorisation net for person re-identification. In *2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 2109–2118, Salt Lake City, UT, USA, 2018.

[7] L.-C. Chen, G. Papandreou, F. Schroff, and H. Adam. Rethinking atrous convolution for semantic image segmentation. *arXiv preprint arXiv:1706.05587*, 2017.

[8] G. Coleman and H. Andrews. Image segmentation by clustering. *Proceedings of the IEEE*, 67(5):773–785, May 1979.

[9] Y. Dai, J. Tao, C. Ouyang, and X. Wang. Clothing recognition based on improved ResNet18 model. In *2022 IEEE 5th Advanced Information Management, Communicates, Electronic and Automation Control Conference (IMCEC)*, volume 5, pages 1297–1301, 2022.

[10] K. Duan, S. Bai, L. Xie, H. Qi, Q. Huang, and Q. Tian. Centernet: Keypoint triplets for object detection. *arXiv preprint arXiv:1904.08189*, 2019.

[11] M. Edman. Segmentation using a region growing algorithm. Release 0.00, October 2007. Available via license: CC BY 4.0. Content may be subject to copyright.

[12] M. Everingham, L. Van Gool, C. K. Williams, J. Winn, and A. Zisserman. The pascal visual object classes (voc) challenge. *International Journal of Computer Vision (IJCV)*, 88(2):303–338, 2010.

[13] P. F. Felzenszwalb and D. P. Huttenlocher. Efficient graph-based image segmentation. *Artificial Intelligence Lab, Massachusetts Institute of Technology*, year.

[14] R. Girshick. Fast R-CNN. In *2015 IEEE International Conference on Computer Vision (ICCV)*, pages 1440–1448. IEEE, 2015.

[15] K. He, G. Gkioxari, P. Dollar, and R. Girshick. Mask r-cnn. *arXiv preprint arXiv:1703.06870*, 2017.

[16] K. He, G. Gkioxari, P. Dollár, and R. Girshick. Mask R-CNN. *IEEE Trans. Pattern Anal Mach. Intell.*, 42(2):386–397, 2020.

[17] K. He, X. Zhang, S. Ren, and J. Sun. Deep residual learning for image recognition. In *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 770–778, Las Vegas, NV, USA, 2016.

[18] J. Hu, L. Shen, and G. Sun. Squeeze-and-excitation networks. In *2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 7132–7141, Salt Lake City, UT, USA, 2018.

[19] G. Huang, Z. Liu, L. V. D. Maaten, and K. Q. Weinberger. Densely connected convolutional networks. In *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 2261–2269, Honolulu, HI, USA, 2017.

[20] X. Huang, X. Wang, W. Lv, X. Bai, X. Long, K. Deng, Q. Dang, S. Han, Q. Liu, X. Hu, D. Yu, Y. Ma, and O. Yoshie. Pp-yolov2: A practical object detector. *arXiv preprint arXiv:2104.10419*, Apr 2021.

[21] F. N. Iandola, S. Han, M. W. Moskewicz, K. Ashraf, W. J. Dally, and K. Keutzer. Squeezenet: Alexnet-level accuracy with 50x fewer parameters and <0.5mb model size, 2016. Submitted on 24 Feb 2016 (v1), last revised 4 Nov 2016 (this version, v4).

[22] K. Islam. Deep learning for video-based person re-identification: A survey. *arXiv preprint arXiv:2303.11332*, 2023.

[23] G. Jocher, A. Chaurasia, and J. Qiu. YOLO by Ultralytics. `https://github.com/ultralytics/ultralytics`, 2023. Accessed: May 12, 2023.

[24] N. Klingler. Deep learning for person re-identification. viso.ai, 2023. Accessed Apr. 22, 2023.

[25] C. Li, L. Li, H. Jiang, K. Weng, Y. Geng, Z. Ke, Q. Li, M.-M. Cheng, W. Nie, et al. Yolov6: A single-stage object detection framework for industrial applications. *arXiv preprint*, arXiv:2209.02976, 2022.

[26] W. Li, X. Zhu, and S. Gong. Harmonious attention network for person re-identification. In *2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 2285–2294, Salt Lake City, UT, USA, 2018.

[27] T.-Y. Lin, P. Goyal, R. B. Girshick, K. He, and P. Dollár. Focal loss for dense object detection. *arXiv preprint arXiv:1708.02002*, 2017.

[28] T.-Y. Lin, M. Maire, S. Belongie, L. Bourdev, R. Girshick, J. Hays, P. Perona, D. Ramanan, C. L. Zitnick, and P. Dollár. Microsoft COCO: Common Objects in Context. *arXiv preprint arXiv:1405.0312*, 2014.

[29] S. Liu and W. Deng. Very deep convolutional neural network based image classification using small training sample size. In *Proceedings 3rd IAPR Asian Conference on Pattern Recognition, ACPR 2015*, 2016.

[30] W. Liu, D. Anguelov, D. Erhan, C. Szegedy, S. Reed, C.-Y. Fu, and A. C. Berg. SSD: single shot multibox detector. *arXiv preprint arXiv:1512.02325*, 2016.

[31] J. Long, E. Shelhamer, and T. Darrell. Fully convolutional networks for semantic segmentation. *arXiv preprint arXiv:1411.4038*, 2015.

[32] S. Minaee, Y. Boykov, F. Porikli, A. Plaza, N. Kehtarnavaz, and D. Terzopoulos. Image segmentation using deep learning: A survey. *arXiv preprint arXiv:2001.05566*, 2020.

[33] A. Mukherjee. Yolo: Algorithm for object detection explained, 2023. LinkedIn.

[34] S. J. Narayanan. Deep learning: Algorithms and applications in surveillance and identification. In Pedrycz and S.-M. Chen, editors, *Deep Learning: Algorithms and Applications*, pages 263–297. Springer International Publishing, Cham, 2020.

[35] P. Pathak, A. E. Eshratifar, and M. Gormish. Video person re-id: Fantastic techniques and where to find them. *arXiv preprint arXiv:1912.05295*, Nov 2019.

[36] X. Qian, Y. Fu, Y.-G. Jiang, T. Xiang, and X. Xue. Multi-scale deep learning architectures for person re-identification. In *2017 IEEE International Conference on Computer Vision (ICCV)*, pages 5409–5418, Venice, Italy, 2017.

[37] J. Redmon, S. Divvala, R. Girshick, and A. Farhadi. You only look once: Unified, real-time object detection. *arXiv preprint arXiv:1506.02640*, 2016. Available: `https://doi.org/10.48550/arXiv.1506.02640`.

[38] J. Redmon and A. Farhadi. Yolov3: An incremental improvement. *arXiv preprint*, arXiv:1804.02767, 2018.

[39] D. Reis, J. Kupec, J. Hong, and A. Daoudi. Real-time flying object detection with yolov8. *arXiv preprint arXiv:2305.09972*, May 2023.

[40] E. Ristani, F. Solera, R. Zou, R. Cucchiara, and C. Tomasi. Performance measures and a data set for multi-target, multi-camera tracking. In *European Conference on Computer Vision*, pages 17–35. Springer, 2016.

[41] O. Ronneberger, P. Fischer, and T. Brox. U-Net: Convolutional networks for biomedical image segmentation. *arXiv preprint arXiv:1505.04597*, 2015.

[42] F. Schroff, D. Kalenichenko, and J. Philbin. Facenet: A unified embedding for face recognition and clustering. pages 815–823, 2015.

[43] N. Senthilkumaran and R. Rajesh. Edge detection techniques for image segmentation – a survey of soft computing approaches. *International Journal of Recent Trends in Engineering*, 1(2):250, May 2009.

[44] S. Seo and J. Kim. Efficient weights quantization of convolutional neural networks using kernel density estimation based non-uniform quantizer. *Applied Sciences*, 9(12), 2019.

[45] N. K. Singh, M. Khare, and H. B. Jethva. A comprehensive survey on person identification approaches: various aspects. *Multimed. Tools Appl.*, 81(11):15747–15791, 2022.

[46] Y. Sun, L. Zheng, Y. Yang, Q. Tian, and S. Wang. Beyond part models: Person retrieval with refined part pooling (and a strong convolutional baseline). In V. Ferrari, M. Hebert, C. Sminchisescu, and Y. Weiss, editors, *Computer Vision – ECCV 2018*, volume 11208 of *Lecture Notes in Computer Science*. Springer, 2018.

[47] C. Szegedy, S. Ioffe, V. Vanhoucke, and A. Alemi. Inception-v4, inception-resnet and the impact of residual connections on learning, 2016. Submitted on 23 Feb 2016 (v1), last revised 23 Aug 2016 (this version, v2).

[48] M. Tan, R. Pang, and Q. V. Le. Efficientdet: Scalable and efficient object detection. *arXiv preprint arXiv:1911.09070*, 2020.

[49] J. R. Terven and D. M. Esparza. A comprehensive review of yolo: From yolov1 to yolov8 and beyond. *arXiv preprint arXiv:2304.00501*, 2023.

[50] Ultralytics. YOLOv5: You Only Look Once v5. `https://github.com/ultralytics/yolov5`, 2020.

[51] Ultralytics. Yolo5 to yolo8 trade off. 2023.

[52] C.-Y. Wang, A. Bochkovskiy, and H.-Y. M. Liao. Yolov7: Trainable bag-of-freebies sets new state-of-the-art for real-time object detectors. *arXiv preprint*, arXiv:2207.02696, 2022.

[53] S. Wang and J. Fan. Image thresholding method based on tsallis entropy correlation. *Multimedia Tools and Applications*, 1:1–10, 2024.

[54] L. Wei, S. Zhang, W. Gao, and Q. Tian. Person transfer gan to bridge domain gap for person re-identification. In *Proceedings of the IEEE Conference on Computer Vision*, 2018.

[55] N. Wojke, A. Bewley, and D. Paulus. Simple online and realtime tracking with a deep association metric. pages 3645–3649, 2017.

[56] W. Zhang and D. Jiang. The marker-based watershed segmentation algorithm of ore image. *IEEE Journal*, year.

[57] X. Zhang, X. Zhou, M. Lin, and J. Sun. Shufflenet: An extremely efficient convolutional neural network for mobile devices. In *2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 6848–6856, Salt Lake City, UT, USA, 2018.

[58] H. Zhao et al. Spindle net: Person re-identification with human body region guided feature decomposition and fusion. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, volume 2017-Janua, pages 907–915, 2017.

[59] L. Zheng, L. Shen, L. Tian, S. Wang, J. Wang, and Q. Tian. Scalable person re-identification: A benchmark. In *Proceedings of the IEEE international conference on computer vision*, pages 1116–1124, 2015.

[60] K. Zhou, Y. Yang, A. Cavallaro, and T. Xiang. Omni-scale feature learning for person re-identification. In *2019 IEEE/CVF International Conference on Computer Vision (ICCV)*, pages 3701–3711, Seoul, Korea (South), 2019.

[61] B. Zoph, V. Vasudevan, J. Shlens, and Q. V. Le. Learning transferable architectures for scalable image recognition. In *2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 8697–8710, Salt Lake City, UT, USA, 2018.