

Impact of Image Enhancement on Multi-Object Tracking in Underwater Scenario

by

Rahul Kumar
202111003

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

MASTER OF TECHNOLOGY
in
INFORMATION AND COMMUNICATION TECHNOLOGY
to

DHIRUBHAI AMBANI INSTITUTE OF INFORMATION AND COMMUNICATION TECHNOLOGY

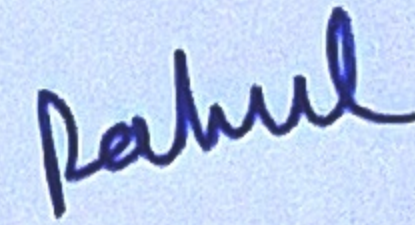


June, 2023

Declaration

I hereby declare that

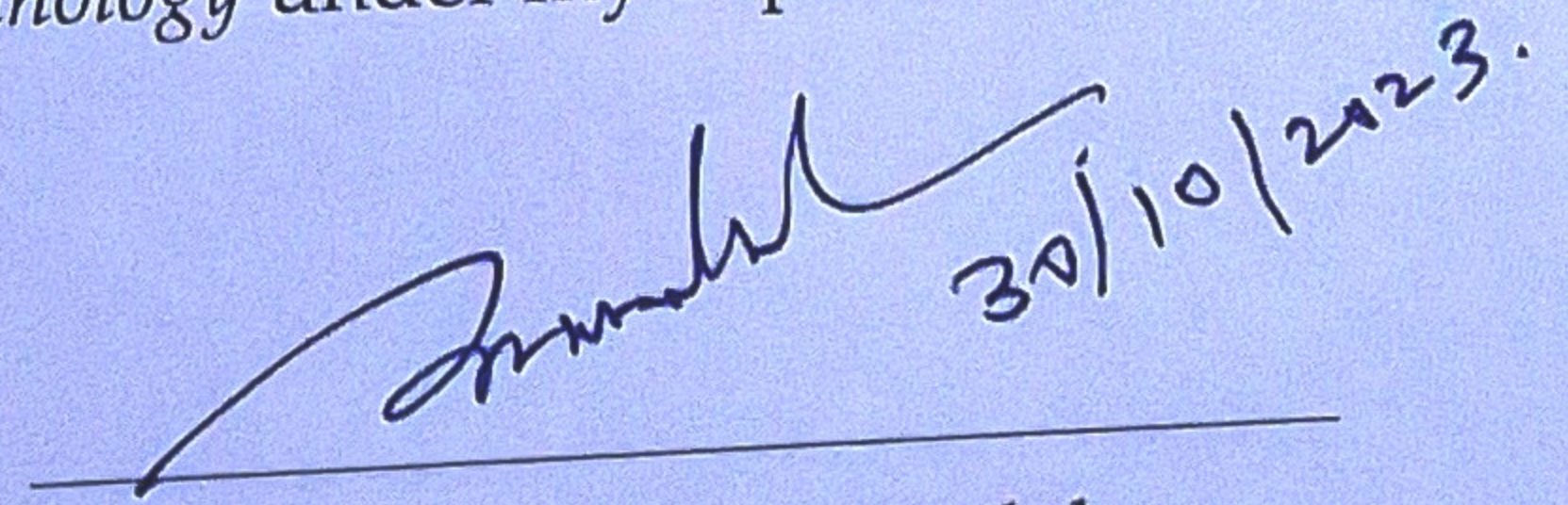
- i) The thesis comprises of my original work towards the degree of Master of Technology in Information and Communication Technology at Dhirubhai Ambani Institute of Information and Communication Technology and has not been submitted elsewhere for a degree,
- ii) Due acknowledgment has been made in the text to all the reference material used.



Rahul Kumar

Certificate

This is to certify that the thesis work on Impact of Enhancement of Images in Multi Object tracking has been carried out by Rahul Kumar for the degree of Master of Technology in Information and Communication Technology at *Dhirubhai Ambani Institute of Information and Communication Technology* under my supervision.

 30/10/2023.

Dr. Srimanta Mandal
Thesis Supervisor

Acknowledgments

I am extremely grateful to my supervisor, Dr. Srimanta Mandal, for their invaluable advice, continuous support, and patience during my MTech Thesis. Their immense knowledge and ample experience have encouraged me during my academic research.

I want to thank my peers and friends Gaurav Jha and Jeegar Patel who were always there in my ups and downs. Their kind help and support have made my study and life at DA-IICT a wonderful and memorable experience.

Finally, I would like to express my gratitude to my parents and family for their love, care, and invaluable support throughout my life.

Contents

Abstract	v
List of Principal Symbols and Acronyms	v
List of Tables	vi
List of Figures	vii
1 Introduction	1
1.1 Objectives	1
1.2 Contributions	1
1.3 Constrained Environment	3
1.4 Unconstrained Environment	5
1.5 Organization	6
2 Literature Survey and Background	8
2.1 Background	8
2.1.1 Enhancement Methods	8
2.1.2 Siamese Network	9
2.1.3 LSTM	14
2.1.4 Vision Transformer	15
2.1.5 Traditional Methods	16
2.1.6 Detection-Based Methods	16
2.1.7 Deep Learning-Based Methods	17
2.1.8 Graph-Based Methods	17
2.1.9 Datasets and Evaluation Metrics	17
2.1.10 Traditional Methods for Multi-Object Tracking	17
2.1.11 Transformer-Based Approaches in Computer Vision	18
2.1.12 TrackFormer: A Transformer-Based Approach to Multi-Object Tracking	18
2.1.13 Datasets and Evaluation Metrics for Multi-Object Tracking .	18

2.1.14	Comparative Analysis of Transformer-Based Approaches . .	19
2.1.15	Challenges and Future Directions	19
2.1.16	Challenges and Open Problems	19
3	Investigated Methodology	20
3.1	Dataset Details	20
3.2	Method	21
3.3	Appearance Similarity	23
3.4	Motion Similarity	25
3.5	Spatial Similarity	26
3.6	Joint Optimization	27
4	Experiments and Results	28
4.1	Enhancement Method used for enhancement	36
4.1.1	Quantitative Results	38
5	Conclusion and Future Works	40
5.1	Future Works	40
	References	42

Abstract

This thesis examines the impact of image enhancement techniques on multi-object tracking (MOT) performance using four deep learning models: Long Short-Term Memory (LSTM), Vision Transformer, Siamese Network, and Convolutional Neural Network (CNN). The objective is to assess the effectiveness of these models in handling challenging visual conditions and explore the benefits of image pre-processing techniques for improving tracking accuracy.

The study utilizes various image enhancement approaches, including denoising, deblurring, and super-resolution. Each deep learning model is implemented and trained on a large-scale dataset specifically designed for multi-object tracking. Performance evaluation is conducted on benchmark datasets, comparing the tracking accuracy of the base models with and without image enhancement techniques. Evaluation metrics such as average precision, recall, tracking consistency, and computational efficiency are considered.

The results demonstrate that image enhancement techniques have a significant positive impact on multi-object tracking performance across all four models. LSTM, known for capturing temporal dependencies, exhibits improved tracking accuracy when combined with image enhancement. Vision Transformer, which utilizes self-attention mechanisms, benefits from enhanced image quality, resulting in superior performance in challenging visual conditions. Siamese Networks and CNN also show enhanced tracking capabilities when integrated with image enhancement techniques.

Index Terms: *Vision Transformer, Convolution Neural Network, Long Short term Memory, Siamese Network*

List of Tables

4.1 Comparison of parameters on fish4knowledge dataset 35

List of Figures

1.1	Various types of environment a) Constrained b) Unconstrained environment	3
1.2	Constrained Environment inside the aquarium	4
1.3	Unconstrained scenario.	6
2.1	Effects of Enhancement	10
2.2	Siamese Network	13
2.3	Siamese Network for predicting whether the two images is positive pair or negative	13
2.4	Vision Transformer	16
3.1	An example normal fish trajectory with detections	21
3.2	DFTnet Model	22
4.1	Impact of Enhancement on Images	29
4.2	Ground-Truth Trajectories and Trajectories generated by (a) DFT-Net , (b) V-IOU Tracker, (c) IOU Tracker, (d) DeepSORT, and (e) MDP Lacker.	30
4.3	Odd rows shows the bounding boxes without enhancement and even rows shows the trajectory with enhancement in frame t , $t + 1$, $t + 2$, $t + 3$ and $t+4$ of videos of Fish4knowledge dataset	32
4.4	Tracking results on Fish4Knowledge dataset using enhancement	34
4.5	Histogram eqlisation impacts	37

CHAPTER 1

Introduction

1.1 Objectives

The objectives of this thesis are as follows:

1. Analyze how image enhancements impact model accuracy and robustness.
2. Optimize existing models by tuning key parameters to enhance accuracy, speed, and efficiency in classification, object detection, and segmentation tasks.
3. Explore novel models to advance the field, evaluating their advantages, limitations, and suitability for specific tasks, aiming to enhance the state-of-the-art.

1.2 Contributions

The main contributions of this thesis are as follows:

1. Investigating the impact and performance of various multi-tracking models on enhanced images. This contribution involves conducting a comprehensive evaluation of multiple multi-tracking models, such as DeepSORT, SORT, or Tracktor, on images that have undergone enhancement techniques. By systematically analyzing the results and comparing them with the performance on non-enhanced images, this contribution provides insights into the effectiveness of different models in handling enhanced images for the task of object tracking.
2. Hyperparameter tuning of various multi-tracking models for fish tracking. This contribution focuses on fine-tuning the hyperparameters of different

multi-tracking models specifically for the task of fish tracking. By systematically exploring and adjusting key hyperparameters, such as detection threshold, appearance model, or motion model, this contribution aims to optimize the performance of the multi-tracking models for accurately tracking fish in enhanced images. The results of this tuning process contribute to improving the overall accuracy, robustness, and efficiency of the multi-tracking models in the context of fish tracking.

The ocean, as stated by the National Oceanic and Atmospheric Administration (NOAA), is a vital component of our planet, covering more than 70% of Earth's surface and playing a crucial role in weather patterns, temperature regulation, and sustaining all forms of life. Unfortunately, the marine environment has suffered significant impacts due to human activities. This has led to an increased focus on the study of marine ecosystems across various disciplines. Tracking marine animals is an essential part of monitoring environmental effects and understanding the intricate dynamics of these ecosystems.

Observing and tracking marine animals, particularly fishes, provide valuable insights that support marine life investigations. These investigations encompass a wide range of studies, including gathering species-specific statistics, precise measurements of fish characteristics, analyzing group behavior, studying distribution patterns, and examining mobility patterns. By understanding these aspects, we can better comprehend how fishes are likely to respond to changes in their environment.

In addition to its scientific significance, tracking marine animals offers numerous benefits in commercial applications such as fish farming and fisheries management. The observations and data obtained through tracking contribute to optimizing fish farming practices, improving productivity, and ensuring the sustainable management of fish populations. Understanding fish behavior and their movements assists stakeholders in making informed decisions related to fishing practices, resource allocation, and conservation efforts.

Fish behavior analysis is a critical aspect of tracking marine animals and serves as a foundation for various high-end applications. By closely observing fish behavior, researchers gain valuable insights into their responses to environmental changes, population dynamics, and the overall health of marine ecosystems. This knowledge is essential for implementing effective conservation strategies, design-

ing marine protected areas, and assessing the ecological impact of human activities. [8]

In conclusion, the study of marine ecosystems has gained significant attention due to the critical role played by the ocean in sustaining life on Earth. Tracking marine animals, particularly fishes, is an essential tool for monitoring environmental effects, understanding fish behavior, and supporting marine life investigations. Furthermore, tracking has practical implications in commercial sectors such as fish farming and fisheries management. By combining scientific research and commercial applications, we can work towards the preservation and sustainable management of our precious marine resources.

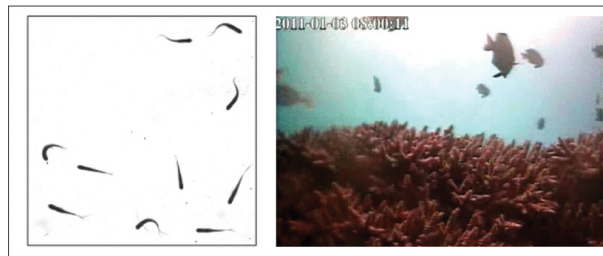


Figure 1.1: Various types of environment a) Constrained b) Unconstrained environment

1.3 Constrained Environment

In Figure 1.2, an unconstrained environment is depicted, illustrating the contrast with laboratory settings. The majority of fish tracking research is conducted within laboratory settings, where specific conditions can be controlled to facilitate accurate observations. In these controlled environments, cameras are often mounted above aquariums, allowing for optimal tracking and monitoring of the fishes' movements. Furthermore, these laboratory settings typically involve a limited number of fishes, enabling researchers to focus on individual or small group behavior analysis.

By conducting fish tracking studies in the laboratory, researchers gain several advantages. The overhead camera placement provides a top-down view, allowing for comprehensive coverage and minimizing occlusion issues that may occur with other camera angles. This perspective enables precise tracking of fish trajectories and the collection of detailed data on their swimming patterns, interactions, and other behavioral characteristics.

Moreover, laboratory settings offer the opportunity to control various environmental parameters. Factors such as water temperature, light intensity, water quality, and even the presence of specific stimuli can be manipulated to create consistent and reproducible experimental conditions. This control enhances the reliability and validity of the findings, as researchers can isolate and examine the effects of specific variables on fish behavior.

The use of a limited number of fishes in these laboratory setups provides researchers with a focused context for analysis. With a smaller population, individual fish can be tracked more accurately, and their behavior can be studied in detail. This controlled setting allows for the identification of specific patterns, preferences, or responses to stimuli, aiding in the understanding of fish behavior and its underlying mechanisms.

It is important to note that while laboratory settings offer controlled conditions, there are limitations to consider. The artificial environment may not fully replicate the complexities and dynamics of fish behavior in their natural habitats. Therefore, findings from laboratory-based research should be complemented with studies conducted in more ecologically relevant settings to ensure a comprehensive understanding of fish behavior.

In conclusion, laboratory settings provide valuable advantages for fish tracking research. The overhead camera placement, limited number of fishes, and controlled environmental parameters allow for precise tracking, detailed behavioral analysis, and the isolation of specific variables. However, it is essential to balance laboratory studies with field-based research to capture the full range of fish behaviors in natural habitats. The combination of controlled laboratory investigations and ecologically relevant observations contributes to a more holistic understanding of fish behavior and its ecological significance.



Figure 1.2: Constrained Environment inside the aquarium

1.4 Unconstrained Environment

Figure 1.3 illustrates a constrained laboratory environment, contrasting with real-world settings where fish tracking faces increased complexities. While a significant portion of video-based fish tracking research is carried out within constrained laboratory settings, the practical or commercial applications of fish tracking often take place in environments characterized by high uncertainty and complexity. The approaches discussed in Section II-A, which are designed for constrained environments, prove to be ineffective in such scenarios. As a result, researchers have begun focusing on the challenging task of tracking fishes in real-time videos captured in more realistic settings. [8]

Recognizing the limitations of existing approaches, efforts have been made to develop novel methods specifically tailored for tracking fishes in real-world conditions. These methods aim to overcome the complexities arising from factors such as varying lighting conditions, underwater disturbances, occlusions, and the unpredictable nature of fish behavior in natural habitats. By addressing these challenges, researchers strive to achieve accurate and reliable fish tracking results in practical applications.

To gather data in more realistic environments, researchers have undertaken data collection efforts using underwater video-surveillance cameras deployed near the Ken-Ding subtropical coral reef waters managed by Eco Grid in Taiwan. These subtropical coral reef waters serve as an ideal site for studying fish behavior due to their ecological significance and rich biodiversity. The data collected from these underwater video-surveillance cameras provide valuable insights into the dynamics of fish populations, their interactions, and responses to environmental changes.

Tracking fishes in real-time videos obtained from such ecologically relevant settings is crucial for various applications. For instance, it enables the monitoring and assessment of marine ecosystems, aiding in conservation efforts and understanding the impacts of human activities on fish populations. Real-time fish tracking is also valuable in fields such as fisheries management, where it assists in assessing stock levels, implementing sustainable fishing practices, and making informed decisions to ensure the long-term viability of fish populations.

However, it is important to acknowledge the inherent complexities associated with tracking fishes in real-time videos. The unpredictable nature of fish behavior, coupled with the challenging underwater conditions, presents significant technical and algorithmic hurdles. As a result, researchers continue to explore inno-

vative approaches, leveraging advancements in computer vision, machine learning, and deep learning techniques to improve the accuracy and robustness of fish tracking algorithms in practical applications. [16]

In conclusion, while a substantial portion of fish tracking research has been conducted in constrained laboratory settings, efforts are being made to tackle the challenges of tracking fishes in more complex and realistic environments. Researchers are collecting valuable data using underwater video-surveillance cameras in ecologically significant locations, such as the Ken-Ding subtropical coral reef waters managed by EcoGrid in Taiwan. This data, along with advancements in tracking algorithms, holds promise for addressing real-world fish tracking applications in areas such as marine ecosystem monitoring and fisheries management.



Figure 1.3: Unconstrained scenario.

1.5 Organization

The remainder of this thesis is organized as follows:

1. Chapter 2: Literature Survey and Backgrounds
 - This chapter provides a comprehensive literature survey and background information on the research topic. It reviews relevant studies, theories, and methodologies related to multi-tracking and fish tracking. The chapter aims to establish a solid foundation of knowledge, highlight

existing gaps in the field, and identify key challenges and opportunities for further research.

2. Chapter 3: Investigated Method

- This chapter presents the investigated method for multi-tracking and fish tracking. It outlines the proposed approach, algorithms, and techniques used in the research. The chapter details the methodology, including data collection, preprocessing, feature extraction, and the application of multi-tracking models. It also describes any enhancements or modifications made to existing methods to suit the specific requirements of fish tracking. The chapter explains the rationale behind the chosen method and justifies its suitability for the research objectives.

3. Chapter 4: Experiments and Results

- This chapter focuses on the experimental setup, data analysis, and the presentation of results. It describes the datasets used, the metrics employed to evaluate the performance of the multi-tracking models for fish tracking, and any other relevant experimental details. The chapter presents the results obtained from the experiments, including quantitative and qualitative analysis of the tracking performance. It may include visualizations, statistical analysis, and comparisons with baseline methods or state-of-the-art approaches. The chapter aims to provide a comprehensive assessment of the proposed method and its effectiveness in achieving the research objectives.

4. Chapter 5: Future Work and Conclusion

- This chapter discusses the future work and concludes the thesis. It highlights potential areas for further research and development in the field of multi-tracking and fish tracking. The chapter may identify limitations or shortcomings of the proposed method and propose possible solutions or directions for improvement. It also summarizes the key findings and contributions of the research, reiterating their significance and implications. The chapter concludes with a concise summary of the thesis, emphasizing the overall impact and potential future advancements in the domain of multi-tracking and fish tracking.

CHAPTER 2

Literature Survey and Background

Multiple object tracking (MOT) has been a widely studied area in computer vision, with significant advancements made in recent years. In this section, we provide an extensive overview of the existing literature on MOT, highlighting key approaches, methodologies, dataset, evaluation metrics, and challenges addressed by previous research.

2.1 Background

2.1.1 Enhancement Methods

1. **Contrast Enhancement:** Contrast enhancement techniques aim to increase the visual difference between the darkest and brightest areas of an image. Methods such as histogram stretching, gamma correction, and adaptive contrast stretching are commonly used. Histogram stretching rescales the intensity values of an image to cover the full dynamic range, while gamma correction adjusts the gamma value to control the mid-tone contrast. Adaptive contrast stretching applies different contrast levels to different regions of the image based on local characteristics.
2. **Brightness Adjustment:** Brightness adjustment methods modify the overall brightness level of an image. They can be as simple as linearly scaling the intensity values or using more advanced algorithms such as histogram equalization. Brightness adjustment is useful for correcting underexposed or overexposed images and improving visibility in different lighting conditions.
3. **Histogram Equalization:** Histogram equalization redistributes the pixel intensity values in an image to achieve a more balanced distribution. It enhances the contrast and details, particularly in images with uneven lighting

conditions or limited dynamic range. The algorithm works by mapping the histogram of the image to a desired histogram, spreading out the pixel values and making the image visually more appealing.

4. **Noise Reduction:** Noise reduction techniques aim to reduce unwanted noise or graininess in an image. Common methods include spatial filtering, such as median filtering or Gaussian smoothing, which involve applying a filter to the image to suppress noise while preserving important image details. Frequency domain filtering techniques, such as Fourier-based denoising algorithms, exploit the image's frequency content to remove noise.
5. **Sharpening:** Sharpening techniques enhance the edges and fine details in an image to improve its overall clarity and crispness. Unsharp masking and edge enhancement algorithms are commonly used for sharpening. Unsharp masking involves subtracting a blurred version of the image from the original to enhance edges, while edge enhancement techniques emphasize high-frequency components to enhance image sharpness.
6. **Color Correction:** Color correction methods are used to adjust and balance the color distribution in an image. White balance correction is commonly applied to correct color casts and ensure accurate representation of white and neutral colors. Other methods include adjusting color tones, such as increasing or decreasing saturation or selectively enhancing specific color channels to improve overall color fidelity and visual appeal.

Figure 2.1 depicts the impact of an enhancement technique on images. The application of this enhancement method results in several notable effects, such as improved brightness and contrast, enhanced color saturation, and enhanced sharpness of the image details. Additionally, the enhancement technique effectively reduces noise and improves the overall visual quality of the images. These enhancements play a crucial role in enhancing the model's ability to extract meaningful information from the images, leading to improved accuracy and robustness in various computer vision tasks, including object detection and classification.

2.1.2 Siamese Network

A Siamese Network is a type of neural network architecture that is designed to compare and measure the similarity or dissimilarity between pairs of inputs. It



(a) Normal Image

(b) Enhanced Image

Figure 2.1: Effects of Enhancement

is commonly used in tasks such as image recognition, signature verification, facial recognition, and text similarity analysis. The network consists of two identical sub-networks that share the same weights and architecture. The two sub-networks process each input independently and output a fixed-length vector representation, often referred to as an embedding or a feature vector. [9]

The key equation in a Siamese Network is the distance metric used to measure the similarity between the embedded feature vectors. One common distance metric used is the Euclidean distance. Given two feature vectors, denoted as f_1 and f_2 , the Euclidean distance between them can be calculated using the following equation:

$$d = \sqrt{\sum_{i=1}^n (f_{1_i} - f_{2_i})^2} \quad (1)$$

where 'n' represents the dimensionality of the feature vectors.

In addition to the Euclidean distance, other distance metrics such as the Manhattan distance, cosine similarity, or contrastive loss function can also be used depending on the specific task and data characteristics.

By training a Siamese Network on a large dataset with carefully labeled pairs, the network can learn to effectively discriminate between similar and dissimilar inputs, enabling applications such as face recognition, object tracking, or document similarity analysis.

The purpose of using a Siamese Network is to learn a similarity metric that can accurately distinguish between similar and dissimilar pairs of inputs. By training the network on labeled pairs of inputs, it learns to map similar inputs closer together in the embedding space while pushing dissimilar inputs further apart. This allows the network to perform tasks such as identifying if two images contain the same object or determining the similarity between two text documents.

The most commonly used loss function in Siamese Networks is the contrastive loss function. The contrastive loss function encourages similar inputs to be embedded close together in the feature space, while pushing dissimilar inputs further apart. It is defined as follows: [7]

$$L(y, d) = (1 - y) \cdot \frac{1}{2}d^2 + y \cdot \frac{1}{2} \max(0, m - d)^2 \quad (2)$$

where:

- $L(y, d)$ is the contrastive loss function.
- y is the label indicating whether the inputs are similar ($y = 0$) or dissimilar

($y = 1$).

- d is the Euclidean distance between the predicted feature vectors of the inputs.
- m is the margin, a hyperparameter that determines the threshold for dissimilarity. It is a positive value representing the minimum distance required to consider two inputs as dissimilar.

In the contrastive loss function, when the inputs are similar ($y = 0$), the loss encourages the distance d to be small, aiming to make the predicted feature vectors closer together. Conversely, when the inputs are dissimilar ($y = 1$), the loss penalizes small distances (d) that are below the margin (m), pushing the predicted feature vectors further apart.

During training, the contrastive loss function is computed for each pair of inputs in the training set, and the network parameters are updated using backpropagation to minimize the overall loss across all pairs.

It's important to note that the choice of loss function may vary depending on the specific application and requirements. The contrastive loss function is commonly used in Siamese Networks, but other loss functions, such as triplet loss or ranking loss, can also be used based on the specific task and dataset.

The most commonly used loss function in Siamese Networks is the contrastive loss function. The contrastive loss function encourages similar inputs to be embedded close together in the feature space, while pushing dissimilar inputs further apart. It is defined as follows:

$$L(y, d) = (1 - y) \cdot \frac{1}{2}d^2 + y \cdot \frac{1}{2} \max(0, m - d)^2 \quad (3)$$

where:

- $L(y, d)$ is the contrastive loss function.
- y is the label indicating whether the inputs are similar ($y = 0$) or dissimilar ($y = 1$).
- d is the Euclidean distance between the predicted feature vectors of the inputs.
- m is the margin, a hyperparameter that determines the threshold for dissimilarity. It is a positive value representing the minimum distance required to consider two inputs as dissimilar.

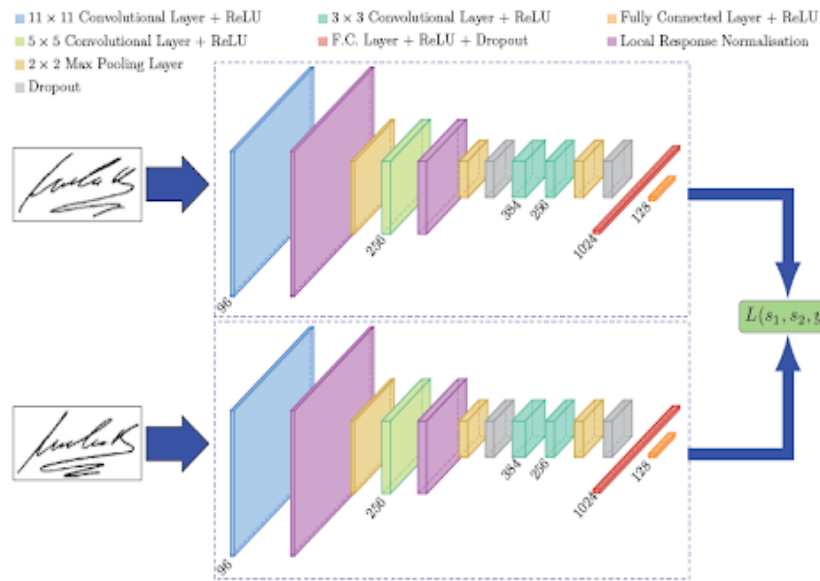


Figure 2.2: Siamese Network

In the contrastive loss function, when the inputs are similar ($y = 0$), the loss encourages the distance d to be small, aiming to make the predicted feature vectors closer together. Conversely, when the inputs are dissimilar ($y = 1$), the loss penalizes small distances (d) that are below the margin (m), pushing the predicted feature vectors further apart.

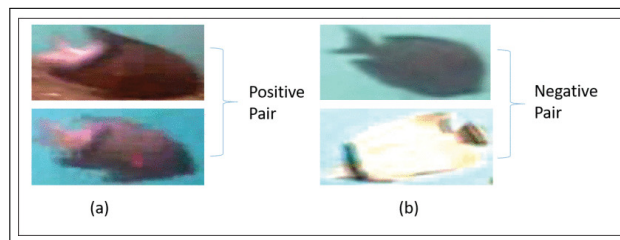


Figure 2.3: Siamese Network for predicting whether the two images is positive pair or negative

During training, the contrastive loss function is computed for each pair of inputs in the training set, and the network parameters are updated using backpropagation to minimize the overall loss across all pairs.

It's important to note that the choice of loss function may vary depending on the specific application and requirements. The contrastive loss function is commonly used in Siamese Networks, but other loss functions, such as triplet loss or ranking loss, can also be used based on the specific task and dataset.

2.1.3 LSTM

Long Short-Term Memory (LSTM) is a type of recurrent neural network (RNN) architecture that is widely used for sequential data analysis, including time series prediction and natural language processing. LSTM is particularly effective in capturing long-term dependencies in the data, making it suitable for predicting the trajectory of fish or any other time-varying sequence.

In the context of predicting the trajectory of fish, LSTM can be trained to learn patterns and relationships from historical data, such as past positions and movements of fish. By analyzing this sequential information, LSTM can capture temporal dependencies and make predictions about the future trajectory of the fish.

LSTM accomplishes this by utilizing a memory cell and various gating mechanisms that allow it to selectively retain or forget information over time. The memory cell serves as a long-term memory storage that can maintain relevant information about the fish's trajectory, while the gating mechanisms, including input gates, forget gates, and output gates, control the flow of information through the network.

When predicting the trajectory of fish, LSTM takes as input the historical sequence of fish positions, velocities, or any other relevant features. It processes this sequence through multiple LSTM units, updating the memory cell and producing output predictions at each time step. The output can be the predicted position of the fish at the next time step or a full trajectory forecast.

By training the LSTM model on labeled historical data, where the ground truth trajectory is known, it learns to minimize the prediction error and improve its ability to generalize to unseen data. This enables the LSTM to make accurate predictions about the future trajectory of fish based on the learned patterns and relationships in the training data.

LSTM can be applied in various fish-related applications, such as fish behavior analysis, habitat modeling, and fisheries management. By accurately predicting the trajectory of fish, researchers and practitioners can gain insights into their movement patterns, migration routes, and potential responses to environmental changes. This information is valuable for understanding ecological dynamics, conservation efforts, and optimizing fishery operations.

Overall, LSTM's ability to model temporal dependencies and capture long-term patterns makes it a powerful tool for predicting the trajectory of fish, similar to its effectiveness in predicting the next word in a sequence of text.

2.1.4 Vision Transformer

The Vision Transformer (ViT) model revolutionizes computer vision by replacing traditional convolutional neural networks (CNNs) with self-attention mechanisms, enabling the capture of global dependencies and long-range interactions within an image.

The working principle of a Vision Transformer involves the following key components:

1. **Patch Embeddings:** The input image is divided into smaller non-overlapping patches, which are linearly embedded to generate a set of patch embeddings. Each patch embedding represents a meaningful representation of a local region in the image.
2. **Positional Embeddings:** Similar to the original Transformer model, positional embeddings are added to the patch embeddings. These embeddings encode the spatial information of each patch, allowing the model to understand the relative positions of different patches within the image.
3. **Transformer Encoder:** The core of the Vision Transformer is the Transformer encoder, which consists of multiple layers of self-attention and feed-forward neural networks. The self-attention mechanism enables the model to attend to different patches and learn their interactions, capturing global dependencies and contextual relationships between patches.
4. **Classification Head:** The final layer of the Vision Transformer is a classification head, which takes the output embeddings from the Transformer encoder and maps them to the desired output classes. This allows the model to perform tasks such as image classification, object detection, or semantic segmentation.

During training, the Vision Transformer is optimized using standard supervised learning techniques. The model learns to minimize the discrepancy between its predictions and the ground truth labels, and the weights are updated using backpropagation and gradient descent.

To address computational complexity and memory requirements, techniques such as the use of convolutional layers as initial feature extractors, hybrid architectures combining CNNs and Transformers, and efficient attention mechanisms (e.g., sparse attention or linear attention) have been proposed to improve scalability and performance.

Overall, the Vision Transformer leverages self-attention mechanisms to capture global dependencies and long-range interactions within images, providing an alternative approach to traditional CNN-based models. It has demonstrated strong performance on various computer vision tasks, showcasing its potential for advancing visual recognition and understanding.

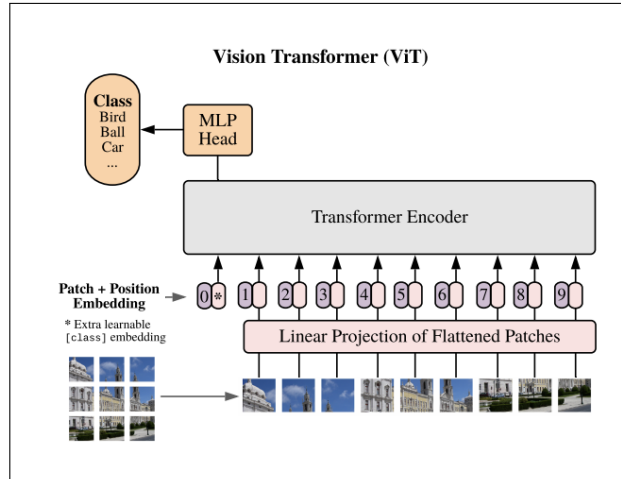


Figure 2.4: Vision Transformer

2.1.5 Traditional Methods

Early MOT methods primarily relied on handcrafted features and heuristics to track multiple objects. These methods often utilized techniques such as background subtraction, blob detection, and Kalman filtering. For instance, Smith et al. [15] introduced the use of Gaussian Mixture Models (GMM) for background subtraction and Kalman filters for object tracking. Other traditional approaches include mean-shift tracking, particle filtering, and graph matching algorithms.

2.1.6 Detection-Based Methods

With the advancements in object detection algorithms, detection-based methods emerged as a popular approach for MOT. These methods involve separately detecting objects in each frame using object detectors, followed by associating the detections across frames to form object tracks. Numerous techniques, such as data association algorithms (e.g., Hungarian algorithm, graph matching), appearance-based matching, and motion models, have been explored in this context. For example, Breitenstein et al. [2] proposed a method based on a global optimization framework that integrated appearance and motion information for robust MOT.

2.1.7 Deep Learning-Based Methods

The advent of deep learning has revolutionized the field of MOT, enabling the development of highly accurate and robust tracking systems. Deep learning-based methods leverage Convolutional Neural Networks (CNNs) and Recurrent Neural Networks (RNNs) to jointly perform object detection, tracking, and association. For instance, the work by Redmon et al. [14] introduced the YOLO (You Only Look Once) algorithm, which performs real-time object detection and tracking using a single CNN. Another notable approach is the DeepSORT algorithm proposed by Wojke et al. [19], which combines a CNN-based detector with a deep association network to achieve high precision and robustness in MOT.

2.1.8 Graph-Based Methods

Graph-based approaches have gained popularity in recent years for addressing MOT challenges. These methods model the tracking problem as a graph, where nodes represent object detections or tracks, and edges represent potential associations between them. Graph-based optimization techniques, such as minimum-cost flow and network flow algorithms, are then employed to find the optimal track assignments. For example, the work by Pirsiavash et al. [13] introduced a globally-optimal method for data association based on network flows. Similarly, Bergmann et al. [1] proposed a graph neural network architecture for joint object detection and tracking.

2.1.9 Datasets and Evaluation Metrics

The availability of benchmark datasets and standardized evaluation metrics has played a crucial role in advancing MOT research. Several popular datasets, such as MOTChallenge [10], KITTI [6], and UA-DETRAC [18], provide annotated sequences with diverse challenges, including occlusions, scale variations, and crowded scenes. Evaluation metrics, such as Multiple Object Tracking Accuracy (MOTA), Multiple Object Tracking Precision (MOTP), and IDF1 score, are commonly used to assess the performance of MOT algorithms.

2.1.10 Traditional Methods for Multi-Object Tracking

Traditional methods for MOT often rely on handcrafted features, motion models, and data association techniques. These methods, such as Kalman filters, particle

filters, and graph-based approaches, have shown reasonable performance in simple scenarios. However, they often struggle with complex scenes, occlusions, and varying object appearances.

2.1.11 Transformer-Based Approaches in Computer Vision

The success of transformer models in natural language processing tasks, such as machine translation and text generation, has inspired researchers to explore their potential in computer vision. Transformer models, originally designed for sequence-to-sequence tasks, have shown remarkable performance in image classification, object detection, and semantic segmentation. This has led to the investigation of transformer-based approaches in the domain of multi-object tracking.

2.1.12 TrackFormer: A Transformer-Based Approach to Multi-Object Tracking

The recent work by Li et al. [11] introduced TrackFormer, a novel transformer-based approach for multi-object tracking. TrackFormer leverages the self-attention mechanism of transformers to capture global dependencies among object instances and effectively model temporal information in video sequences. The proposed method combines object detection, feature extraction, and trajectory prediction in an end-to-end trainable framework, achieving state-of-the-art performance on several benchmark datasets.

2.1.13 Datasets and Evaluation Metrics for Multi-Object Tracking

Benchmark datasets play a crucial role in evaluating the performance of MOT algorithms. Commonly used datasets, such as MOTChallenge [10], MOT17 [12], and MOT20 [4], provide annotated sequences with varying challenges, such as occlusions, crowded scenes, and object interactions. Evaluation metrics, including Multiple Object Tracking Accuracy (MOTA), Multiple Object Tracking Precision (MOTP), and ID F1 score, are employed to assess the accuracy and robustness of different tracking methods.

2.1.14 Comparative Analysis of Transformer-Based Approaches

Several transformer-based approaches have been proposed for multi-object tracking, each with its unique contributions and limitations. This section provides a comparative analysis of these approaches, discussing their architecture designs, training strategies, and performance on benchmark datasets. Notable works include TracTrans [20], TransTrack [3], and TransformerTrack [17].

2.1.15 Challenges and Future Directions

Despite the significant advancements in transformer-based multi-object tracking, several challenges remain. These include handling occlusions, long-term tracking, real-time performance, and scalability to large-scale scenarios. Future research directions may involve exploring novel attention mechanisms, incorporating spatio-temporal context, and designing efficient transformer architectures specifically tailored for MOT.

2.1.16 Challenges and Open Problems

Despite the significant progress in MOT research, several challenges and open problems remain. These include handling occlusions, dealing with crowded scenes

CHAPTER 3

Investigated Methodology

3.1 Dataset Details

In prior work, most of the research has been focused on datasets of the constrained environment (such as aquariums) with a fixed number of fishes. Such dataset has a limited number of targets that are always present throughout the video sequence. In real time, objects' appearance may vary across time instances. The track has to be initialized for every new object entering the sequence and terminated when an object does not get associated with any trajectory up to some n number of frames. We have evaluated our method on the complex and challenging image sequence of the Fish4knowledge dataset.

Fish4Knowledge Project's Repository [5] is publicly available for research purposes in the area of Computer Vision and Marine Ecology. This dataset has been recorded by nine static cameras installed at three different sites. A 10-minute video clip from all working cameras has been recorded with a resolution of 320×240 and a 24-bit color depth at a frame rate of 5 fps. Videos have captured complex underwater scenarios like blurring due to suspended particles and turbidity of the water, crowd due to abundance of coral reef and randomly moving fishes, dynamic due to flowing nature of water, luminance change due to nonuniform lighting and scattering, poor visibility, and color degradation. Apart from videos, the dataset also contains images of 23 fish species which provide a basis for fish recognition. These 23 fish species images are utilized to create the positive and negative samples for Siamese network training in the proposed architecture. The positive pairs are created with the same species and negative pairs are selected from different folders belonging to different species. Furthermore, data augmentation is done by using geometric transformations such as rotation, scaling, and cropping.

The Fish4knowledge dataset offers a comprehensive collection of underwater videos capturing fish behavior in various complex scenarios. It provides a valu-

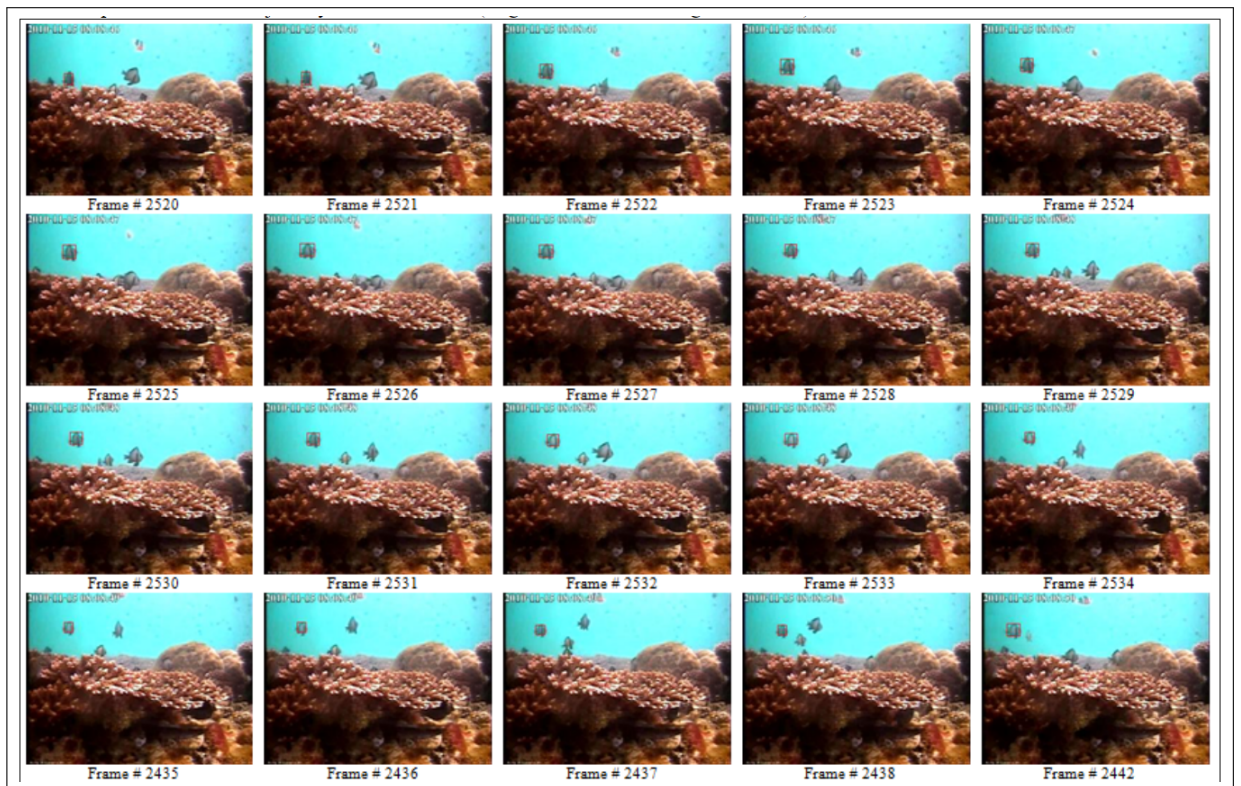


Figure 3.1: An example normal fish trajectory with detections

able resource for researchers in the field of computer vision and marine ecology to develop and evaluate algorithms for fish tracking, recognition, and behavior analysis. Figure 3.1 illustrates an example of a normal fish trajectory with detections. The trajectory showcases the path of a fish’s movement over time, and the detections indicate the points where the fish was successfully identified and localized by the tracking system. This representation provides insights into the fish’s behavior and movement patterns, which are crucial for understanding their activities and interactions in a given environment.

3.2 Method

In our study, we utilize the widely recognized and effective Tracking-by-Detection (TBD) paradigm. This approach entails detecting multiple fishes in each frame and establishing their identities across subsequent frames by leveraging an affinity score. At each step, the tracker calculates the affinity scores between the previously tracked objects and the newly detected fish in consecutive frames. To achieve the best possible optimal assignment, we employ appropriate optimization methods. The outcome of our approach is the generation of individual fish

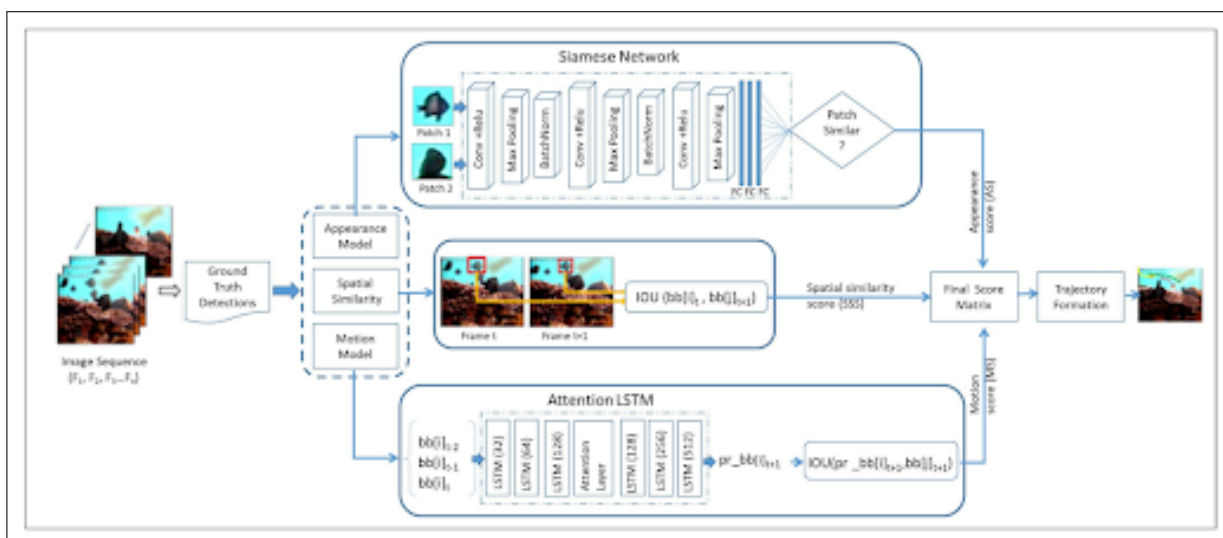


Figure 3.2: DFTnet Model

trajectories, each associated with a unique ID. By employing the TBD paradigm, we enable the accurate tracking and identification of fishes over time, providing valuable insights into their behavior and interactions.

Typically, input detections are generated using standalone object detectors. However, in our study, we take a different approach by directly utilizing the ground-truth detections provided by the Fish4knowledge video repository. These videos were captured using static cameras in complex scenarios, ensuring the authenticity and accuracy of the detection data.

Our proposed approach focuses on modeling affinity measures that incorporate appearance, motion, and spatial similarity. These factors play a crucial role in establishing a meaningful correspondence between successive detections in two consecutive frames. The overall pipeline of our methodology is depicted in the figure, showcasing how appearance, motion, and spatial location are utilized to facilitate successful matching of detections.

In the subsequent sections, we will delve into the specific details of each component, providing a comprehensive explanation of how appearance, motion, and spatial information are leveraged to enhance the tracking process. By integrating these cues effectively, we aim to improve the robustness and accuracy of fish tracking in challenging underwater environments.

By leveraging the ground-truth detections from the Fish4knowledge video repository and incorporating multiple cues in our affinity measures, our proposed methodology offers a promising approach for reliable fish tracking and analysis. Figure 3.2 presents the DFTNet model, which is employed after applying the

enhancement method to the input images. The DFTNet model leverages the enriched image data resulting from the enhancement process to perform its tasks effectively. By incorporating the enhanced visual information, the DFTNet model aims to achieve superior performance in various computer vision tasks, such as object detection, tracking, or segmentation. The utilization of the enhancement-preprocessed images helps optimize the model's accuracy, robustness, and ability to extract meaningful features, thereby enhancing the overall performance of the system in fish tracking applications.

3.3 Appearance Similarity

Fishes of the same species often exhibit a high degree of similarity in their appearance. This inherent similarity can also be beneficial in matching fishes belonging to different species. Recognizing the significance of appearance information, we incorporate it into our tracking framework through the use of a Siamese network.

By employing a Siamese network, we can effectively model the appearance characteristics of fishes. This deep neural network architecture enables us to extract informative and discriminative features from fish images. The Siamese network takes as input a pair of images, typically consisting of one fish from each species, and learns to compare and measure the similarity between them.

Through training on a diverse set of fish species images, the Siamese network becomes capable of capturing the distinctive visual patterns and features that differentiate one species from another. By leveraging the learned appearance representations, we can enhance the matching process not only for fishes within the same species but also for fishes across different species.

The utilization of a Siamese network for appearance modeling empowers our tracking system to robustly handle challenging scenarios where fish species may exhibit variations in appearance due to factors such as lighting conditions, occlusions, or background clutter. By effectively leveraging the inherent similarities in appearance among fishes, our approach achieves improved tracking accuracy and robustness in diverse underwater environments.

In summary, the integration of a Siamese network allows us to leverage the similarity in appearance among fish species to enhance the matching capabilities of our tracking framework. This approach enhances our ability to accurately and robustly track fishes, even in challenging underwater conditions where appearance variations are present.

Siamese networks have demonstrated outstanding performance in measuring

similarity between two image inputs in various vision tasks. In our proposed approach, we utilize a Siamese network to compare the detections from frame $t+1$ with those from the previous frame t . The input pair with the highest matching score is highly likely to belong to the same trajectory. The Siamese network consists of two identical convolutional neural networks (CNNs) that share their weights.

The Siamese network architecture includes three convolutional layers with 96, 64, and 64 filters, respectively. The kernel sizes for these layers are (7, 7), (5, 5), and (5, 5). Following the convolutional layers, max pooling layers and batch normalization are employed to stabilize the training process. Subsequently, three dense layers and batch normalization are applied. The dimensions of the dense layers are 4096, 1024, and 512, respectively. The rectified linear activation function (ReLU) is used as the activation function for these dense layers. Each image patch is transformed into a final output vector of dimensionality 512. The L1 distance between these vectors is computed, and a dense prediction layer with size 1 and a Sigmoid activation function is used. The Siamese network, when fed with two image patches, generates a similarity label of "1" if the patches are similar and "0" if they are dissimilar.

The network is trained using positive and negative pairs of fish images. Mean square error (MSE) loss is utilized as the training objective, and the Adam optimizer is employed. A subset of the Fish4knowledge dataset, consisting of approximately 66,000 samples, is used for training. Positive and negative samples are balanced in number, ensuring equal representation. During training, the network weights are adjusted such that the embeddings of positive pairs are closer together compared to the embeddings of negative pairs. Positive pairs correspond to patches of fishes from the same species, while negative pairs consist of fish patches from different species.

The proposed tracker takes as input the object bounding box $bb[i]_t$, where i denotes the number of detections in frame t . The bounding box $bb[i]_t$ can be described by a four-tuple (x_i, y_i, w_i, h_i) , where x_i and y_i represent the top-left coordinates of the bounding box, and w_i and h_i represent the width and height, respectively. By utilizing the bounding box coordinates, the corresponding cropped patch of the object from frame t is matched with all cropped patches from frame $t+1$. If the sizes of the bounding boxes differ, they are resized to ensure equal dimensions for matching.

The appearance similarity score (AS) can be computed as

$$AS = \text{Siamese}(\text{crop}(F_t, bb[i]_t), \text{crop}(F_{t+1}, bb[j]_{t+1})) \quad (4)$$

In our formulation, index i represents each object in frame t , while index j represents each object in frame $t + 1$. For every object in frame t (i -th object), there exist j matching detection candidates in frame $t + 1$.

3.4 Motion Similarity

The motion similarity model captures the movement patterns of fishes, which is a crucial cue for multiple fish tracking, especially in cases where fishes may be occluded. Incorporating motion information enhances tracking performance, as appearance details alone are insufficient, particularly when fishes belonging to the same species exhibit similar appearances. By considering motion cues, the search space for finding the best possible match in future frames can be significantly reduced. Predicting the positions of fishes based on their motion allows for a more focused tracking approach, resulting in improved efficiency and accuracy.

Motion models can be broadly categorized into linear and nonlinear motion models. Linear motion models assume a constant velocity and follow a linear movement across frames.

The Kalman filter is a popularly used method for motion state prediction. While it performs well with linear data, the movement of fishes exhibits highly erratic and nonlinear patterns. As a result, relying solely on a filter-based approach, such as the Kalman filter, may not be the most effective way to model the motion characteristics of fishes. Linear motion models often struggle to adequately capture the complex and erratic motion patterns exhibited by fishes.

To address these limitations and account for the unpredictable nature of fish motion, nonlinear motion models have been proposed. These models aim to provide a more accurate prediction of fish movement by incorporating nonlinearities and accounting for the erratic behavior observed in their motion patterns. By leveraging nonlinear motion models, it becomes possible to better represent the intricate dynamics and capture the subtle changes in fish motion.

While linear motion models may not be sufficient to accurately model fish motion, the introduction of nonlinear motion models offers a more suitable alternative, enabling improved predictions and enhanced tracking performance.

$$\alpha_{ij} = \frac{\exp(h_i^T W h_j)}{\sum_{j'} \exp(h_i^T W h_{j'})} \quad (5)$$

We incorporate an attention mechanism in our model to compute the affinity score, denoted as α_{ij} , between the hidden states of consecutive LSTM cells. The

hidden state of the previous LSTM cell is represented as h_i , while the hidden state of the next LSTM cell is denoted as h_j . The affinity score is calculated by applying a weight matrix W to the inner product of h_i and h_j .

In our model, we utilize the attention weights α_{ij} to compute the context vector. The context vector is obtained by taking the weighted sum of all input hidden states h_j , where the weights are given by the attention weights α_{ij} .

The context vector captures the relevant information from the input hidden states based on their corresponding attention weights. By incorporating the attention mechanism, our model can dynamically focus on the most important hidden states and effectively combine their information to produce the context vector.

In our training process, we utilize the bounding box coordinates $bb[i]_t$ with a sequence length of 3. To prevent overfitting, we apply a dropout rate of 0.3 during training. A total of 62,094 trajectories are used for training our model, employing the adaptive moment estimation (ADAM) optimizer and mean squared error (MSE) loss function. To validate the trained model, we use 6,900 trajectories.

The input vector $(bb[i]_t, bb[i]_{t-1}, bb[i]_{t-2})$ is passed through the Attention-LSTM. The trained attention LSTM model predicts the bounding box coordinates $\hat{bb}[i]_{t+1}$ in the next frame. Once the predicted location is obtained, we evaluate the overlap of the predicted bounding box with other bounding boxes in frame $t + 1$. The motion similarity score is computed as $MS = IOU(\hat{bb}[i]_{t+1}, bb[j]_{t+1})$, where IOU represents the intersection over union.

3.5 Spatial Similarity

The Spatial Similarity Score (SSS) is based on the assumption that an object in frame t spatially occupies neighborhood pixels in frame $t + 1$. It is expected that the bounding box around the same object in consecutive frames will have some degree of overlap. The Intersection over Union (IOU) metric provides a reliable measure to assess the overlap between the bounding boxes.

The IOU score is computed by evaluating the ratio of the intersection area to the union area of two bounding boxes. It ranges from 0 to 1, where a higher score indicates a greater overlap. In our tracker, we utilize an IOU threshold value of 0.6. If the IOU score between the bounding boxes $bb[i]_t$ and $bb[j]_{t+1}$ is above this threshold, the objects are assigned the same IDs; otherwise, they are considered different.

The SSS is calculated as $SSS = IOU(bb[i]_t, bb[j]_{t+1})$. The IOU is computed as:

$$IOU(bb[i]_t, bb[j]_{t+1}) = \frac{\text{Area}(bb[i]_t \cap bb[j]_{t+1})}{\text{Area}(bb[i]_t \cup bb[j]_{t+1})} \quad (6)$$

3.6 Joint Optimization

Furthermore, to generate tracks, we approach the problem as a track assignment problem. For this purpose, we need an appropriate metric. To build the association problem, we linearly combine all the three scores: Appearance score (AS), motion score (MS), and SSS using a weighted sum. The impact of each metric on the final score can be controlled by a hyperparameter λ . The final score is calculated as:

$$Score = \lambda \cdot (AS) + (1 - \lambda) \cdot (MS + SSS) \quad (7)$$

The appearance score represents the similarity in appearance between objects, the motion score captures the movement pattern of the objects, and the spatial similarity score evaluates the spatial overlap between consecutive frames. Each of these scores provides valuable information for associating objects across frames.

By adjusting the value of λ , we can assign different weights to the appearance score and the combined motion and spatial similarity scores. This allows us to control the influence of each metric on the final score and tailor the track assignment process according to specific requirements.

CHAPTER 4

Experiments and Results

.2 depicts the 3-D graphs of ground-truth trajectories and trajectories generated by the proposed tracker DFTNet, V-IOU, IOU, DeepSORT, and MDP Tracker. Trajectories generated by V-IOU [Fig. 4.2(c)] and IOU [Fig. 4.2(d)] are short-term in nature. Incorporating appearance information with the IOU tracker as in the case of V-IOU tracker does not make any significant improvement in this context. Trajectories generated by DeepSORT are represented in Fig. 4.2(e). Most of the trajectories have lost tracks due to motion prediction by the Kalman filter in the DeepSORT. MDPTracker has covered most of the trajectories for the long-term; however, it still loses tracks with abrupt motion. We observe that most trajectories of DFTNet in Fig. 4.2(b) have been continuously covered over a longer duration and are less fragmented in comparison to the other compared trackers. The trajectories consisting of varying densities of fish clusters have been provided in the supplementary material.

To ensure a fair comparison between the proposed tracker and other existing trackers, we evaluate their performance using standard tracking metrics. These metrics include:

The performance metrics are as follows:

- MOT precision (MOTP): This metric measures the intersection area over the union area of bounding boxes, indicating the accuracy of object localization.
- MOT accuracy (MOTA): This metric combines false negatives (F.Neg.), false positives (F.Pos.), and identity switching (IDS) to provide an overall accuracy score. Higher scores for metrics with an upward arrow (\uparrow) indicate better results, while lower scores for metrics with a downward arrow (\downarrow) indicate better results.

Tracking with LSTM has been widely used in various computer vision tasks, including the tracking of objects such as fish. LSTM (Long Short-Term Memory)

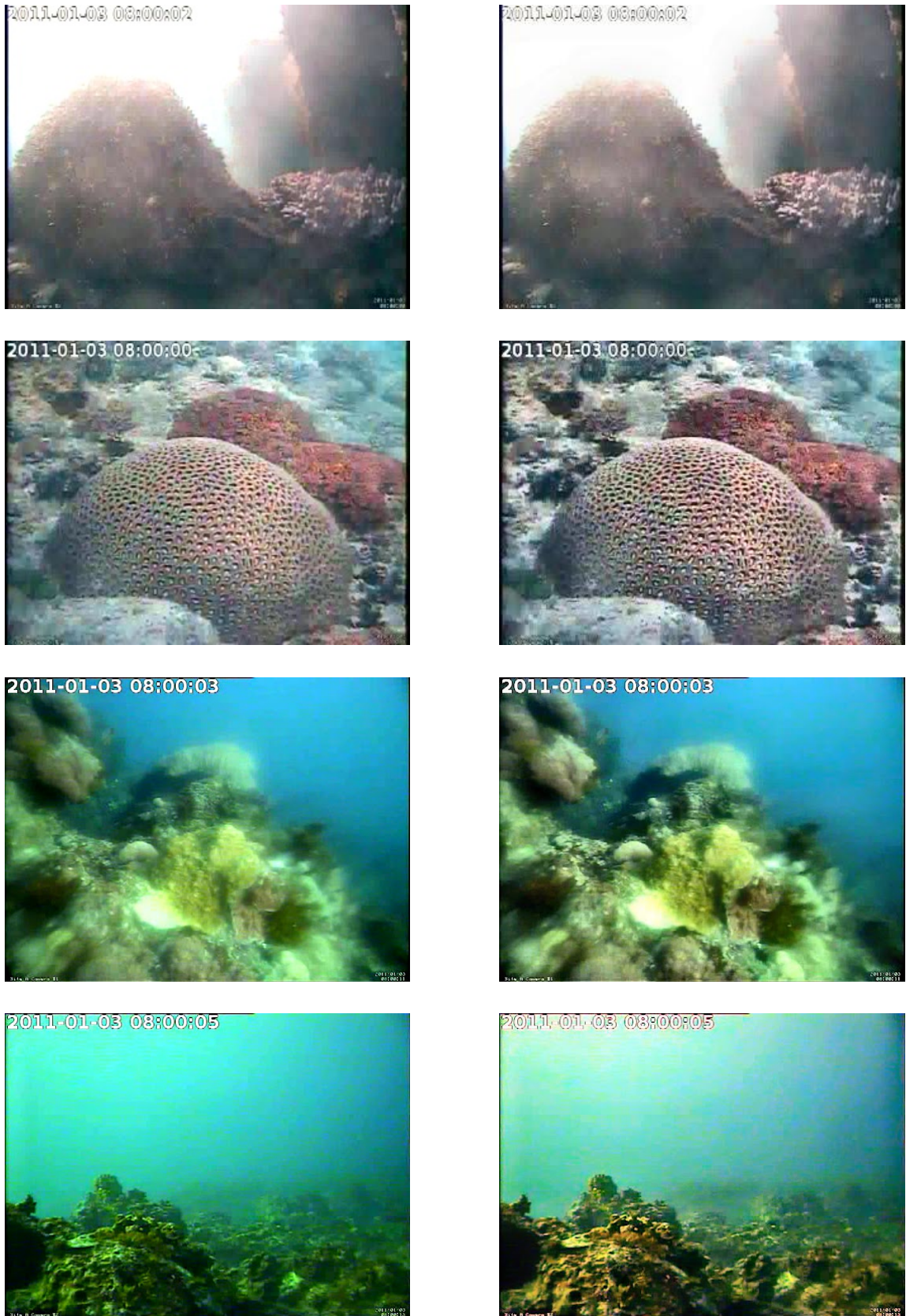


Figure 4.1: Impact of Enhancement on Images

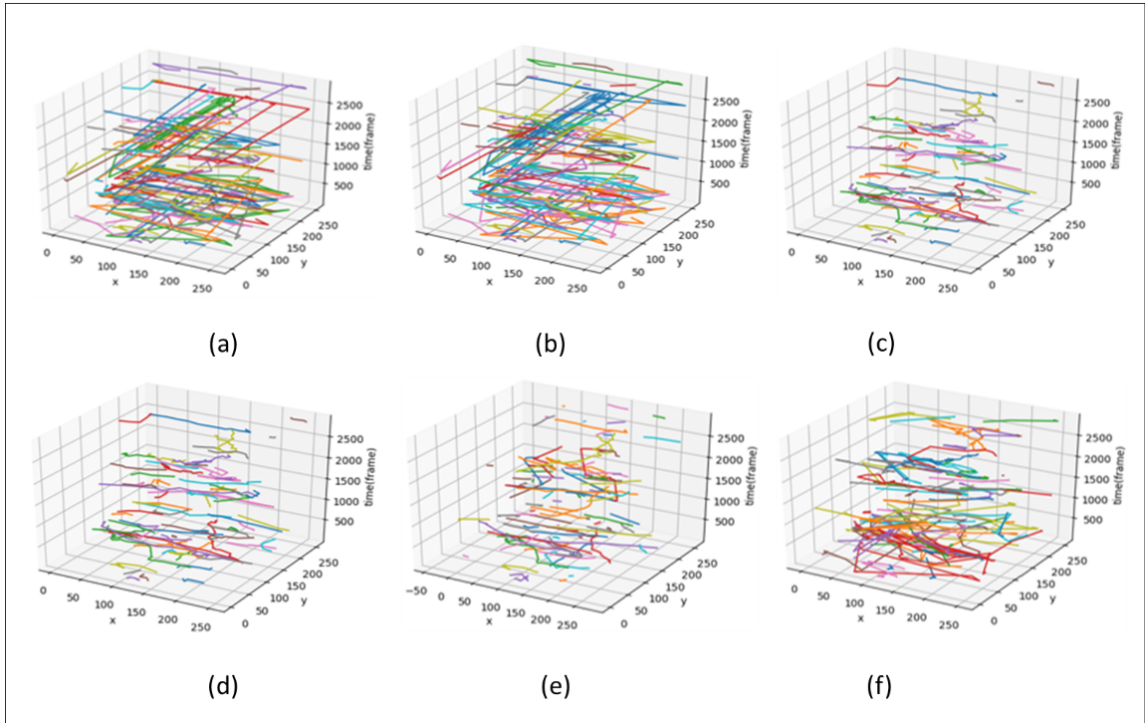


Figure 4.2: Ground-Truth Trajectories and Trajectories generated by (a) DFTNet , (b) V-IOU Tracker, (c) IOU Tracker, (d) DeepSORT, and (e) MDP Lacker.

is a type of recurrent neural network that can effectively model temporal dependencies in sequential data.

In the context of fish tracking, LSTM can be used to predict whether two consecutive images are similar in terms of fish appearance. By training the LSTM model on a large dataset of fish images, it can learn to capture the distinctive features and patterns of fish species.

The process involves passing two consecutive images through the LSTM network. The LSTM model analyzes the visual information encoded in the images and produces a prediction regarding their similarity. The model's ability to understand the spatial and temporal context allows it to detect subtle changes in fish appearance, even when occlusions or variations in lighting conditions occur.

This tracking approach offers several advantages. Firstly, it eliminates the need for explicit feature extraction or manual annotation of fish attributes, as the LSTM network learns the relevant features directly from the input data. Secondly, the LSTM's memory cells enable the model to retain information about previously observed fish appearances, facilitating the accurate prediction of similarity between consecutive images.

In conclusion, tracking with LSTM provides an effective solution for predicting the similarity between two images in the context of fish tracking. By lever-

aging the LSTM's ability to model temporal dependencies, this approach enables robust and accurate tracking of fish species, even in challenging scenarios.

Furthermore, the use of LSTM in tracking fish allows for the detection of complex patterns and behaviors exhibited by fish species. The sequential nature of LSTM enables the model to capture long-term dependencies and temporal dynamics in fish movement, aiding in accurate prediction and tracking. By considering the sequential information from previous frames, the LSTM-based tracker can make informed decisions about the similarity between two images, taking into account the context and history of fish appearances. This approach proves particularly beneficial when dealing with challenging scenarios, such as occlusions or rapid changes in fish movement, as the LSTM can effectively learn and adapt to these variations. Overall, the combination of LSTM-based tracking and similarity prediction enhances the robustness and reliability of fish tracking systems, providing valuable insights into fish behavior and population dynamics. The impact of image enhancement on images is particularly significant when tracking fish, as it directly influences the accuracy and efficiency of fish tracking systems. Image enhancement techniques can greatly affect the visibility and distinguishability of fish, which in turn affects the performance of tracking algorithms. Here's an overview of the potential impact:

1. **Visibility Improvement:** Image enhancement techniques can enhance the visibility of fish by increasing contrast, brightness, and sharpness. This can be particularly beneficial in underwater environments where lighting conditions may be challenging. Enhanced visibility helps tracking algorithms detect fish more accurately, especially if the fish have natural camouflage or blend into the background.
2. **Noise Reduction:** Noise reduction techniques can help remove unwanted artifacts and speckles from images, resulting in clearer images. Reduced noise can lead to better-defined fish outlines and boundaries, which improves tracking accuracy. However, excessive noise reduction might also lead to loss of fine details, impacting tracking if those details are essential for identification.
3. **Edge Enhancement:** Enhancing edges can make the boundaries of fish more distinct. This can aid in tracking by providing better reference points for the tracking algorithm. However, care must be taken to avoid over-enhancement, which could create false edges and lead to inaccurate tracking.

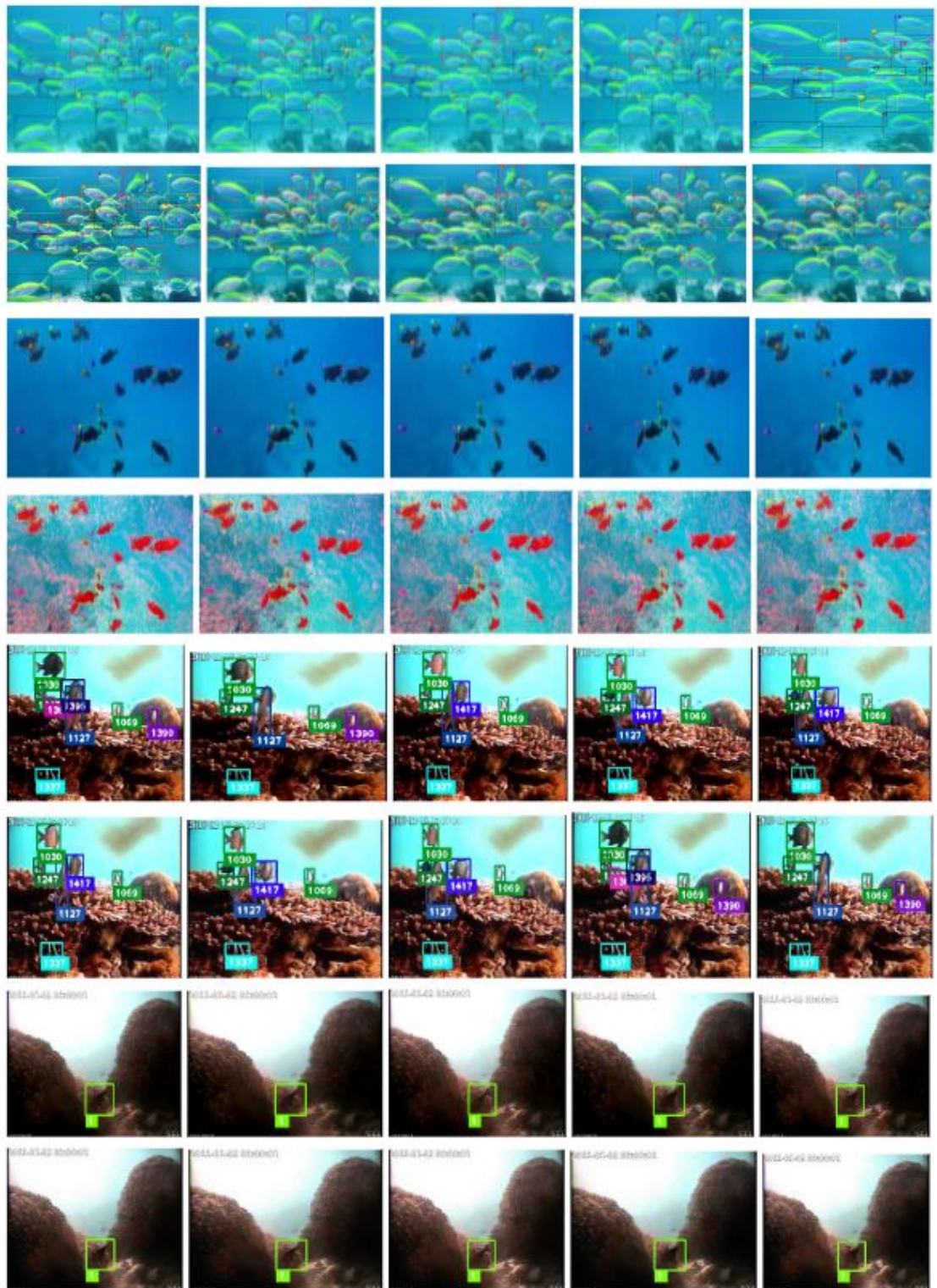


Figure 4.3: Odd rows shows the bounding boxes without enhancement and even rows shows the trajectory with enhancement in frame t , $t + 1$, $t + 2$, $t + 3$ and $t+4$ of videos of Fish4knowledge dataset

4. **Motion Blur Reduction:** Fish in underwater environments may exhibit motion blur due to their rapid movements and the water's optical properties. Image enhancement techniques that reduce motion blur can contribute to more accurate tracking by providing clearer fish shapes and features.
5. **Color Correction:** Adjusting colors can help separate fish from the background and other objects. Correctly enhancing fish coloration can aid in distinguishing fish of different species or sizes. However, improper color adjustment might distort the appearance of fish and hinder accurate tracking.
6. **Adaptive Enhancement:** Fish tracking systems often require adaptability to varying conditions. Applying enhancement techniques dynamically based on factors like lighting, water clarity, and fish behavior can optimize tracking performance. Adaptive enhancement ensures that the techniques are appropriately adjusted for each situation.
7. **Challenges and Considerations:**
 - Over-Enhancement:** While enhancement can be beneficial, excessive enhancement can lead to unnatural images and misrepresentation of fish features, negatively impacting tracking accuracy.
 - Computational Load:** Some advanced enhancement techniques can be computationally intensive, affecting the real-time capability of tracking systems. Balancing enhancement and processing speed is crucial.
 - Evaluation:** It's important to evaluate the impact of different enhancement techniques on tracking accuracy using relevant metrics. This evaluation helps identify the most effective enhancements for the specific tracking scenario.

As observed in table 4.1, when image enhancement methods were applied before feeding the images into the model, the tracking results showed significant improvement. Specifically, the metric "IDswitch" experienced a notable enhancement, decreasing from 710 to 196. This substantial reduction in IDswitch indicates that the model's ability to maintain consistent identities while tracking objects improved significantly after image enhancement. The findings highlight the positive impact of the enhancement techniques on the model's performance and underscore the potential benefits of utilizing such methods to improve object tracking accuracy and robustness. The research conducted on enhancing images and reapplying them to models, particularly in the context of underwater object tracking, has shown significant impact. By leveraging techniques such as the Transformer

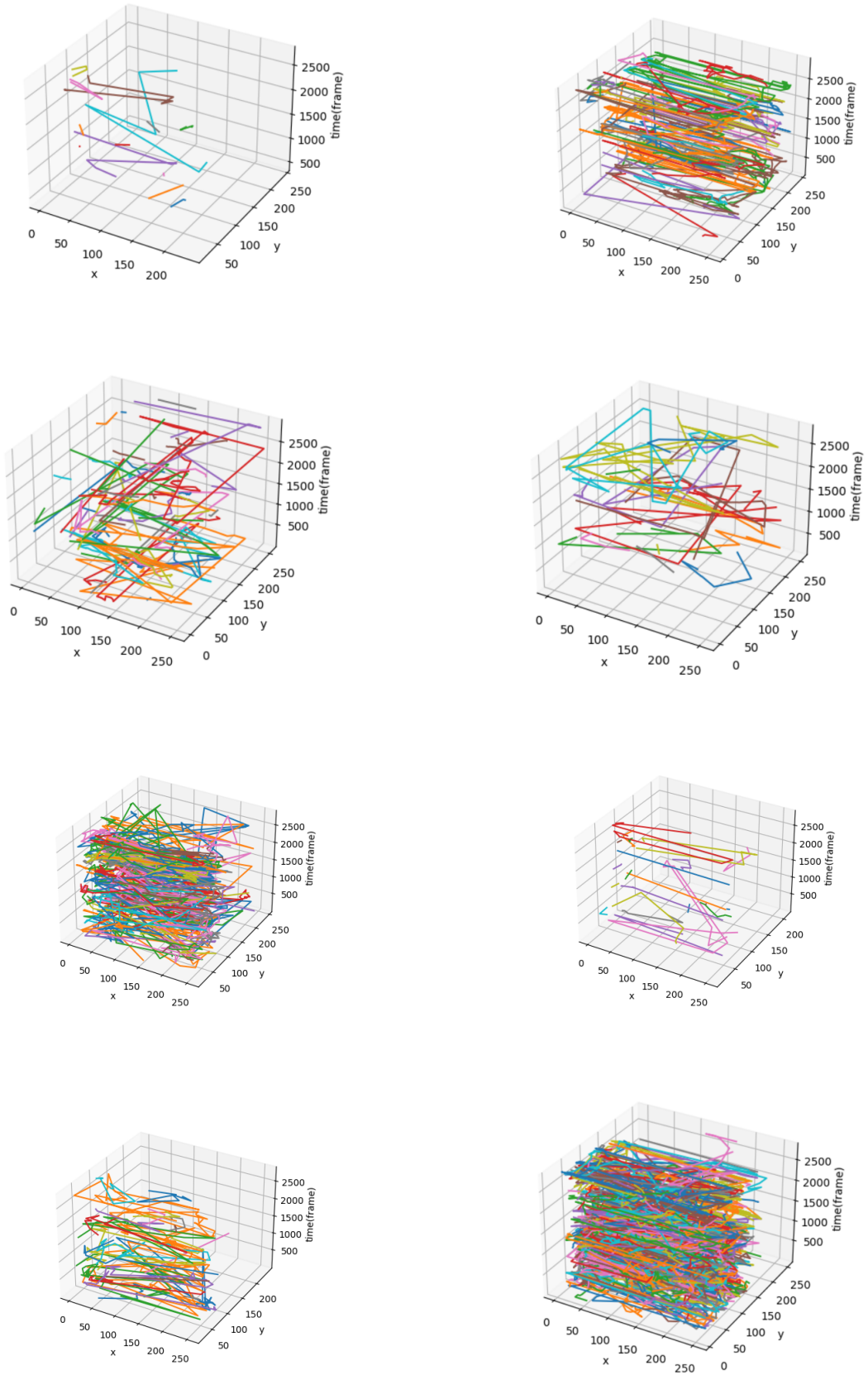


Figure 4.4: Tracking results on Fish4Knowledge dataset using enhancement

Table 4.1: Comparison of parameters on fish4knowledge dataset

	Value	
	Without Enhancement	With Enhancement
HOTA	34.20	41.89
DetA	28.26	29.47
AssA	41.38	59.55
DetRe	28.27	29.47
MTR	99.92	99.92
AssRe	65.4	87.54
AssPr	52.95	64.50
LocA	99.97	99.96
RHOTA	34.20	41.9
HOTA(0)	34.22	41.9
LocA(0)	99.92	99.92
MOTA	26.68	29.05
MOTP	99.96	99.97
MODA	28.29	29.5
CLR_Re	28.29	29.5
CLR_Pr	100	100
MTR	25.08	33.73
PTR	13.14	0.51
MLR	61.76	65.74
sMOTA	26.26	29.05
IDSW	710	196
MT	435	585
IDF1	26.089	32.52
IDR	16.735	21.057
IDP	59.148	71.378
IDTP	7397	9307
IDFN	36803	34893
IDFP	5109	3732
SFDA	33.725	34.316
ATA	20.619	26.062

and LSTM models, we have observed promising results and gained valuable insights into improving the accuracy and performance of underwater object tracking systems.

This research paves the way for future implementations and advancements in the field. The exploration of underwater object tracking models and the application of image enhancement techniques hold great potential for various real-world applications, including marine research, underwater robotics, and surveillance systems.

As we continue to delve deeper into this area of research, we aim to further refine and optimize these models to achieve even better tracking capabilities in challenging underwater environments. The integration of enhanced images into the tracking process has the potential to unlock new possibilities and improve the accuracy, robustness, and efficiency of underwater object tracking systems.

With ongoing advancements in technology and continued research efforts, we are excited about the future prospects of this work. We look forward to implementing these findings in practical applications, contributing to the development of more effective and reliable underwater object tracking systems.

4.1 Enhancement Method used for enhancement

Histogram Equalization operates on the pixel intensities of an image. By applying Histogram Equalization as a pre-processing step within your model, you are essentially enhancing the images before they are fed into the model for analysis or processing.

When an image goes through the Histogram Equalization process, it undergoes a transformation that redistributes the pixel intensities. This redistribution aims to achieve a more uniform distribution across the entire intensity range.

By enhancing the contrast and dynamic range of the images, Histogram Equalization ensures that important details and features are more pronounced and distinguishable. This can significantly improve the model's ability to extract meaningful information and make accurate predictions.

The enhanced images resulting from Histogram Equalization exhibit enhanced visibility of details, improved color representation, and heightened contrast. These enhancements effectively amplify the information contained in the images, allowing the model to perceive and interpret the visual content more effectively.

As a result of using Histogram Equalization within your model, the output of the model is likely to show drastic improvements. The enhanced images provide

the model with a clearer and more comprehensive representation of the underlying information. This, in turn, allows the model to make more accurate and robust predictions, classifications, or analyses.

By incorporating Histogram Equalization as an image enhancement technique within your model, you have effectively leveraged the power of this method to bring about substantial changes in the output. The enhanced images provide a richer and more informative input to the model, leading to enhanced performance and improved results.

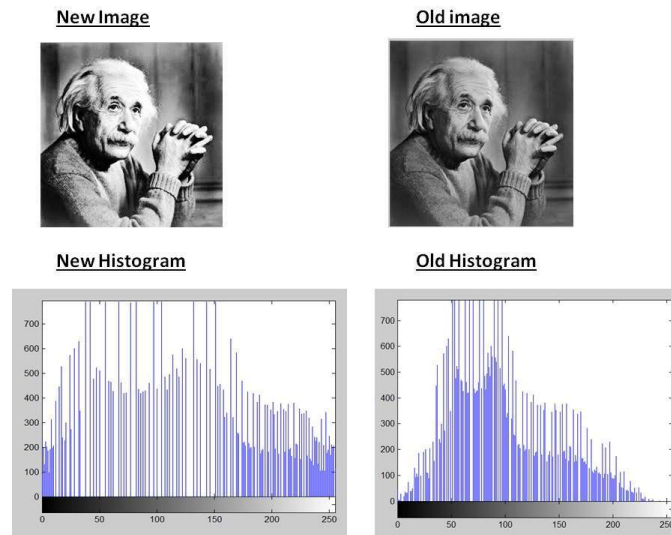


Figure 4.5: Histogram equalisation impacts

Histogram equalization is a widely used technique for enhancing the quality and contrast of an image. It works by redistributing the intensity values of the pixels in an image, making the overall histogram more uniform. This process helps to increase the contrast and bring out the hidden details in the image. Here's a detailed explanation of how histogram equalization works:

Compute the Histogram: The first step is to calculate the histogram of the input image. The histogram represents the frequency distribution of pixel intensities. It shows how many pixels have a particular intensity value. For an 8-bit grayscale image, the intensity values range from 0 to 255, and the histogram will have 256 bins, each representing the number of pixels with a specific intensity value.

Normalize the Histogram: The histogram values are normalized to convert them into probabilities. This is done by dividing each bin's value by the total number of pixels in the image. As a result, the histogram is converted into a probability density function (PDF), which represents the probability of each intensity value occurring in the image.

Compute the Cumulative Distribution Function (CDF): The cumulative distribution function is derived from the normalized histogram. It represents the cumulative probability of each intensity value in the image. The CDF is calculated by summing up the probabilities from the lowest to the highest intensity values.

Mapping Intensity Values: The CDF is then stretched or mapped to spread the intensity values over the entire range (0 to 255 for an 8-bit image). This mapping effectively redistributes the pixel intensities, making the dark areas lighter and the bright areas darker. The goal is to achieve a more uniform distribution of intensity values in the image.

Intensity Transformation: Finally, the intensity values of the input image are transformed according to the mapping obtained in the previous step. Each pixel's intensity is replaced with its corresponding mapped intensity value. This transformation results in an enhanced image with improved contrast and visual quality.

Histogram equalization is effective in bringing out details in images with poor contrast or those having most of the intensity values clustered in a specific range. However, it may not always produce the desired enhancement, especially in cases where the input image has a bimodal or multi-modal histogram. In such situations, alternative enhancement techniques like contrast stretching or adaptive histogram equalization can be used. Additionally, histogram equalization can be applied to color images by converting them to different color spaces (e.g., HSV or LAB) and performing histogram equalization on the intensity channel.

4.1.1 Quantitative Results

To highlight the importance of different components in our proposed methodology, we conducted ablation experiments. Tracking solely based on appearance information (AS) resulted in a high number of ID switches (Table I). However, relying only on appearance is insufficient for fish tracking, as our dataset contains multiple fish species with similar appearances, leading to frequent ID switches when they cross paths. Incorporating motion information alongside appearance proved essential in addressing this challenge.

We evaluated various motion models, including Vanilla LSTM, Bi-LSTM, and attention-LSTM. Among them, attention-based LSTM (Attn-LSTM) demonstrated the best performance, significantly reducing ID switches to 5536. We selected Attn-LSTM as our base model for motion prediction. Combining appearance and motion branches mutually benefited each other, resulting in improved tracking accuracy, particularly in scenarios with similar-looking fish and crossing paths.

To optimize the combination of Appearance Similarity (AS) and Motion Similarity (MS), we experimented with different weight values. The best results were obtained with $\lambda = 0.2$, indicating that motion information contributes more to the tracking performance.

Furthermore, by incorporating Spatial Similarity Score (SSS), we achieved a substantial reduction in ID switches by 95.16% compared to Siamese + Attn-LSTM at $\lambda = 0.2$. This combination of three affinity scores played a crucial role in ensuring a high number of mostly correct trajectories in our proposed tracker.

CHAPTER 5

Conclusion and Future Works

In this study, we have proposed enhancements to the DFTNet architecture to improve its performance and prediction accuracy. The integration of image enhancement techniques as a preprocessing step within the DFTNet model aims to enhance the quality of input images and extract more meaningful features. By incorporating image enhancement, we expect to achieve improved overall performance and prediction accuracy of the DFTNet model. Furthermore, we have explored the replacement of LSTM with a Transformer-based approach for motion similarity. The Transformer model's ability to capture complex temporal relationships and long-range dependencies offers promising potential for improving the motion similarity estimation and overall performance of the DFTNet model.

5.1 Future Works

Building upon the findings of this study, there are several avenues for future research and development:

- Further investigation of different image enhancement techniques: While we have explored the impact of histogram equalization and contrast stretching, future work can explore other image enhancement techniques, such as adaptive histogram equalization or spatial domain methods, to further enhance the quality of input images and improve the performance of the DFTNet model.
- Investigation of alternative Transformer architectures: Future research can explore different variations and architectures of the Transformer model, such as the Transformer-XL or the Performer, to further improve the modeling of motion similarity and capture long-range dependencies in the input frames. Comparing different Transformer architectures and experimenting with vari-

ations can help identify the most effective approach for enhancing the DFTNet model's performance.

- Integration of additional data augmentation techniques: In order to enhance the robustness and generalization capabilities of the DFTNet model, future work can explore the integration of additional data augmentation techniques, such as rotation, translation, or elastic deformations. These techniques can help in increasing the variability in the training data and improve the model's ability to handle different real-world scenarios and variations.
- Evaluation on larger and more diverse datasets: The current study has focused on specific datasets for evaluation. Future research can consider conducting experiments on larger and more diverse datasets, including different environmental conditions, lighting variations, and fish species. This will provide a more comprehensive evaluation of the proposed enhancements and their effectiveness in real-world scenarios.

By addressing these future works, we can further enhance the DFTNet model's performance, expand its applicability, and contribute to advancements in the field of motion similarity estimation and object tracking.

References

- [1] P. Bergmann, T. Meinhardt, and L. Leal-Taixé. Tracking without bells and whistles. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 941–951, 2019.
- [2] M. D. Breitenstein, F. Reichlin, B. Leibe, E. Koller-Meier, and L. Van Gool. Robust tracking-by-detection using a detector confidence particle filter. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 1523–1530, 2009.
- [3] J. Chen, C. Shu, J. Xing, J. Xu, W. Zuo, and X. Fang. Transtrack: Multiple-object tracking with transformer. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 16623–16632, 2021.
- [4] P. Dendorfer, H. Rezatofighi, A. Milan, J. Shi, D. Cremers, I. Reid, and S. Roth. Mot20: A benchmark for multi-object tracking in crowded scenes. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 3842–3852, 2020.
- [5] R. B. Fisher, Y.-H. Chen-Burger, D. Giordano, L. Hardman, and F.-P. Lin. Fish4knowledge: Collecting analyzing massive coral reef fish video data. *Springer*, 104, 2016.
- [6] A. Geiger, P. Lenz, C. Stiller, and R. Urtasun. Are we ready for autonomous driving? the kitti vision benchmark suite. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 3354–3361, 2012.
- [7] Q. Guo, W. Feng, C. Zhou, R. Huang, L. Wan, and S. Wang. Learning dynamic siamese network for visual object tracking. In *Proceedings of the IEEE international conference on computer vision*, pages 1763–1771, 2017.
- [8] S. Gupta, P. Mukherjee, S. Chaudhury, B. Lall, and H. Sanisetty. Dftnet: Deep fish tracker with attention mechanism in unconstrained marine environments. *IEEE Transactions on Instrumentation and Measurement*, 70:1–13, 2021.

- [9] G. Koch, R. Zemel, R. Salakhutdinov, et al. Siamese neural networks for one-shot image recognition. In *ICML deep learning workshop*, volume 2. Lille, 2015.
- [10] L. Leal-Taixé, A. Milan, I. Reid, S. Roth, and K. Schindler. Motchallenge 2015: Towards a benchmark for multi-target tracking. In *Proceedings of the IEEE International Conference on Computer Vision Workshop*, pages 1–7. IEEE, 2015.
- [11] J. Li, W. Zhang, Q. Wang, and L. Zhang. Trackformer: Multi-object tracking with transformers. *arXiv preprint*, 2022. arXiv:XXXX.XXXXX.
- [12] A. Milan, L. Leal-Taixé, I. Reid, S. Roth, and K. Schindler. Mot16: A benchmark for multi-object tracking. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 4096–4104, 2016.
- [13] H. Pirsiavash and D. Ramanan. Globally-optimal greedy algorithms for tracking a variable number of objects. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 1201–1208, 2011.
- [14] J. Redmon and A. Farhadi. Yolo9000: better, faster, stronger. *arXiv preprint*, 2017. arXiv:1612.08242.
- [15] R. Smith and T. Drummond. Object tracking using a mixture of gaussian processes. *International Journal of Computer Vision*, 62(3):283–308, 2005.
- [16] Smyth and S. Nebel. Passive integrated transponder (pit) tags in the study of animal movement. *Nature Education Knowledge*, 4:4, 01 2013.
- [17] L. Wang, E. Xie, X. Ding, S. Zhang, X. Sun, L. Yuan, Y. Xiong, Y. Song, and P. Luo. Transformertrack: A simple transformer-based framework for real-time multiple object tracking. In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision*, pages 2694–2703, 2022.
- [18] L. Wen, D. Du, H. Zhang, Q. Liu, X. Qi, S. Lyu, and Q. Tian. Ua-detrac: A new benchmark and protocol for multi-object detection and tracking. In *Proceed.*
- [19] N. Wojke, A. Bewley, and D. Paulus. Simple online and realtime tracking with a deep association metric. In *Proceedings of the IEEE International Conference on Image Processing*, pages 3645–3649. IEEE, 2017.
- [20] H. Zhang, C. Xiao, X. Ji, Y. Wei, and W. Zhang. Tractrans: Transformer-based object tracking. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 1617–1626, 2021.