Advancing Human Tracking in Multi Object Tracking Using Video Features and Depth Cues

V.Vignagajan
2024

Advancing Human Tracking in Multi Object Tracking Using Video Features and Depth Cues

Vignagajan
Index No: 19001746

Supervisor: Dr.K.D.Sandaruwan
Co-Supervisor: Dr.P.V.K.G.Gunawardana

May 2024

Submitted in partial fulfillment of the requirements of the
B.Sc. (Honours) in Computer Science Final Year Project

Declaration

I certify that this dissertation does not incorporate, without acknowledgement, any material previously submitted for a degree or diploma in any university and to the best of my knowledge and belief, it does not contain any material previously published or written by another person or myself except where due reference is made in the text. I also hereby give consent for my dissertation, if accepted, be made available for photocopying and for interlibrary loans, and for the title and abstract to be made available to outside organizations.

Candidate Name:
V.Vinothini

.....

SignatureofCandidate Date: 30.09.2024

This is to certify that this dissertation is based on the work of

V.Vignagajan

under my supervision. The thesis has been prepared according to the format stipulated and is of acceptable standard.

Principle Supervisor's Name: Dr.K.D.Sandaruwan

.....

Signature of Supervisor: Date: 30.09.2024

Co-Supervisor's Name: Dr.P.V.K.G.Gunawardana

.....

Signature of Co-Supervisor: Date: 30 - 09 - 2024

Abstract

The identification of each objects uniquely in each video frame is known as the Multi Object Tracking(MOT). MOT is well- known for its potential uses in autonomous driving, human-robot interaction, and surveillance. The goal of MOT is to correctly identify each item and to maintain that identification over time in the face of occlusions, changes in appearance, and other obstacles. This range of uses demonstrates the effectiveness of MOT and encourages scientists to address knowledge gaps in MOT fields. This has greatly influenced the development of all these large-scale MOT works.

The MOT has certain limitations. We try ton resolve this limitation using novel video features and depth cues integration. We explored freely available cross domain models such as human action recognition model and zero-shot depth extracting model. With that we explored use of depth cues to give additional information to improve tracking.

We analyzed the performance of our approaches by using benchmark datasets and standard metrics. These outcome of our research is , we can use the existing cross domain models to improve MOT in the aspect of detection without further fine-tuning or complex tricks. In the other hand, usage of depth cues improves MOT performance in short videos by fusion with the existing appearance features.

The major contribution of our research is, creating a unexplored path in solving MOT problem without complex algorithms and resource intensive training by utilizing existing cross domain knowledge.

Preface

This document has been written for the partial fulfillment of the requirements of the B.Sc. in Computer Science (Hons) Final Year Project in Computer Science (SCS4124). The problem of detecting the objects in each video frame is known as the MOT problem, and it is a prominent area in computer vision. MOT is well-known for its potential uses in autonomous driving, human-robot interaction, and surveillance. The goal of MOT is to correctly identify each item and to maintain that identification over time in the face of occlusions, changes in appearance, and other obstacles. This range of uses demonstrates the effectiveness of MOT and encourages scientists to address knowledge gaps in MOT fields. This has greatly influenced the development of all these large-scale MOT works. The MOT sector is still in its infancy and has certain drawbacks, though.

Next to address these problems, we explored addition methods of utilizing video features, using freely available cross domain models. With that we explored use of depth cues to give additional information to improve tracking. We validate the performance of our approaches by using benchmark datasets and standard metrics.

With constant guidance and supervision of my supervisor more conclusions were drawn about MOT which, we believe are new contributions to the body of knowledge.

Acknowledgement

First, I would like to give my sincere gratitude towards my university, University of Colombo School of Computing (UCSC) for giving me this great opportunity to carry out an individual research in which I could developed my research and other academic skills. My research is not possible without my supervisor Prof Dr.K.D Sandaruwan, lecturer of University of Colombo School of Computing and my co-supervisor Dr. P.V.K.G Gunawardana, senior lecturer of University of Colombo School of Computing, for providing me their valuable guidance and supervision throughout this research project. I am really grateful to help me in all the time, from beginning to the end of this research project with fruitful discussions. I also bestow other lecturers for the guidance they offered me by providing their valuable feedback as examiners in proposal defense and interim presentations. Further I would like to pay my sincere gratitude to Dr. H.N.D. Thilini, computer science project coordinator, and all other UCSC staffs members for all the assistance they provided to make this project success. Finally a special thanks go to my family for their continuous support, encouragement and being there for me in all my hardships.

Table of Contents

List of Figures

List of Tables

Chapter 1 - Introduction

1.1 Background to the Research

Multi Object Tracking(MOT) is a computer vision problem where we have to track multiple objects throughout its lifetime in a video. The evolving applications of MOT, such as surveillance, self-driving cars, sports analytics, etc. make it as an inevitable problem to be addressed with potential business value. In all of these applications, Humans and their interactions with other objects plays an important role. For example, in surveillance we track human movements to identify malicious activities, in self-driving cars, tracking pedestrians is a crucial part and in sports analytics, we track players and their interactions with football to analyze the game. If we improve human tracking(HT) in MOT, we can improve overall MOT performance.

MOT is an extension to Object Detection(OD) problem, where we locate specific objects in an image or sequence of images. They key difference of OD and MOT is, in OD, we won't re-identify the same objects instead we only locate them. But in MOT, instance IDs are assigned to different objects, called reidentification(ReID), such that the same object has a consistent unique IDs throughout the video sequence. ReID should be improved to improve MOT performance. In ReID object information are used, mainly appearance features and compare objects with our target object to identify best match.

In MOT appearance features should be highly distinguishable for objects like humans,because humans are similar in appearance when compare to other objects. We need some additional information such as depth to compare humans more accurately in MOT. Depth information can be easily extracted from the image using other existing methods in computer vision domain such image classification, video classification, etc. .

We are focusing on this particular problem by introducing new video features to improve MOT performance and analyzed its effect on MOT. We used depth information extracted from scene and analyzed its contribution towards MOT per-

formance improvement.

1.2 Research Problem and Research Questions

1.2.1 Research Problem

There are many appearance feature extractors have been introduced in the literature but all of them are only extract features based on static images. In MOT, when we track humans we track humans in motion in most cases. The features we are going to use should have the combination of both appearance and motion information. To extract information we need new feature extractors.

We are focusing on tracking humans in MOT. The motion features with appearance knowledge existing on models which are trained on videos. Especially if we need human motion features, we need models which trained on human activities. Human Activity Recognition models perform very well in capturing motion features. Another promising domain we can refer for our problem is monocular depth estimation. With existing zero-shot monocular depth estimators we can extract depth without need for training models.

Therefore we are going to utilize these cross domain models such as human action recognition models and monocular depth estimators to extract additional features such as video features and depth cues to improve MOT performance by improving human tracking.

1.2.2 Research Questions

1.2.3 Research Question(RQ1): How can we improve discrimination of humans using video features from video classifiers in MOT?

We built a new feature extractor to extract video features and replacing the appearance features which currently used in the domain.

1.2.4 Research Question(RQ2): How to enhance the occlusion handling using depth information during human tracking?

We focused on the challenge of distinguishing objects with similar appearances or shapes, especially when multiple similar objects are present in a scene. Incorporate additional information to improve discrimination and maintain accurate object identities throughout the tracking process.

1.3 Justification for the research

We are trying to solve some important limitations and challenges in MOT, to advance MOT performance.

- **Motion Specific Features:** The existing MOT pipelines have the important component as feature extractors. Even though there are many MOT tracking algorithms are available, they are trying to use same existing features from feature extractors which are not optimized for MOT tasks. Trying to optimize the tracking algorithm for the irrelevant features limits MOT performance. Using appropriate representations from pretraining, we enable the model to capture more meaningful and contextual information from the input data. This can lead to more accurate and robust tracking results.
- **Old wine in a new package:** We try to enhance the performance of MOT algorithms with the help of existing models which are performing in other computer vision downstream tasks. So we find a way to use those models which have potential information required for MOT. It help us to use valuable knowledge which can be utilized without any additional burdens such as training.
- **Training Free Modules:** Training deep learning models are resource and time intensive task require more experience and knowledge. Especially tasks like MOT, require multiple training modules from feature extractor to tracker, but there are few tracking algorithms which are only relies on features, there fore they do not require training in nature. We in cooperate those tracking algorithms in our experiments and try push their limits with our experiments.

- **Data Efficiency:** We are utilizing existing scene information, which can be extracted from the scene and use them to feed more information into model which they have to learn in hard way during training. This makes MOT more feasible in scenarios where obtaining large-scale annotated datasets is challenging or costly.

We aims to improve MOT performance by addressing limitations and challenges. It focuses on motion-specific features, using pretraining representations to capture more meaningful information from input data.

1.4 Methodology

1.4.1 Datasets

The MOT Challenge dataset was first introduced in 2015 (Milan, Leal-Taixé, Reid, Roth & Schindler 2016). It is a benchmark dataset that provides various challenges for MOT algorithms such as crowded scenes, partial occlusions, fast motions, and different camera viewpoints. The dataset is annually updated with new challenging scenarios, making it one of the most comprehensive and diverse MOT datasets available.

The key advantage of the MOT Challenge dataset is its large size and diverse nature, which allows for fair comparisons of MOT algorithms across different domains. Moreover, the MOT Challenge dataset is publicly available and allows researchers to test their algorithms on a standard benchmark, which helps to promote reproducibility and comparability in the field.

We used three main MOT Challenge datasets. MOT16(Milan, Leal-Taixé, Reid, Roth & Schindler 2016), MOT17(Milan, Leal-Taixé, Reid, Roth & Schindler 2016) and MOT20(Dendorfer, Rezatofighi, Milan, Shi, Cremers, Reid & Roth 2019)

1.5 Outline of the Dissertation

The Thesis is organized as follows format. In Chapter 2 , we first addresses what MOT problem is and how the domain is evolving with additional problems. We thoroughly analyzed te applications of the MOT methods and how MOT applica-

tions are evolved over the past few years with evolution of modern science. We also cover some other areas of computer vision domain where we are going to refer some methodologies and models to improve MOT by analyzing the potential of their domains and how they are going to helpful in improving MOT. We are highlighting those limitations in MOT and place for the improvements introduced in this Chapter 1. In Chapter 3, the our research design and the high-level architecture of our novel video feature extractor of our first research question and second research questions designs are given in that chapter. While Chapter 4 addresses the implementation details for research design which describe in Chapter 3 along with the experimental benchmarks and standard evaluation related methods. In Chapter 5 we compare the experimental insights obtained with a ablation study by following the various methodologies and breakdown them for clear understanding. Conclusion and future works will be in Chapter 6.

1.6 Delimitations of Scope

We are using extensive evaluation metrics to evaluate various aspects of our methodologies and comparing with one of the standard baseline. Because we are exploring new approach into the domain where it cannot be compared with complex approaches.

1.6.1 In scope

- We are introducing novel video features for tracking by utilizing existing computer vision models.
- We extract depth information and fusion strategy to integrate with existing tracking algorithm to analyze contribution of depth cues in MOT.

1.6.2 Out Scope

- We are are focusing only on tracking humans in the MOT domain due to the complexity of distinguishing humans and real world MOT applications.
- We are using off-the-shelf models and existing algorithms, we are not implementing any of new algorithms.

Chapter 2 - Literature Review

2.1 Mutli Object Tracking(MOT) problem

The MOT problem, which is a significant area in computer vision, is the task of identifying the objects in each video frame. MOT is famous for its potential applications, including surveillance, human-robot interaction, and autonomous driving. MOT's objective is to associate the correct identity to each object and to maintain the association over time despite occlusions, appearance changes, and other challenges.

The number of surveys is increasing. Some surveys are application-oriented, like this paper, (Alimi et al. 2021) provides a survey of object detection based on deep learning and tracking in self-driving cars. Some surveys are at a high level and only include the recent approaches. Even though these surveys provide valuable insights into the SOTA in MOT. They do not particularly address the MOT domain in the whole spectrum from classical models to modern deep learning-based models. We attempted to cover most of the important aspects of the MOT instead of an in-depth analysis of the models by delivering our main contributions through this paper are:

- Revisiting classical MOT approaches
- Analyzing deep learning-based MOT approaches
- Identifying benchmark datasets
- Examining evaluation metrics of MOT
- Exploring applications of MOT

Our primary goal in doing this work is to establish a solid foundation of MOT for researchers and practitioners, who wants to know or work with MOT. They can use this review to get a better comprehension of the state of MOT approaches today and identify areas for future research without getting stuck into complex modalities.

2.1.1 Mathematical Formulation

We can identify multiple-object tracking (MOT) as an optimization problem that uses joint distribution estimation. We created the following formulation to understand MOT in a mathematical way.

Let D_t indicate the set of detections at t time, and L_t denote the set of objects present in the frame at t time. Estimating the joint distribution $p(D_{1:T}, D_{L:T})$ is the aim of MOT, where T denotes how many total frames there are in the video sequence.

The joint distribution can be factorized as follows:

$$p(L_{1:T}, D_{1:T}) = p(L_1)p(D_1|L_1)\prod_{t=2}^{T} p(L_t|L_{t-1})p(D_t|L_t) \tag{2.1}$$

The beginning term $p(L_1)$ is the scene's prior distribution of objects at $t = 1$. The second term $p(D_1|L_1)$ is the likelihood of the detections given to the objects at time $t = 1$. The third term $p(L_t|L_{t-1})$ denotes the transition probability from the time step $t - 1$ to the time step t. The fourth term $p(D_t|L_t)$ is the likelihood of the detections given to the objects at time t. The joint distribution is estimated by factoring each term to provide a more computationally efficient solution. By estimating the joint distribution, we can track multiple objects over time, which is crucial for many computer vision applications.

The joint distribution's estimation $p(L_{1:T}, D_{1:T})$ is a fundamental problem in MOT. Various filter-based methods are commonly used to estimate this distribution. In the upcoming chapter, we will provide more detailed information on these methods and their evolution.

2.1.2 Challenges in MOT

When we try to solve the problem of MOT, there are specific challenges that should be tackled. The following are some of the major challenges in MOT:

- Occlusions: When tracking objects it is very hard to handle occlusions, where objects are temporarily or fully hidden from the camera's view. This can lead to tracking failures or errors when objects reappear.
- Appearance changes: While we are tracking, objects can change their appearance over time due to factors such as illumination, orientation, or scale.

Tracking may fail to recognize and track objects under such conditions, leading to tracking errors.

- Ambiguity: In crowded scenes with multiple objects, it can be challenging for tracking objects. Because distinguishing between objects with similar appearances leads to tracking confusion and errors.
- Computational complexity: MOT is a complex problem to solve, therefore the solutions can be computationally expensive, especially when dealing with large-scale or long-term tracking scenarios like surveillance. This can limit their real-time performance and practical applicability

Even though MOT is a challenging problem as stated above, the following factors make it a significant issue as well.

2.1.3 Importance of MOT

MOT is a significant research area in computer vision due to its wide range of applications. Some of the key reasons for its importance are:

- MOT algorithms provide a foundation for more advanced tasks including object recognition, activity recognition, and behavior analysis.
- MOT is a crucial component in domains, where we need to accurately track multiple objects in real-time. Those domains need MOT for decision-making and to ensure people's safety.
- MOT is a significant part of many business domains such as surveillance, robotics, and autonomous driving, where MOT is a core component, and without perfect MOT that business will be at risk.
- MOT can help to unlock the full potential of other tasks in computer vision, including scene understanding and action recognition. For example, tracking objects over time can provide further details about the objects and their interactions, which can be used to better understand the scene and identify important events.
- The development of MOT algorithms has led to numerous technological innovations and advances when it comes to computer vision, including the application of deep learning and the fusion of several modalities such as RGB, depth, and radar.

We could notice that these reasons are driving forces in the disruptive development of the MOT domain in recent years. Unleashing high-powered computational systems for building perfect MOT models and low-powered edge devices to deploy efficient MOT algorithms makes MOT more accessible to a variety of people from highly intellectual people like researchers to ordinary people. Therefore, the need for a strong foundation in MOT is needed for all levels of people. We tried to make this work as possible as to reach all those people. Therefore we will start with the classical methods used to solve MOT problems.

2.1.4 Applications of MOT

Advancements in more and more accurate and fast MOT models attracted many researchers to use on practical applications. is famous for its potential applications, such as surveillance and security, autonomous vehicles, robotics, sports analytics, and medical imaging.

Surveillance and Security

MOT has several applications in the field of surveillance and security. It can be used for detecting and tracking suspicious objects or persons, monitoring crowd movements, and identifying potential threats in public places. For instance, in airports and train stations, MOT can be used to track individuals and identify their movement patterns to detect any suspicious behavior. Multiple objects can be tracked in real-time with high accuracy can enhance situational awareness and improve response times in critical situations.

MOT can also be used in traffic surveillance to monitor the movement of vehicles and pedestrians, ensuring safety and security on roads and highways. By tracking the movements of vehicles and pedestrians, authorities can identify traffic congestion and take necessary measures to ensure smooth traffic flow.

Moreover, MOT can be used in border control and maritime surveillance to find and track any suspicious waterborne objects or vessels. In this case, MOT is used to detect and monitor potential security threats in the sea, including illegal fishing, drug trafficking, and smuggling.

The ability of MOT to accurately monitor several objects in real time has the

potential to enhance the safety and security of public places and improve response times to critical situations.(Ali & Shah, 2015)

Autonomous Vehicles

Multi-object tracking has become an important technology for autonomous vehicles, which rely on accurate tracking and detection of items in their surroundings in real-time. Autonomous vehicles require the ability to track multiple objects, such as other vehicles, pedestrians, and obstacles, to navigate and avoid collisions. Multi-object tracking can also be used for traffic flow analysis and prediction, as well as for intelligent intersection management.

An important aspect of MOT in autonomous vehicles is the integration with decision-making and control systems. The tracking information can be used to generate high-level situational awareness and to inform decision-making algorithms for path planning and trajectory prediction (Papadimitriou et al., 2015).

MOT is an essential technology for autonomous vehicles, enabling them to navigate and operate safely and efficiently in complex and dynamic environments.

Robotics

MOT has several applications in the field of robotics. MOT is particularly useful in robotics applications such as mobile robots, autonomous vehicles, and drones. MOT is also applicable to improve the safety and efficiency of industrial robots by tracking the movement of workers and other objects in the robot's workspace. Furthermore, MOT can be used in robotics to enable robots to interact with humans more effectively, by detecting and tracking human gestures and movements.

Multi-object tracking in robotics can be achieved using various sensors, including LiDAR, cameras, and radar. Many robotics applications require the capacity to precisely monitor several objects in real-time, which has prompted the creation of novel MOT algorithms and techniques.

The use of MOT in robotics has the potential to enable new applications and capabilities, such as autonomous delivery robots, mobile robots for environmental monitoring, and intelligent transportation systems. As such, MOT in robotics continues to be a busy field for research and development.

Sports Analytics

MOT has become an essential tool in sports analytics (Luo & Li 2018). It provides coaches, players, and fans with detailed insights into the performance of athletes and teams during games and training sessions. Multi-object tracking can be used to measure various aspects of player and ball movement, such as speed, acceleration, trajectory, and proximity to other players. This information can be used to analyze game strategies, evaluate player performance, and improve training programs.

MOT has several applications in various sports, including soccer, basketball, football, and hockey. In soccer, MOT can be used to analyze the movement of players and the ball during games, providing coaches with valuable insights into team performance (Cuturi & Blondel 2018). In basketball, MOT can be used to analyze the movement of players and the ball during games, providing coaches with valuable insights into team performance (Liu et al. 2019). In football, MOT can be used to analyze the movement of players and the ball during games, providing coaches with valuable insights into team performance (Pfister et al. 2014). In hockey, MOT can be used to analyze the movement of players and the puck during games, providing coaches with valuable insights into team performance (Del Giorno et al. 2013).

The use of MOT in sports analytics has revolutionized the way coaches and players approach training and games. It has also provided fans with a more in-depth understanding of the sports they love. As the technology behind MOT continues to improve, it is anticipated to have a bigger impact on sports analytics in the future.

Medical Imaging

MOT is a powerful tool in medical imaging, allowing for the identification and tracking of multiple objects such as cells, tissues, and organs (Shaham 2010). MOT is used to track the movement of cells during development, monitor the progression of diseases, and assess the effectiveness of treatments.

In cancer research, MOT can be used to track the growth and movement of tumors over time. This can help to identify the spread of cancer and assess the effectiveness of treatments. MOT is also applicable in neuroimaging to track the movement of neuronal cells and identify abnormalities in brain function.

Another application of MOT in medical imaging is in the field of ophthalmology. The movement of retinal vessels and identifying changes in blood flow can be an early indicator of diseases such as glaucoma and diabetic retinopathy. MOT is used to track the movement of retinal vessels.

The ability to track multiple objects simultaneously in medical imaging can provide a more comprehensive understanding of disease progression and treatment efficacy. Multi-object tracking also has the potential to increase the accuracy and efficiency of medical diagnoses, leading to better patient outcomes. Therefore MOT is a promising tool in medical imaging and has the capacity to significantly improve our comprehension of disease and improve patient care.

This variety of applications clearly explains the power of MOT and motivates researchers to get involved in resolving research gaps in MOT domains. This has a significant impact on how all of these extensive works in MOT have developed. But there is a catch, the MOT field is still emerging and it has its own shortcomings. We have analyzed a few significant aspects of MOT in the next chapter.

2.2 Human Action Recognition

2.2.1 Problem

Human action recognition task indicates classify the human's action using sensor data. It is the ability to recognize and deduce human activity or movement from human body motions or movements using sensors. There are several kinds of human behavior. These acts can be divided into two major groups: voluntary acts and involuntary activities (Buehner (2015)). Numerous CV approaches are presented in the literature (Hassaballah & Hosny (2019), Khan et al. (2020)) to assist with the laborious and error-prone operation of manually recognising these motions in real-time. Numerous classical strategies, including shape, texture, point, and geometric aspects, form the basis of the majority of the suggested solutions (Kolekar & Dash (2016)). A number of methods are predicated on human temporal information (Hermansky (2006)), and some of them extract human silhouettes prior to feature extraction (Krzeszowski et al. (2019)).

2.2.2 Significance of Problem

Due of the multitude of human events that occur in daily life, the HAR process is a difficult undertaking. Deep learning models are applied to address this problem. The quantity of training data a deep learning model has is always what determines its performance [20]. Several datasets are available to the public for use in the action recognition challenges.These datasets contain a wide range of behaviors, such as walking, running, jumping out of a car, throwing, kicking, boxing, falling, bending down, and many more.

The following are a few additional major HAR challenges:

1. In order to identify the focal point in the most recent frame, query video sequence resolution is essential.
2. It is challenging to classify the correct human activities using automatic activity recognition under multi-view cameras due to the complexity of the background, shadows, lighting, and outfit conditions that extract irrelevant information using classical techniques of human action.
3. (iii) Unbalanced datasets affect a model's ability to train Because, change in motion variation catches incorrect actions under multi-view cameras. So, For model to learn, a large volume of training images is always required.
4. when features are extracted from whole video sequences, numerous irrelevant characteristics are included, which can negatively impact classification accuracy.

2.2.3 Datasets

In the Human Action Recognition (HAR) task, there are four publicly available datasets - KTH (Chen et al. (2021)), Hollywood (Melhart et al. (2022)), WVU (Hassan et al. (2018)), Kinetics-400 (K400) (Wojke et al. (2017)) and IXMAS (Joshi et al. (2020)). Each dataset consists of 10,000 video frames used for experimentation.

2.2.4 Model

In the field of computer vision (CV), deep learning has recently demonstrated encouraging results (Hassaballah & Awad (2020)). By simulating how the human brain processes information, deep learning generates models that facilitate learning and data representation on several levels (Voulodimos et al. (2018)). In the HAR also, Deep learning models achieved best results. Initially, For the purpose of recognizing the final activity, (Ahmed Bhuiyan et al. (2020))used acidometer sensors to extract the spatial characteristics and multiclass SVM for classification. After that a unified framework was presented by Zhao et al. (2020) for activity recognition. The end outcomes were a combination of short- and long-term traits. For impleenting action recognition with CNN , Muhammad et al. (2021) coupled the dilated CNN model characteristics with the attention-based LSTM network. Similarly, Li et al. (2021) presents a skeleton based attention framework for action recognition. The shape and the OFF features were both used in the HAR framework that Kolekar & Dash (2016) described (Im et al. (2020)).

SVM and the Hidden Markov Model (HMM) are used to create the framework that is being described. The HMM classifier is used to extract and use the shape and OFF features for HAR. The background extraction of the image was done using the multi-frame averaging method. The length feature set from the middle to the body contour had its magnitude reduced using a discrete Fourier transform (DFT). The principle component analysis was suggested as a means of choosing features. The proposed framework demonstrated optimal accuracy when evaluated on real-time video recordings.Weifeng and his colleagues introduced a new approach called Laplacian Regularized Sparse Coding (LRSC) for Human Activity Recognition (HAR). This method was a more advanced version of graph Laplacian with improved performance. They also developed a fast-iterative algorithm to optimize LRSC. The sparse codes generated by LRSC were then used in a support vector machine (SVM) for classification purposes. The experiments were conducted using USAA and HMDB51 datasets, showing the effectiveness of LRSC in HAR tasks.

The authors Jalal et al. (2017) introduced a model for Human Activity Recognition (HAR) utilizing depth video analysis. Hidden Markov Model (HMM) was utilized to identify common activities performed by elderly individuals living inde-

pendently. The initial phase involved analyzing the depth maps through a temporal motion identification technique using human silhouette segments in a specific setting. Durable characteristics were chosen and combined to detect changes in gradient orientation, temporal intensity differences, and local movements of the body organs.

The researchers conducted experiments on three different datasets: Online Self-Annotated (Melhart et al. (2022)), Smart Home, and Three Healthcare. They achieved accuracies of 84.4 percentage, 87.3 percentage, and 95.97 percentage respectively. Muhammed and colleagues (Hassan et al. (2018)) introduced a framework for human activity recognition using smartphone inertial sensors. The framework involved three steps: extracting efficient features, reducing features using KPCA and LDA, and training resultant features with DBN for improved accuracy.

In their study, **?** introduced a model for action recognition to address the issue of multi-view human activity recognition (HAR). This algorithm, known as adaptive fusion and category-level dictionary learning (AFCDL), incorporated dictionary learning by implementing query sets and a regularization scheme for assigning adaptive weights. Additionally, Khan et al. (2021) proposed a new framework for composite action classification using a 26-layered CNN.

In their study, Kun and colleagues Xia et al. (2020) introduced a Human Activity Recognition (HAR) model based on Deep Neural Networks (DNN) that combines convolutional layers with Long Short-Term Memory (LSTM). This model was able to automatically extract features and classify them using standard parameters. Recently, there has been significant progress in developing deep learning models for HAR using high-dimensional datasets. Traditional methods for HAR were not performing well, especially on large datasets. However, modern techniques such as LSTM, SV-GCN, and CNNs have shown improved performance.

Finally, MAE-ST (He et al. (2022)) explored a straightforward extension of Masked Autoencoders (MAE) to learn spatiotemporal representations from videos. By randomly masking spacetime patches in the videos, an autoencoder is trained to reconstruct them in pixels. Surprisingly, the MAE method is able to learn powerful representations without any specific bias towards spacetime (except for patch and positional embeddings), and random masking without considering spacetime

performs the most effectively.

2.2.5 Potential Applications

Applications for HAR can be found in many different fields, such as video reclamation, human-computer interface (HCI) Chen et al. (2021), surveillance Mishra et al. (2021), and visual information interpretation Khan et al. (2024). Video surveillance is the most significant use of action recognition (Liu et al. (2021)). Governments employ this application for security purposes Ahmed et al. (2021), intelligence gathering, crime investigation Wang et al. (2021), and even to lower the crime rate. The primary driving force behind the expansion of HAR research is its application in video surveillance Zin et al. (2021). When it comes to visual surveillance, HAR is essential for identifying people' movements in public areas. Additionally, the surveillance of smart cities can benefit from these kinds of systems (Farnoosh et al. (2021)).

2.3 Zeroshot monocular depth estimation

2.3.1 Monocular Depth Estimation

Estimating depth from a single camera view is called monocular depth estimation, is crucial in the realm of computer vision, especially in disciplines like robotics and autonomous driving. The idea of turning any regular camera into a tool that can detect depth levels across a wide range is quite exciting, as it not only lowers costs but also enhances the depth of information captured. With that, we can estimate depths from videos without need for any additional information.

2.3.2 Siginificance

There are many devices available for depth information, but they are often too expensive, slow, and limited in range for consumer use. Devices like the Kinect sensor are popular for consumer products. These sensors use Time-of-Flight technology to measure depth by calculating how long it takes light to travel from a source to an object and back. Time-of-Flight sensors work well indoors and for short distances (¡2 meters). Meanwhile, LiDAR scanners are used for outdoor 3D

measurements. LiDAR sensors have several advantages over other types of sensors. LiDAR sensors have advantages such as high resolution, accuracy, performance in low light, and speed. Yet, they are costly and demand a lot of power, making them unsuitable for consumer products. Moreover, buying a 3D camera can be expensive. Alternatively, we could buy two cheaper cameras and utilize the stereo camera method to gauge depth. But, with two cameras also there is a cost.

Monocular Depth Estimation helps to reduce this cost. And many binocular or multi-view methods can accurately estimate depth information, but they face significant challenges in terms of computational time and memory requirements, which can hinder their application in various scenarios. In machine perception, recognizing important factors like scene shape and image independence is crucial. Depth Estimation (DE) shows promise in various applications, from robotics to computer graphics.

2.3.3 Limitations

1. MDE face a challenge when it comes to diversity in training data, especially in different subjects and types of images. For example, if the training dataset doesn't have enough photos of the sky, it can make it harder to accurately measure depth in those specific areas.
2. Capturing the overall characteristics of a scene, such as texture changes or blurry details, is challenging from a computational standpoint.

2.3.4 Applications

Monocular depth estimation has many uses, such as creating 3D models, enhancing virtual reality experiences, aiding self-driving vehicles, and improving robot capabilities.

Chapter 3 - Design

In this chapter, we outlined our approach to answer the research questions introduced in chapter 01. The first section focused on answering "How can we improve discrimination of humans using video features from video classifiers in MOT?". In the second section, we explored strategies such that to answer the second question "How to enhance the occlusion handling using depth information during human tracking?". Our experiments follow the traditional MOT pipeline structured into five main stages: frame extraction, localization, feature extraction, feature association, and reidentification. Then we evaluate our results with benchmark datasets.

3.1 Conventional MOT pipeline

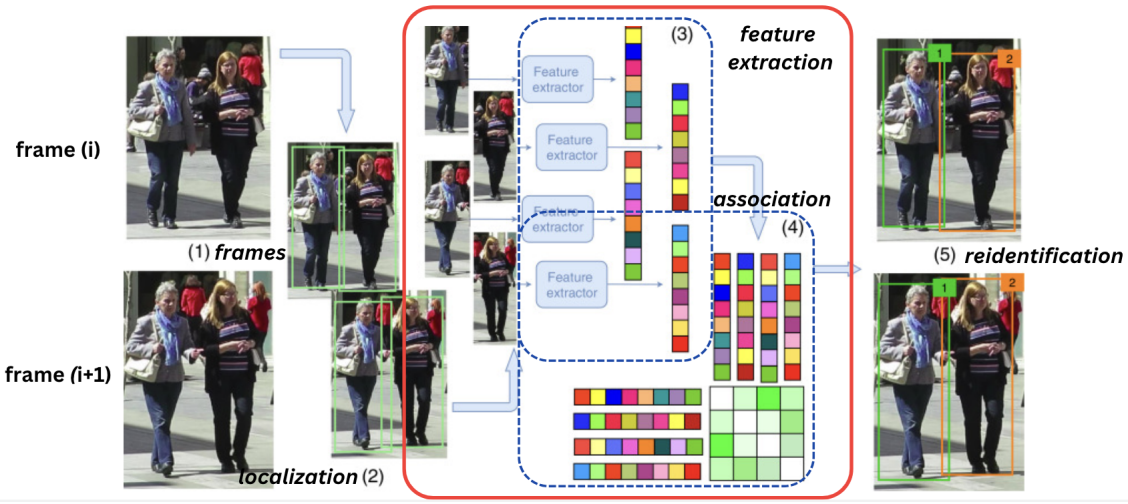

Figure 3.1: The conventional MOT pipeline with five stages. We denoted the stages where we are contributing in red rounded rectangle.

In the conventional MOT pipeline, initially, we extract individual frames from the video sequence because we always process videos as set of **frames** in MOT context. Then we **localizate** the objects in each frame by extracting the spatial coordinates or bounding boxes of the target objects using object detectors /(¡Required citations for YOLO and other object detectors/). Using those bounding boxes, we extract **features** of objects in the frame by inserting into the **feature extractor**

which gives features having appearance attributes such as color, shape, or texture, essential for object identification and tracking. We perform **feature association** on those extracted features to link objects across consecutive frames to establish coherent object trajectories over time. We give the unique identification number for highly correlated objects with our target object in the **reidentification** phase to ensure the continuity and accuracy of the tracking process throughout the video sequence.

We are using this conventional MOT pipeline as the backbone of our experimental methodology, for fair comparison and evaluation within the domain of MOT research.

3.2 Research Design for RQ1: Research Feature Extractor

In this part of the study, we aim to address our first research question through our experiments with new feature extractor. To do so, we selected the top video classifier, MAE-ST(Feichtenhofer et al. 2022), as we require video features. After customizing the video classifier to identify human attributes, we assessed the tracking accuracy of humans using video features and compared it to that of human features.

3.2.1 Data Pre-processing

We used all the model specific preprocessing techniques for the best performance of the models. We did't use any additional or new preprocessing techniques.

3.2.2 Baseline model

We used image feature extractor addressed in the Deepsort(Wojke et al. 2017) to achieve optimal results as it's crucial to have a feature embedding with strong discrimination abilities. This basleline was nalready trained before using the online tracking application. To do this, Convolutional Neural Network (CNN) trained on a large dataset (Zheng et al. (2016)) of over 1,100,000 images of 1,261 pedestrians for deep metric learning in people tracking. The structure of our CNN network can

be found in the following table:

Name	Patch Size/Stride	Output Size
Conv 1	$3 \times 3/1$	$32 \times 128 \times 64$
Conv 2	$3 \times 3/1$	$32 \times 128 \times 64$
Max Pool 3	$3 \times 3/2$	$32 \times 64 \times 32$
Residual 4	$3 \times 3/1$	$32 \times 64 \times 32$
Residual 5	$3 \times 3/1$	$32 \times 64 \times 32$
Residual 6	$3 \times 3/2$	$64 \times 32 \times 16$
Residual 7	$3 \times 3/1$	$64 \times 32 \times 16$
Residual 8	$3 \times 3/2$	$128 \times 16 \times 8$
Residual 9	$3 \times 3/1$	$128 \times 16 \times 8$
Dense 10		128
Batch and ℓ_2 normalization	128	

Table 3.1: Overview of the CNN architecture. The final batch and ℓ_2 normalization projects features onto the unit hypersphere. (Wojke et al. 2017)

Essentially, our baseline has wide residual network (Zagoruyko & Komodakis (2016)) with two convolutional layers followed by six residual blocks is used in this CNN. The global feature map, having a size of 128, is calculated in dense layer 10. After that, a final batch and L2 normalization process projects features onto the unit hypersphere to ensure compatibility with our cosine appearance metric. In total, the network has 2,800,864 parameters, and it takes around very less time for a single forward pass with **32** bounding boxes on mobile GPU.

3.2.3 Masked Auto Encoder(MAE) architecture

MAE is transformer architecture which is trained using self-supervised pre-training approach, which is already outperfrom in natural language processing (NLP) through methods like autoregressive language modeling in GPT(Brown et al. 2020) and masked autoencoding in BERT(Devlin et al. 2018). These approaches involve removing parts of data such as text, image, etc. and training models to predict the missing part.

While masked autoencoders(mae) are useful in both language and computer vision, there are differences. In vision, convolutional networks (LeCun et al. 1989) have historically dominated, posing challenges for integrating masking mechanisms due to their regular grid operation. However, this obstacle has been overcome with the introduction of Vision Transformers (ViT) (Dosovitskiy et al. 2020). The information density varies between language and vision, with language being highly semantic and information-dense, while images exhibit spatial redundancy. To address this, a strategy of masking a high proportion of random patches in images is proposed to encourage holistic understanding and reduce redundancy. The autoencoder's decoder differs in its role between text and images, with the latter reconstructing missing patches in pixel space. Based on this analysis, the Masked Autoencoder (MAE) is designed to mask random patches from input images and reconstruct missing patches, employing an asymmetric encoder-decoder design to efficiently handle both tasks

3.2.4 Video Feature Extractor

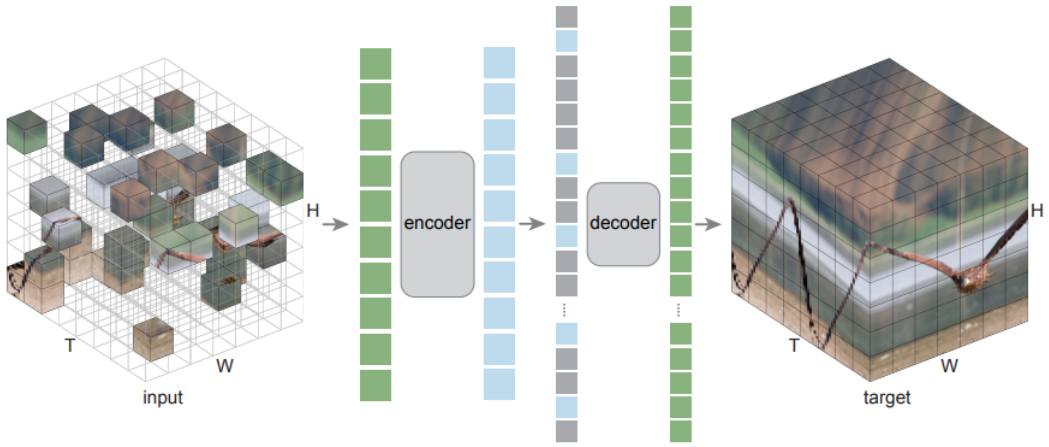

Figure 3.2: Masked Autoencoders as spatiotemporal learners

MAE-ST Architecture is build upon the foundation laid out by MAE architecture by He et al. (2022) principles which is mentioned above, adapting them to analyze spacetime data. The goal of this MAE-ST is to create a method that works within a broad and cohesive framework, reducing the need for specialized domain knowledge. The main flow of this architecture is divided into 3 parts. They are Patch embedding, Masking and Autoencoding which are explained below.

Patch embedding. Similar to the original Vision Transformer (ViT Dosovitskiy et al. (2020)) , in the MAE-ST video clip is divided into a grid of patches that do not overlap (Bertasius et al. (2021), Fan et al. (2021), Wei et al. (2022)). These patches are then flattened and projected linearly (Dosovitskiy et al. (2020)). To enhance the embedded patches, positional embeddings (Vaswani et al. (2017)) are included in the MAE-ST. This combined process of embedding patches and positional information is the only spacetime-aware operation in this MAE-ST methodology.

masking. As a next step,in the realm of processing spacetime data, the masking technique is utilized. This technique involves a random patch sampling method that is unaware of the underlying spacetime structure, similar to techniques used in BERT and MAE. The ideal masking ratio, which is suggested to be linked to data redundancy, is explored through real-world observations, revealing an optimal ratio of 90 perentage. This indicates that natural videos possess more redundancy than images due to their temporal consistency. Furthermore, it showcases the effectiveness of spacetime-agnostic sampling in comparison to structure-aware strategies, as the former efficiently uses visible patches, enabling higher masking ratios and potentially overcoming challenges presented by intricate pre-training tasks.

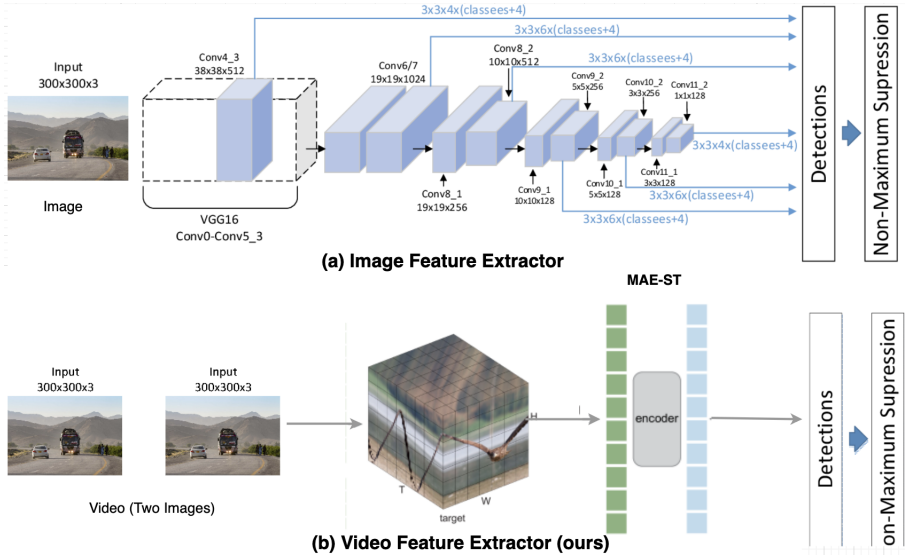

Figure 3.3: High level architecture diagram of our video extractor in comparison with traditional image feature extractor.

Auto encoding. AS a final step, auto encoding was done by the vanila ViT

model (Dosovitskiy et al. (2020))which is used by the encoder to work on visible embedded patches, reducing computational complexity while remaining practical. The encoder's complexity is minimized by a high masking ratio. On the other hand, the decoder, which is smaller in design but also based on a ViT model, processes the combined encoded patch set and mask tokens. Despite working with the entire set, the decoder's complexity is lower than that of the encoder, leading to a notable decrease in the overall complexity of the autoencoder.

We extracted the encoder from the MAE_ST architecture because it has all the knowledge about video understanding and used to get our video features.

3.3 Research Design for RQ2: Depth Cues for MOT

3.3.1 Depth Map Extractor

We use MiDaS (Multiple Depth Estimation Accuracy with Single Network) Ranftl et al. (2020) as a depth map extractor to extract depth maps of target objects to track. MiDAS is a sophisticated deep learning model that leverages residual connections and is built on top of ResNet for predicting depth with a single image. MiDaS has shown remarkable effectiveness in determining depth from single images. Let's take a closer look at the structure of MiDaS:

1. **Encoder-Decoder Architecture:-**The foundation of MiDaS is an encoder-decoder architecture, in which the encoder extracts high-level features and the decoder uses upsampling to create the depth map from these features.
2. **Backbone:-** Since ResNet-50 and ResNet-101 are resistant against vanishing gradients, MiDaS usually employs them for feature extraction. enabling MiDaS to capture hierarchical information at different sizes by extracting multi-channeled feature maps from input photos.
3. **Multi-Scale Feature Fusion:-** MiDaS incorporates skip connections and feature fusion to provide precise depth estimate. In order to access low level details during upsampling, feature maps from older layers are connected to the later layers via skip connections. Feature fusion ensures that both local and global information is effectively exploited for depth estimate by combin-

ing the multi-scale feature maps.

4. **Up-Sampling and Refinement:-**By utilising up-sampling, the final depth map is produced. Bi-linear interpolation and transposed convolutions are often employed approaches for upsampling, which aim to enhance the spatial resolution of feature maps. Refining the depth estimation involves combining the depth maps with associated skip links using feature fusion.

3.4 Evaluation

We use domain standard benchmark datasets and evaluation metrics to evaluate the reliability and accuracy of our approach in a quantitative way. We compare the performance of several existing approaches, and improve our approach with the help of the benchmark performance evaluation.

3.4.1 Benchmark Datasets

MOT Challenge

The MOT Challenge dataset was first introduced in 2015 (Milan, Leal-Taixé, Reid, Roth & Schindler 2016). It is a benchmark dataset that provides various challenges for MOT algorithms such as crowded scenes, partial occlusions, fast motions, and different camera viewpoints. The dataset is annually updated with new challenging scenarios, making it one of the most comprehensive and diverse MOT datasets available.

The key advantage of the MOT Challenge dataset is its large size and diverse nature, which allows for fair comparisons of MOT algorithms across different domains. Moreover, the MOT Challenge dataset is publicly available and allows researchers to test their algorithms on a standard benchmark, which helps to promote reproducibility and comparability in the field.

The MOT Challenge dataset has played a significant role in advancing SOTA MOT research by offering a comprehensive benchmark for evaluating the performance of MOT algorithms. The continued development and use of the MOT Challenge dataset will continue to drive progress in this field and help to address the remaining challenges in MOT (Dendorfer, Rezatofighi, Milan, Shi, Cremers &

Reid 2019, Leal-Taixé et al. 2015, Milan et al. 2017, Milan, Heng & Sminchisescu 2016, Milan et al. 2018).

3.4.2 Evaluation Metrics

MOT problem has always been a challenge to assess properly. Some existing metrics tend to focus too much on either detecting objects or connecting them together. To tackle this issue, higher order tracking accuracy (HOTA) (Luiten et al. 2021) is a suitable metric. Because this metric carefully considers the impact of accurate detection, association, and localization, combining them into one comprehensive measure for comparing different trackers (Luiten et al. 2021).

HOTA metric is combined metric of three IoU scores, assessing detection, association, and localization tasks separately and then amalgamating them into a final HOTA score by combining the individual IoU scores. We apply the HOTA on our experiments because of its ability to capture critical aspects of MOT performance by combining three different aspects of MOT.

IOU Scores

The IoU, also called the Jaccard Index, is used to measure the overlap between predicted and ground truth bounding boxes.

Location Accuracy (LocA)

Localization Accuracy (LocA) is a way to check how well predicted detections match with actual detections. The measure called Localization Intersection over Union (Loc-IoU) is commonly used to measure how accurate the localization is. It is calculated by comparing the overlap between the two detections with the total area covered by both. Check out the diagram below for a visual representation.

We can apply this metric for both bounding boxes and segmentation masks. As we can observe, when the Loc-IoU score goes up, the predicted detections and the actual detections are more closely positioned and the accuracy of localization is enhanced. We can assess the overall Localization Accuracy (LocA) by averaging the Loc-IoU for all matching pairs of predicted and actual detections in the entire dataset (we will explain later how we determine these matches).

$$\text{LocA} = \frac{1}{|\text{TP}|} \sum_{c \in \text{TP}} \text{Loc} - \text{IoU}(c)$$

Detection Accuracy (DetA)

We use DetA to evaluate how well the algorithms perform. These metrics assess the object detector's ability to detect objects in video sequences and the number of false positives or false negatives it generates. Detection IoU (Intersection over Union) is a common metric used to evaluate detection accuracy. The process involves determining which predicted detections intersect with the ground-truth detections by setting a localization threshold (e.g. generally we set Loc-IoU $\geq$ 0.5). It is important to note that a single predicted detection may overlap with multiple ground-truth detections, and vice versa.

To address this, the Hungarian algorithm(Kuhn 1955) is employed to establish a one-to-one correspondence between predicted and ground-truth detections. When considering True Positives (TP), they represent the overlapping detections between the two sets. False Positives (FP) are the predicted detections that do not match, while False Negatives (FN) are the ground-truth detections that do not match. The measure of intersection, known as the detection IoU, is calculated as follows:

$$\text{Det-IoU} = \frac{|\text{TP}|}{|\text{TP}| + |\text{FN}| + |\text{FP}|}$$

This measure ois similar to Loc-loU, where we calculate the intersection of matches (TPs) divided by the union of all detections. Unlike Loc-IoU which compares a single predicted detection with a ground-truth detection, Det-loU compares all predicted detections with all ground-truth detections. This set-based loU calculation is also known as the Jaccard Index. To determine the overall Detection Accuracy (DetA), we can calculate Det-loU using the counts of TPs, FNs, and FPs across the entire dataset.

$$\text{DetA} = \text{Det-IoU} = \frac{|\text{TP}|}{|\text{TP}| + |\text{FN}| + |\text{FP}|}$$

Association Accuracy (AssA)

Association measures how effectively a tracker connects detections over time to the correct identities, based on the true set of identity links in the true tracks. We can calculate this by comparing a predicted detection with a true detection that are matched together, and evaluating the alignment between the predicted detection's track and the true detection's track. This alignment can be expressed using an Intersection over Union (IoU) formulation.

$$\text{Ass-IoU} = \frac{|\text{TPA}|}{|\text{TPA}| + |\text{FNA}| + |\text{FPA}|}$$

The red square represents the matched true positive pair of prediction and ground-truth detection that we are trying to find an association score for. To evaluate how well the temporal association lines up between these detections, we look at all the detections in these two tracks that match (true positive associations in green) and those that don't match (false positive associations in yellow and false negative associations in brown).

$$\begin{aligned}\text{AssA} &= \frac{1}{|\text{TP}|} \sum_{c \in \text{TP}} \text{Ass-IoU}(c) \\ &= \frac{1}{|\text{TP}|} \sum_{c \in \text{TP}} \frac{|\,\text{TPA}(c)|}{|\,\text{TPA}(c)| + |\,\text{FNA}(c)| + |\,\text{FPA}(c)|}\end{aligned}$$

The overall Association Accuracy (AssA) can be calculated by averaging the intersection over union of all pairs of matching predicted and ground-truth detections in the entire dataset.

Higher Order Tracking Accuracy (HOTA)

Even though all three aspects (localization, detection, and association) play a crucial role in determining the success of tracking. It is vital to evaluate each of these components. Nonetheless, there is a need for a unified measure to assess the overall performance of trackers. This measure is known as HOTA, which combines the three loU scores discussed previously:

$$
\begin{aligned}
\text{HOTA}_{\alpha} &= \sqrt{\text{DetA}_{\alpha} \cdot \text{AssA}_{\alpha}} \\
&= \sqrt{\frac{\sum_{c \in \text{TP}_{\alpha}} \text{Ass}^2 - \text{IO}_{\alpha}(c)}{|\text{TP}_{\alpha}| + |\text{FN}_{\alpha}| + |\text{FP}_{\alpha}|}} \\
\text{HOTA} &= \int_{0<\alpha\leq 1} \text{HOTA}_{\alpha} \\
&\approx \frac{1}{19} \sum_{\substack{\alpha=0.05 \\ \alpha+0.05}}^{0.95} \text{HOTA}_{\alpha}
\end{aligned}
$$

Earlier, we described DetA and AssA using a Hungarian matching method(Kuhn 1955) with a specific Loc-IoU threshold (a). As DetA and AssA scores are influenced by Loc-loU values, we compute these scores across various thresholds. For each threshold, we determine the overall score as the geometric mean of detection and association scores. By incorporating different thresholds, we account for localization accuracy in the final score.

When combining detection and association, the geometric mean is used to give equal weight to both aspects in the final score. If either detection or association is zero, the score will also be zero. This means that the HOTA score can be seen as a formulation of Det-IoU, where each true positive is weighted by the corresponding Ass-IoU. For example, the average of the Ass-IoU scores across all detections.

3.5 Summary

In this chapter, we have discussed the pipeline, architecture and the evaluation design to address the two research questions we are focusing on. As the initial step, methodology to extract video features was presented. Then we focus on depth feature extraction using depth feature extractor and depth fusion strategy. With that this chapter discussed the techniques of Evaluation methodologies to extensively evaluate various aspects of our methodologies. Finally, based on the results we observed by following these techniques and how we can improve them in future work are discussed in Chapter 5.

Chapter 4 - Implementation

In this chapter, we will outline how we experimented the methodologies covered in the previous Chapter 3. We explain into the preprocessing methods and provide insights into the research tool employed to address our research query.

All of our implementations were done using Python language and deep learning framework, Pytroch.

4.1 Data-preprocessing

One of the main limitation of the video feature extractor we discussed in the previous Chapter 3 is the backbone is mainly designed to work with videos. Therefore we have to modify the input data, image segment into video by duplicating image segment such that we will create video with minimum number of frames available, that is two.

4.2 Video feature extractor

We modified the official implementation of MAE_ST (Feichtenhofer et al. 2022) as a feature extractor by modifying the forward_encoder's predict function to return the features we need.

4.3 Segmentation Patch Extraction

Our segementation patch extraction algorithm code is in Figure 4.2.

4.4 Depth Extractor

Our depth feature extractor code is in Figure 4.4.

4.5 Depth Fusion

Our depth fusion strategy is implemented in Figure 4.4.

```python
def predict(self, np_images):
    """
    batch inference

    Params
    ------
    np_images : list of ndarray
        list of (H x W x C), bgr or rgb according to self.bgr

    Returns
    ------
    list of features (np.array with dim = 1280)

    """
    all_feats = []

    preproc_imgs = [self.preprocess(img) for img in np_images]

    for this_batch in batch(preproc_imgs, bs=self.max_batch_size):
        this_batch = torch.cat(this_batch, dim=0)
        this_batch = this_batch.unsqueeze(2)
        if self.gpu:
            this_batch = this_batch.cuda()
            if self.half:
                this_batch = this_batch.half()
        feats,_,_=self.model.forward_encoder(this_batch.repeat(1,1,2,1,1),0.9)
        all_feats.extend(feats.sum(dim=1).cpu().data.numpy())

    return all_feats
```

Figure 4.1: Code of modified forward_encoder's predict function

4.6 Research Tools

- We used Pytorch deeplearning framework to build our models.
- Official implementation of Deepsort(Danelljan et al. 2017) as a tracking algorithm.
- Used the Kaggle (GPU: NVIDIA P100 16GB) for all the experiments.
- Ultralytics library was used to get segmentation models.
- Trackeval script was used to evaluate the model performance.

```python
def extract_masked_patch(segmentor,img,is_masks=False,cls=0):
    """Extract image patch from bounding box.
    Parameters
    ----------
    segmentor : torch.module
        Image segmentation module.
    img : ndarray
        The full image.
    Returns
    -------
    ndarray
        Binary mask of object for future usage
    ndarray | NoneType
        An image patch showing the :arg:`bbox`, optionally reshaped to
        :arg:`patch_shape`.
        Returns None if the bounding box is empty or fully outside of the image
        boundaries.

    """
    #Get segmentationn masks
    results= segmentor.predict(Image.fromarray(img),max_det=1,classes=cls)
    # breakpoint()
    if len(results[0].boxes) == 0:
        if is_masks:
            return np.ones_like(img[...,0]), img
        else:
            return [0,0,img.shape[0],img.shape[0]], img
    # Get mask bounding boxes
    x1, y1, x2, y2 = results[0].boxes.xyxy.cpu().numpy().squeeze().astype(np.int32)

    if is_masks:
        # Create contour mask
        b_mask = results[0].masks.masks.cpu().numpy().astype(np.int32)[0,...].copy()
        # Resize mask to image size
        b_mask  = st.resize(b_mask , img.shape[:2], order=0, preserve_range=True, anti_aliasing
        # Maskout background
        idx=(b_mask==0)
        img[idx]=0
        return b_mask, cv2.resize(img[y1:y2, x1:x2], (img.shape[1],img.shape[0]))
    else:
        # Return new bounding boxes
        return [x1, y1, x2, y2], cv2.resize(img[y1:y2, x1:x2], (img.shape[1],img.shape[0]))
```

Figure 4.2: Code of image_patch_extraction function

```python
def depth_estimatation(img,midas,transform):

    # img = cv2.imread(filename)
    # img = cv2.cvtColor(img, cv2.COLOR_BGR2RGB)
    device = torch.device("cuda") if torch.cuda.is_available() else torch.device("cpu")
    input_batch = transform(img).to(device)

    with torch.no_grad():
        prediction = midas(input_batch)

        prediction = torch.nn.functional.interpolate(
            prediction.unsqueeze(1),
            size=img.shape[:2],
            mode="bicubic",
            align_corners=False,
        ).squeeze()

    return prediction.cpu().numpy()

def calculate_depthcof(masks,depth_map):

    depth_map=cv2.resize(depth_map, dsize=np.flip(masks[0].shape), interpolation=cv2.INTER_CU
    depth_points = []
    for mask in masks:
        depth_ = ((depth_map * 0.1) + 5.) * mask
        mask_sum = np.sum(mask)
        depth_cof = np.sum(depth_)/ mask_sum
        depth_points.append(np.ma.fix_invalid(depth_cof))

    return depth_points
```

Figure 4.3: Code for extracting depth features from object image patches.

```python
features = encoder(bgr_image, rows[:, 2:6].copy(),segmentor, is_mask)
depth_masks = [bbox_to_mask(bbox,bgr_image.shape) for bbox in rows[:, 2:6]]
if len(depth_masks) > 0:
    output = depth_estimatation(bgr_image,depth_encoder[0],depth_encoder[1])
    depth_cofs = calculate_depthcof(np.array(depth_masks),output)
    features = [feat*coff*0.1 for feat,coff in zip(features,depth_cofs)]
```

Figure 4.4: Code for fusion of depth cues with appearance features.

Chapter 5 - Results and Evaluation

In this chapter we present an analysis of effects on MOT performance through the techniques outlined in Chapter 3. We describe how we tackled our study topics and provide explanations and conclusions for the observed outcomes. This gives the reader insight into how various approaches vary in HOTA and other relevant measures.

We carried out extensive experiments on three datasets with comparison to baseline methods for detailed study of each approaches. We break down results in several categories for the ablation study of experimental results.

5.1 Results obtained from the First Research Question

We evaluated our new feature extractor, video feature extractor with our baseline model to identify the effect of video features in improving MOT performance. We modified the official implementation of deep sort algorithm and included our new feature extractor.

5.1.1 Video image feature extractor

MOT16

In our experiments, for MOT16 datasets (refer table 5.1) , model with video feature extractor gives low results for all types of sequences to the HOTA metric. For the combined or average results also, model with video feature feature not works well Overall, the video feature extractor model doesn't work well for the MOT16 dataset as mentioned in the below table; But when we consider DetA metric except for MOT16-02 and MOT16-11 our model performs better than baseline.

MOT17

In our experiments, for MOT17 datasets (refer tables 5.2 and 5.2), specifically in MOT17-02-DPM, MOT17-02-FRCNN,MOT17-02-SDP,MOT17-04-DPM,MOT17-

Sequence	Method	HOTA↑	DetA↑	AssA↑	LocA↑
MOT16-02	MARS	16.384	9.4892	28.299	80.385
	Ours	14.502	9.3878	22.421	80.152
MOT16-04	MARS	26.657	20.863	34.085	83.58
	Ours	24.886	20.867	29.723	83.265
MOT16-05	MARS	23.567	16.473	33.744	81.465
	Ours	18.995	17.11	21.135	81.018
MOT16-09	MARS	30.248	27.742	33.032	79.693
	Ours	25.167	27.797	22.899	79.236
MOT16-10	MARS	24.698	19.019	32.128	80.275
	Ours	22.985	19.175	27.624	79.962
MOT16-11	MARS	40.89	34.294	48.777	82.381
	Ours	31.491	34.093	29.142	81.771
MOT16-13	MARS	12.727	4.8646	33.388	79.043
	Ours	12.557	5.0394	31.395	78.801
COMBINED	MARS	25.592	18.362	35.7	82.195
	Ours	22.67	18.422	27.962	81.819

Table 5.1: HOTA metric for experiments on MOT16 with new feature extractor

05-FRCNN,MOT17-09-DPM,MOT17-10-DPM,MOT17-10-FRCNN,MOT17-10-SDP,MOT17-11-FRCNN,MOT17-11-SDP video sequences, model with video feature extractor gives better results for the HOTA metric. And for MOT17-04-FRCNN,MOT17-04-SDP,MOT17-05-DPM,MOT17-05-SDP,MOT17-09-FRCNN,MOT17-09-SDP,MOT17-11-DPM,MOT17-13-DPM video sequences, the model with video feature not gives better results when we compare with the model without video feature.But, when we see the combined or average results, model with dvideo feature does not works well Overall, most of the video sequences works well for model with video feature based on the HOTA results as shown below,

MOT20

In our experiments, for MOT20 datasets (refer table 5.4, there is no any type of video sequences gives better results for the HOTA metric for model with video feature extractor. And for all video sequences, MOT20-01,MOT20-02, MOT20-

03,MOT20-05 , the model without video feature extractor gives better results But, when we see the combined or average results also, model with feature extractor does not works well. Overall, the all type of video sequences works well for model without feature extractor based on the HOTA results mentioned in the below table.

5.1.2 Segmentation mask experiments

In our experiments, for MOT16 datasets, specifically in MOT16-04,MOT16-05,MOT16-09 video sequences, model with segmentaion masks gives better results for the HOTA metric. And for MOT16-02, MOT16-10,MOT16-11,MOT16-13 video sequaences, the model with segmentation mask not gives better results when we compare with the model without segmentation mask. But, when we see the combined or average results also, model with segmentation mask does not work well. Overall, most of the video sequences does not work well for model with segmentation mask as in the below table;

MOT16

MOT17

In our experiments, for MOT17 datasets, specifically in MOT17-04-DPM, MOT17-04-FRCNN,MOT17-04-SDP,MOT17-05-DPM,MOT17-05-FRCNN,MOT17-05-SDP,MOT17-09-DPM,MOT17-10-FRCNN video sequences, model with segmentation mask gives better results for the HOTA metric. And for MOT17-02-DPM,MOT17-02-FRCNN, MOT17-04-SDP,MOT17-09-FRCNN,MOT17-09-SDP,MOT17-10-DPM,MOT17-10-SDP,MOT17-11-DPM,MOT17-11-FRCNN,MOT17-11-SDP,MOT17-13-DPM video sequaences, the model with segmentation mask not gives better results when we compare with the model without segmentation mask. But, when we see the combined or average results, model with segmentation masks does not works well .Overall, the most of the video sequences does not works well for model with segmentation masks based on the HOTA results as in the below table.

MOT20

In our experiments, for MOT20 datasets, there is no such video sequences works well with segmentation model based on the HOTA metric results. And for all

MOT-20 video sequences, the model without segmentation mask gives better results.But, when we see the combined or average results, model with segmentation mask not gives better results when we compare with model without segmentation mask.Overall, the MOT-20 dataset works well for model without segmentation mask as in the following table:

5.2 Results obtained from the Second Research Question

We evaluated our new feature extractor, video feature extractor with our baseline model to identify the effect of video features in improving MOT performance. We modified the official implementation of deep sort algorithm and included our new feature extractor.

5.3 Results obtained from the second Research Question

5.3.1 Experiments with depth coefficient

MOT16

In our experiments, for MOT16 datasets, specifically in MOT16-02, MOT16-04,MOT16-05,MOT16-11 video sequences, model with depth feature gives better results for the HOTA metric. And for MOT16-04,MOT16-09, MOT16-10,MOT16-13 video sequences, the model with depth feature not gives better results when we compare with the model without depth feature.But, when we see the combined or average results, model with depth feature works well Overall, the equal number of video sequences works well for model with depth feature and model without depth feature respectively.

MOT17

In our experiments, for MOT17 datasets, specifically in MOT17-02-DPM, MOT17-02-FRCNN,MOT17-02-SDP,MOT17-04-DPM,MOT17-05-FRCNN,MOT17-09-DPM,MOT17-

10-DPM,MOT17-10-FRCNN,MOT17-10-SDP,MOT17-11-FRCNN,MOT17-11-SDP video sequences, model with depth feature gives better results for the HOTA metric. And for MOT17-04-FRCNN,MOT17-04-SDP,MOT17-05-DPM,MOT17-05-SDP,MOT17-09-FRCNN,MOT17-09-SDP,MOT17-11-DPM,MOT17-13-DPM video sequences, the model with depth feature not gives better results when we compare with the model without depth feature.But, when we see the combined or average results, model with depth feature does not works well Overall, most of the video sequences works well for model with depth feature for HOTA metric as shown in the below table.

MOT20

In our experiments, for MOT20 datasets, specifically in MOT20-01, MOT20-02,MOT20-03,MOT20-05 video sequences, model with depth feature gives better results for the HOTA metric. And for , there is no video sequaences, the model without depth feature gives better results.But, when we see the combined or average results, model with depth feature works well. Overall, the all types of video sequences works well for model with depth feature based on the HOTA results as shown below.

5.4 Summary

In this chapter, we presented the detailed results of each methodology we used. In addition to the findings, an overview of the BLEU scores was addressed, as well as a com- parison of various experiments with examples. Chapter 6 addresses the possible conclusions that can be taken from these findings.

Sequence	Method	HOTA↑	DetA↑	AssA↑	LocA↑
MOT17-02-DPM	Mars	16.053	9.1108	28.299	80.385
	Ours	13.949	9.1943	21.174	80.753
MOT17-02-FRCNN	Mars	32.512	28.543	37.138	88.501
	Ours	29.878	27.969	32.067	87.705
MOT17-02-SDP	Mars	33.76	37.15	30.761	84.002
	Ours	28.685	36.936	22.431	83.043
MOT17-04-DPM	Mars	26.657	20.863	34.085	83.58
	Ours	24.654	20.966	29.019	83.35
MOT17-04-FRCNN	Mars	51.832	47.735	56.405	90.54
	Ours	48.062	47.265	49.007	90.28
MOT17-04-SDP	Mars	62.12	62.738	61.565	87.9
	Ours	54.269	60.871	48.49	87.25
MOT17-05-DPM	Mars	23.37	16.237	33.665	81.462
	Ours	19.996	16.727	23.926	80.93
MOT17-05-FRCNN	Mars	45.399	39.014	52.901	84.463
	Ours	34.18	39.697	29.496	84.045
MOT17-05-SDP	Mars	51.359	50.114	53.039	86.323
	Ours	39.789	50.074	32.029	85.539
MOT17-09-DPM	Mars	29.965	27.447	32.762	79.723
	Ours	24.052	27.22	21.337	79.103
MOT17-09-FRCNN	Mars	48.71	48.321	49.119	90.525
	Ours	41.734	47.34	36.81	89.897
MOT17-09-SDP	Mars	47.927	55.346	41.593	87.97
	Ours	41.769	53.751	32.533	87.232

Table 5.2: HOTA metric for experiments on MOT17 with new feature extractor (1)

Sequence	Method	HOTA↑	DetA↑	AssA↑	LocA↑
MOT17-10-DPM	Mars	24.241	18.297	32.181	80.368
	Ours	21.646	18.299	25.688	79.993
MOT17-10-FRCNN	Mars	43.912	46.745	41.391	85.419
	Ours	38.029	46.644	31.201	85.229
MOT17-10-SDP	Mars	51.546	56.115	47.494	83.233
	Ours	40.85	55.117	30.454	82.671
MOT17-11-DPM	Mars	40.315	33.428	48.644	82.374
	Ours	32.002	33.264	30.825	81.809
MOT17-11-FRCNN	Mars	54.597	51.399	58.03	91.081
	Ours	39.911	51.044	31.222	90.648
MOT17-11-SDP	Mars	56.993	60.512	53.758	88.156
	Ours	39.598	60.101	26.141	87.409
MOT17-13-DPM	Mars	12.621	4.7848	33.38	79.043
	Ours	12.478	4.9582	31.511	78.816
MOT17-13-FRCNN	Mars	46.354	42.667	50.627	84.304
	Ours	44.183	42.162	46.658	83.854
MOT17-13-SDP	Mars	51.651	41.249	64.777	83.519
	Ours	47.401	40.706	55.367	83.142
COMBINED	Mars	44.377	38.7	51.16	86.618
	Ours	38.538	38.277	39.096	86.101

Table 5.3: HOTA metric for experiments on MOT17 with new feature extractor (2)

Sequence	Method	HOTA↑	DetA↑	AssA↑	LocA↑
MOT20-01	Mars	30.181	18.96	48.045	92.787
	Ours	27.891	19.234	40.455	92.409
MOT20-02	Mars	25.984	20.633	32.732	92.857
	Ours	23.992	20.753	27.755	92.464
MOT20-03	Mars	2.1288	0.48733	9.306	91.188
	Ours	2.0679	0.49727	8.6061	90.841
MOT20-05	Mars	1.7868	0.4072	7.8499	90.466
	Ours	1.7668	0.41899	7.4621	90.183
COMBINED	Mars	10.567	3.5276	31.66	92.627
	Ours	9.7823	3.5622	26.879	92.241

Table 5.4: HOTA metric for experiments on MOT20 with new feature extractor

Sequence	Method	HOTA↑	DetA↑	AssA↑	LocA↑
MOT16-02	w/o seg,	16.384	9.4892	28.299	80.385
	w/ seg	15.456	9.4723	25.229	80.873
MOT16-04	w/o seg,	26.657	20.863	34.085	83.58
	w/ seg	26.935	20.821	34.86	83.539
MOT16-05	w/o seg,	23.567	16.473	33.744	81.465
	w/ seg	23.754	16.322	34.591	81.41
MOT16-09	w/o seg,	30.248	27.742	33.032	79.693
	w/ seg	31.834	27.586	36.769	79.524
MOT16-10	w/o seg,	24.698	19.019	32.128	80.275
	w/ seg	24.045	18.858	30.711	80.23
MOT16-11	w/o seg,	40.89	34.294	48.777	82.381
	w/ seg	36.5	33.8	39.446	82.163
MOT16-13	w/o seg,	12.727	4.8646	33.388	79.043
	w/ seg	12.029	4.8704	29.803	79.011
COMBINED	w/o seg,	25.592	18.362	35.7	82.195
	w/ seg	25.065	18.272	34.409	82.164

Table 5.5: HOTA metric for experiments on MOT16 with fine segmentation masks

Sequence	Method	HOTA↑	DetA↑	AssA↑	LocA↑
MOT17-02-DPM	w/o seg,	16.053	9.1108	28.299	80.385
	w/ seg	15.144	9.0931	25.231	80.865
MOT17-02-FRCNN	w/o seg,	32.512	28.543	37.138	88.501
	w/ seg	31.245	28.592	34.252	88.39
MOT17-02-SDP	w/o seg,	33.76	37.15	30.761	84.002
	w/ seg	34.555	36.769	32.569	83.694
MOT17-04-DPM	w/o seg,	26.657	20.863	34.085	83.58
	w/ seg	26.935	20.821	34.86	83.539
MOT17-04-FRCNN	w/o seg,	51.832	47.735	56.405	90.54
	w/ seg	51.855	47.699	56.497	90.527
MOT17-04-SDP	w/o seg,	62.12	62.738	61.565	87.9
	w/ seg	61.872	62.926	60.89	88.001
MOT17-05-DPM	w/o seg,	23.37	16.237	33.665	81.462
	w/ seg	23.554	16.074	34.538	81.375
MOT17-05-FRCNN	w/o seg,	45.399	39.014	52.901	84.463
	w/ seg	43.769	38.934	49.257	84.469
MOT17-05-SDP	w/o seg,	51.359	50.114	53.039	86.323
	w/ seg	51.997	49.795	54.695	86.206
MOT17-09-DPM	w/o seg,	29.965	27.447	32.762	79.723
	w/ seg	31.676	27.375	36.681	79.694
MOT17-09-FRCNN	w/o seg,	48.71	48.321	49.119	90.525
	w/ seg	46.752	47.64	45.904	90.275
MOT17-09-SDP	w/o seg,	47.927	55.346	41.593	87.97
	w/ seg	47.513	54.475	41.49	87.862
MOT17-10-DPM	w/o seg,	24.241	18.297	32.181	80.368
	w/ seg	23.588	18.14	30.728	80.321
MOT17-10-FRCNN	w/o seg,	43.912	46.745	41.391	85.419
	w/ seg	43.967	46.7	41.535	85.393
MOT17-10-SDP	w/o seg,	51.546	56.115	47.494	83.233
	w/ seg	50.865	56.025	46.301	83.22
MOT17-11-DPM	w/o seg,	40.315	33.428	48.644	82.374
	w/ seg	35.983	32.986	39.284	82.179
MOT17-11-FRCNN	w/o seg,	54.597	51.399	58.03	91.081
	w/ seg	47.866	51.442	44.552	91.09
MOT17-11-SDP	w/o seg,	56.993	60.512	53.758	88.156
	w/ seg	54.308	60.043	49.17	88.076
MOT17-13-DPM	w/o seg,	12.621	4.7848	33.38	79.043
	w/ seg	11.928	4.7905	29.795	79.011

Sequence	Method	HOTA↑	DetA↑	AssA↑	LocA↑
MOT20-01	w/o seg,	30.181	18.96	48.045	92.787
	w/ seg	29.108	18.784	45.11	92.794
MOT20-02	w/o seg,	25.984	20.633	32.732	92.857
	w/ seg	25.778	20.575	32.307	92.833
MOT20-03	w/o seg,	2.1288	0.48733	9.306	91.188
	w/ seg	2.1177	0.48436	9.2654	91.186
MOT20-05	w/o seg,	1.7868	0.4072	7.8499	90.466
	w/ seg	1.7851	0.40625	7.8527	90.497
COMBINED	w/o seg,	10.567	3.5276	31.66	92.627
	w/ seg	10.444	3.5154	31.034	92.611

Table 5.7: HOTA metric for experiments on MOT20 with fine segmentation masks

Sequence	Method	HOTA↑	DetA↑	AssA↑	LocA↑
MOT16-02	w/o depth	14.502	9.3878	22.421	80.152
	w/ depth	14.618	9.4371	22.661	80.145
MOT16-04	w/o depth	24.886	20.867	29.723	83.265
	w/ depth	25.292	20.939	30.568	83.35
MOT16-05	w/o depth	18.995	17.11	21.135	81.018
	w/ depth	19.174	17.136	21.489	81.139
MOT16-09	w/o depth	25.167	27.797	22.899	79.236
	w/ depth	23.372	27.222	20.158	78.858
MOT16-10	w/o depth	22.985	19.175	27.624	79.962
	w/ depth	22.938	19.261	27.388	80.043
MOT16-11	w/o depth	31.491	34.093	29.142	81.771
	w/ depth	32.138	34.101	30.316	81.826
MOT16-13	w/o depth	12.557	5.0394	31.395	78.801
	w/ depth	12.557	5.0394	31.395	78.801
COMBINED	w/o depth	22.67	18.422	27.962	81.819
	w/ depth	22.855	18.445	28.359	81.86

Table 5.8: HOTA metric for experiments on MOT16 with new depth feature

Sequence	Method	HOTA↑	DetA↑	AssA↑	LocA↑
MOT17-02-DPM	w/o depth	13.949	9.1943	21.174	80.753
	w/ depth	14.613	9.0763	23.543	80.317
MOT17-02-FRCNN	w/o depth	29.878	27.969	32.067	87.705
	w/ depth	30.458	28.247	32.981	87.955
MOT17-02-SDP	w/o depth	28.685	36.936	22.431	83.043
	w/ depth	31.589	36.952	27.107	82.946
MOT17-04-DPM	w/o depth	24.654	20.966	29.019	83.35
	w/ depth	24.714	20.907	29.236	83.322
MOT17-04-FRCNN	w/o depth	48.062	47.265	49.007	90.28
	w/ depth	46.596	47.41	45.943	90.331
MOT17-04-SDP	w/o depth	54.269	60.871	48.49	87.25
	w/ depth	53.793	60.977	47.551	87.28
MOT17-05-DPM	w/o depth	19.996	16.727	23.926	80.93
	w/ depth	18.829	16.964	20.931	81.109
MOT17-05-FRCNN	w/o depth	34.18	39.697	29.496	84.045
	w/ depth	34.811	39.477	30.717	83.954
MOT17-05-SDP	w/o depth	39.789	50.074	32.029	85.539
	w/ depth	39.178	50.486	30.86	85.684
MOT17-09-DPM	w/o depth	24.052	27.22	21.337	79.103
	w/ depth	25.122	27.195	23.253	79.138
MOT17-09-FRCNN	w/o depth	41.734	47.34	36.81	89.897
	w/ depth	38.626	47.738	31.273	89.975
MOT17-09-SDP	w/o depth	41.769	53.751	32.533	87.232
	w/ depth	39.588	53.361	29.489	86.923
MOT17-10-DPM	w/o depth	21.646	18.299	25.688	79.993
	w/ depth	21.739	18.275	25.939	79.994
MOT17-10-FRCNN	w/o depth	38.029	46.644	31.201	85.229
	w/ depth	39.389	46.59	33.475	85.181
MOT17-10-SDP	w/o depth	40.85	55.117	30.454	82.671
	w/ depth	46.512	55.097	39.452	82.642
MOT17-11-DPM	w/o depth	32.002	33.264	30.825	81.809
	w/ depth	30.49	33.209	28.039	81.831
MOT17-11-FRCNN	w/o depth	39.911	51.044	31.222	90.648
	w/ depth	40.675	50.833	32.563	90.647
MOT17-11-SDP	w/o depth	39.598	60.101	26.141	87.409
	w/ depth	42.559	59.776	30.351	87.319
MOT17-13-DPM	w/o depth	12.478	4.9582	31.511	78.816
	w/ depth	12.452	4.9569	31.385	78.801

Sequence	**Method**	**HOTA↑**	**DetA↑**	**AssA↑**	**LocA↑**
MOT20-01	w/o depth	27.891	19.234	40.455	92.409
	w/ depth	28.144	19.209	41.247	92.39
MOT20-02	w/o depth	23.992	20.753	27.755	92.464
	w/ depth	24.018	20.734	27.84	92.451
MOT20-03	w/o depth	2.0679	0.49727	8.6061	90.841
	w/ depth	2.07	0.49584	8.6475	90.881
MOT20-05	w/o depth	1.7668	0.41899	7.4621	90.183
	w/ depth	1.7669	0.41927	7.4574	90.182
COMBINED	w/o depth	9.7823	3.5622	26.879	92.241
	w/ depth	9.8044	3.5593	27.022	92.231

Table 5.10: HOTA metric for experiments on MOT20 with new depth feature

Chapter 6 - Conclusions

6.1 Introduction

The aim of this dissertation is to explore cross domain models to improve MOT performance, such as human action recognition and zero-shot monocular depth estimation. This chapter gives an overview of the findings reached as a result of our overall study efforts.

6.2 Conclusions about research problem and research questions

The first research question in subsection 1.2.2 is,"How can we improve discrimination of humans using video features from video classifiers in MOT?". The methods have been discussed thoroughly with results in Chapter 3 and Chapter 5 respectively .

In the MOT domain, for feature extraction image feature extractors are used which are trained in static images. When we are using video features as in our methodology we could see improvement in detection accuracy in most of our experiments in Chapter 5. Mainly in short videos like MOT16(Milan, Leal-Taixé, Reid, Roth & Schindler 2016) we could see consistent improvement throughout the videos. But in longer videos like MOT17(Milan, Leal-Taixé, Reid, Roth & Schindler 2016) and MOT20(Dendorfer et al. 2020) our approach didn't work as expected. Even though there is improvement in detection accuracy, in all these experiments overall performance, HOTA score slightly decreases.

If we consider our second research question, "How to enhance the occlusion handling using depth information during human tracking? " We extracted the depth information and the depth cues are fused with appearance features using our new coefficient, depth coefficient. When we use this coefficient, we observed overall improvement in two datasets, MOT16(Milan, Leal-Taixé, Reid, Roth & Schindler 2016)and MOT20(Dendorfer et al. 2020) gives a positive indication of

usage of depth cues in MOT for the improvement. We coudld the improvement in all three HOTA, DetA and AssA in above two datasets. Therefore, it concludes that the depth information can be utilized to advance MOT.

6.3 Limitations

Main limitation of the our approach is, as we are using the off-the-shelf models we needs very intensive strategy to utilize the potential of cross-domain models. Because when we working in log videos our approaches slightly under perform.

6.4 Future works

With that, as a preliminary research we only used one algorithm to understand our methodology by reducing additional complexity. But if we explored the other MOT algorithms we could get more clear understanding of our approaches.

We could further push the limits of these approaches by involving parameter training as these features are entirely new for the domain. But as these trainings are resource and time intensive, my be in the future introduction of some optimization strategies could make these training feasible and could help to improve our approaches further.

Bibliography

Ahmed Bhuiyan, R., Ahmed, N., Amiruzzaman, M. & Islam, M. R. (2020), 'A robust feature extraction model for human activity characterization using 3-axis accelerometer and gyroscope data', *Sensors* **20**(23), 6990.

Ahmed, M., Ramzan, M., Khan, H. U., Iqbal, S., Khan, M. A., Choi, J.-I., Nam, Y. & Kadry, S. (2021), 'Real-time violent action recognition using key frames extraction and deep learning'.

Ali, S. & Shah, M. (2015), 'Multi-object tracking: a literature survey', *ACM Computing Surveys (CSUR)* **47**(4), 1–34.

Alimi, A. M., Ye, L., Nourani, M. & Azari, M. M. (2021), 'A survey of deep learning-based object detection and tracking techniques for self-driving cars', *Sensors* **21**(3).

Bertasius, G., Wang, H. & Torresani, L. (2021), Is space-time attention all you need for video understanding?, *in* 'ICML', Vol. 2, p. 4.

Brown, T., Mann, B., Ryder, N., Subbiah, M., Kaplan, J. D., Dhariwal, P., Neelakantan, A., Shyam, P., Sastry, G., Askell, A. et al. (2020), 'Language models are few-shot learners', *Advances in neural information processing systems* **33**, 1877–1901.

Buehner, M. J. (2015), 'Awareness of voluntary and involuntary causal actions and their outcomes.', *Psychology of Consciousness: Theory, Research, and Practice* **2**(3), 237.

Chen, X., Xu, L., Cao, M., Zhang, T., Shang, Z. & Zhang, L. (2021), 'Design and implementation of human-computer interaction systems based on transfer support vector machine and eeg signal for depression patients' emotion recognition', *Journal of Medical Imaging and Health Informatics* **11**(3), 948–954.

Cuturi, M. & Blondel, M. (2018), 'Soft-dtw: a differentiable loss function for time-series', *arXiv preprint arXiv:1811.05381* .

Danelljan, M., Bhat, G., Khan, F. S. & Felsberg, M. (2017), Eco: Efficient convolution operators for tracking, *in* 'Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition', IEEE, pp. 6931–6939.

Del Giorno, J., Yin, Z., Tasci, C. & Manjunath, B. (2013), 'Tracking ice hockey players with kernelized correlation filters and graph cuts', *IEEE Transactions on Circuits and Systems for Video Technology* **24**(3), 421–430.

Dendorfer, P., Rezatofighi, H., Milan, A., Shi, J., Cremers, D. & Reid, I. (2019), Cvpr 2019: The 5th Visual Object Tracking Challenge, *in* 'IEEE Conference on Computer Vision and Pattern Recognition (CVPR)'.

Dendorfer, P., Rezatofighi, H., Milan, A., Shi, J., Cremers, D., Reid, I. & Roth, S. (2019), 'The MOTchallenge 2019: A benchmark for multiple object tracking', *arXiv preprint arXiv:1906.04567* .

Dendorfer, P., Rezatofighi, H., Milan, A., Shi, J., Cremers, D., Reid, I., Roth, S., Schindler, K. & Leal-Taixé, L. (2020), 'Mot20: A benchmark for multi object tracking in crowded scenes', *arXiv preprint arXiv:2003.09003* .

Devlin, J., Chang, M.-W., Lee, K. & Toutanova, K. (2018), 'Bert: Pre-training of deep bidirectional transformers for language understanding', *arXiv preprint arXiv:1810.04805* .

Dosovitskiy, A., Beyer, L., Kolesnikov, A., Weissenborn, D., Zhai, X., Unterthiner, T., Dehghani, M., Minderer, M., Heigold, G., Gelly, S. et al. (2020), 'An image is worth 16x16 words: Transformers for image recognition at scale', *arXiv preprint arXiv:2010.11929* .

Fan, H., Xiong, B., Mangalam, K., Li, Y., Yan, Z., Malik, J. & Feichtenhofer, C. (2021), Multiscale vision transformers, *in* 'Proceedings of the IEEE/CVF international conference on computer vision', pp. 6824–6835.

Farnoosh, A., Wang, Z., Zhu, S. & Ostadabbas, S. (2021), 'A bayesian dynamical approach for human action recognition', *Sensors* **21**(16), 5613.

Feichtenhofer, C., Li, Y., He, K. et al. (2022), 'Masked autoencoders as spatiotemporal learners', *Advances in neural information processing systems* **35**, 35946–35958.

Hassaballah, M. & Awad, A. I. (2020), *Deep learning in computer vision: principles and applications*, CRC Press.

Hassaballah, M. & Hosny, K. M. (2019), 'Recent advances in computer vision', *Studies in computational intelligence* **804**, 1–84.

Hassan, M. M., Uddin, M. Z., Mohamed, A. & Almogren, A. (2018), 'A robust human activity recognition system using smartphone sensors and deep learning', *Future Generation Computer Systems* **81**, 307–313.

He, K., Chen, X., Xie, S., Li, Y., Dollár, P. & Girshick, R. (2022), Masked autoencoders are scalable vision learners, *in* 'Proceedings of the IEEE/CVF conference on computer vision and pattern recognition', pp. 16000–16009.

Hermansky, H. (2006), 'Data-driven extraction of temporal features from speech', *Dynamics of Speech Production and Perception* **374**, 207.

Im, W., Kim, T.-K. & Yoon, S.-E. (2020), Unsupervised learning of optical flow with deep feature similarity, *in* 'Computer Vision–ECCV 2020: 16th European Conference, Glasgow, UK, August 23–28, 2020, Proceedings, Part XXIV 16', Springer, pp. 172–188.

Jalal, A., Kamal, S. & Kim, D. (2017), 'A depth video-based human detection and activity recognition using multi-features and embedded hidden markov models for health care monitoring systems'.

Joshi, A. B., Kumar, D., Gaffar, A. & Mishra, D. (2020), 'Triple color image encryption based on 2d multiple parameter fractional discrete fourier transform and 3d arnold transform', *Optics and Lasers in Engineering* **133**, 106139.

Khan, M. A., Javed, K., Khan, S. A., Saba, T., Habib, U., Khan, J. A. & Abbasi, A. A. (2024), 'Human action recognition using fusion of multiview and deep features: an application to video surveillance', *Multimedia tools and applications* **83**(5), 14885–14911.

Khan, M. A., Sharif, M., Akram, T., Raza, M., Saba, T. & Rehman, A. (2020), 'Hand-crafted and deep convolutional neural network features fusion and selection strategy: an application to intelligent human action recognition', *Applied Soft Computing* **87**, 105986.

Khan, M. A., Zhang, Y.-D., Khan, S. A., Attique, M., Rehman, A. & Seo, S. (2021), 'A resource conscious human action recognition framework using 26-layered deep convolutional neural network', *Multimedia Tools and Applications* **80**, 35827–35849.

Kolekar, M. H. & Dash, D. P. (2016), Hidden markov model based human activity recognition using shape and optical flow based features, *in* '2016 IEEE Region 10 Conference (TENCON)', IEEE, pp. 393–397.

Krzeszowski, T., Przednowek, K., Wiktorowicz, K. & Iskra, J. (2019), The application of multiview human body tracking on the example of hurdle clearance, *in* 'Sport Science Research and Technology Support: 4th and 5th International Congress, icSPORTS 2016, Porto, Portugal, November 7-9, 2016, and icSPORTS 2017, Funchal, Madeira, Portugal, October 30-31, 2017, Revised Selected Papers 4', Springer, pp. 116–127.

Kuhn, H. W. (1955), 'The hungarian method for the assignment problem', *Naval Research Logistics Quarterly* **2**(1), 83–97.

Leal-Taixé, L., Milan, A., Schindler, K. & Cremers, D. (2015), MOTChallenge 2015: Towards a benchmark for multi-target tracking, *in* 'IEEE International Conference on Computer Vision (ICCV)'.

LeCun, Y., Boser, B., Denker, J., Henderson, D., Howard, R., Hubbard, W. & Jackel, L. (1989), Handwritten digit recognition with a back-propagation network, *in* D. Touretzky, ed., 'Advances in Neural Information Processing Systems', Vol. 2, Morgan-Kaufmann.

Li, C., Xie, C., Zhang, B., Han, J., Zhen, X. & Chen, J. (2021), 'Memory attention networks for skeleton-based action recognition', *IEEE Transactions on Neural Networks and Learning Systems* **33**(9), 4800–4814.

Liu, C., Wang, R., Shan, S. & Chen, X. (2019), 'Trajectory clustering using dynamic time warping for multi-agent sports tracking', *IEEE Transactions on Image Processing* **28**(3), 1253–1263.

Liu, D., Xu, H., Wang, J., Lu, Y., Kong, J. & Qi, M. (2021), 'Adaptive attention memory graph convolutional networks for skeleton-based action recognition', *Sensors* **21**(20), 6761.

Luiten, J., Osep, A., Dendorfer, P., Torr, P., Geiger, A., Leal-Taixé, L. & Leibe, B. (2021), 'Hota: A higher order metric for evaluating multi-object tracking', *International journal of computer vision* **129**, 548–578.

Luo, Z. & Li, Y. (2018), 'Sports performance analysis using computer vision techniques: A review', *Journal of Imaging* **4**(9), 111.

Melhart, D., Liapis, A. & Yannakakis, G. N. (2022), 'The arousal video game annotation (again) dataset', *IEEE Transactions on Affective Computing* **13**(4), 2171–2184.

Milan, A., Heng, C. & Sminchisescu, C. (2016), Mot16: A comprehensive benchmark for multiple Object Tracking, *in* 'IEEE Conference on Computer Vision and Pattern Recognition (CVPR)'.

Milan, A., Leal-Taixé, L., Reid, I., Roth, S. & Schindler, K. (2016), Mot16: A benchmark for multi-object tracking, *in* 'Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops', pp. –.

Milan, A., Leal-Taixé, L., Reid, I., Roth, S. & Schindler, K. (2017), 'Mot16: A benchmark for multi-object tracking', *IEEE Transactions on Pattern Analysis and Machine Intelligence (PAMI)* **39**(8), 1614–1620.

Milan, A., Leal-Taixé, L., Reid, I., Roth, S. & Schindler, K. (2016), Mot16: A benchmark for multi-object tracking, *in* 'Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition'.

Milan, A., Schindler, K. & Roth, S. (2018), Roadmap for MOTChallenge 2018, *in* 'European Conference on Computer Vision (ECCV) Workshops'.

Mishra, O., Kavimandan, P. S., Tripathi, M., Kapoor, R. & Yadav, K. (2021), Human action recognition using a new hybrid descriptor, *in* 'Advances in VLSI, Communication, and Signal Processing: Select Proceedings of VCAS 2019', Springer, pp. 527–536.

Muhammad, K., Ullah, A., Imran, A. S., Sajjad, M., Kiran, M. S., Sannino, G., de Albuquerque, V. H. C. et al. (2021), 'Human action recognition using attention based lstm network with dilated cnn features', *Future Generation Computer Systems* **125**, 820–830.

Papadimitriou, I., Bekiaris-Liberis, N., Lytrivis, P., Wang, K. & Papageorgiou, M. (2015), 'Autonomous driving beyond urban scenarios: State of the art and future prospects', *IEEE Transactions on Intelligent Transportation Systems* **17**(3), 688–699.

Pfister, S., Westling, E., Mjolsness, E. & Black, M. J. (2014), Optical flow for improved football player tracking, *in* 'Computer Vision–ECCV 2014', Springer, pp. 456–471.

Ranftl, R., Lasinger, K., Hafner, D., Schindler, K. & Koltun, V. (2020), 'Towards robust monocular depth estimation: Mixing datasets for zero-shot cross-dataset transfer', *IEEE transactions on pattern analysis and machine intelligence* **44**(3), 1623–1637.

Shaham, T. (2010), 'Tracking in medical imaging', *Annual review of biomedical engineering* **12**(1), 567–592.

Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A. N., Kaiser, Ł. & Polosukhin, I. (2017), 'Attention is all you need', *Advances in neural information processing systems* **30**.

Voulodimos, A., Doulamis, N., Doulamis, A. & Protopapadakis, E. (2018), 'Deep learning for computer vision: A brief review', *Computational intelligence and neuroscience* **2018**.

Wang, J., Cao, D., Wang, J. & Liu, C. (2021), 'Action recognition of lower

limbs based on surface electromyography weighted feature method', *Sensors* **21**(18), 6147.

Wei, C., Fan, H., Xie, S., Wu, C.-Y., Yuille, A. & Feichtenhofer, C. (2022), Masked feature prediction for self-supervised visual pre-training, *in* 'Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition', pp. 14668–14678.

Wojke, N., Bewley, A. & Paulus, D. (2017), Simple online and realtime tracking with a deep association metric, *in* '2017 IEEE international conference on image processing (ICIP)', IEEE, pp. 3645–3649.

Xia, K., Huang, J. & Wang, H. (2020), 'Lstm-cnn architecture for human activity recognition', *IEEE Access* **8**, 56855–56866.

Zagoruyko, S. & Komodakis, N. (2016), 'Wide residual networks', *arXiv preprint arXiv:1605.07146* .

Zhao, B., Li, S., Gao, Y., Li, C. & Li, W. (2020), 'A framework of combining short-term spatial/frequency feature extraction and long-term indrnn for activity recognition', *Sensors* **20**(23), 6984.

Zheng, L., Bie, Z., Sun, Y., Wang, J., Su, C., Wang, S. & Tian, Q. (2016), Mars: A video benchmark for large-scale person re-identification, *in* 'Computer Vision–ECCV 2016: 14th European Conference, Amsterdam, The Netherlands, October 11-14, 2016, Proceedings, Part VI 14', Springer, pp. 868–884.

Zin, T. T., Htet, Y., Akagi, Y., Tamura, H., Kondo, K., Araki, S. & Chosa, E. (2021), 'Real-time action recognition system for elderly people using stereo depth camera', *Sensors* **21**(17), 5895.