KTH ROYAL INSTITUTE
OF TECHNOLOGY

Degree Project in Computer Science and Engineering

Second cycle, 30 credits

# Tracking with Joint-Embedding Predictive Architectures

Learning to track through representation learning

**RICKARD MAUS**

Stockholm, Sweden, 2023

# Tracking with Joint-Embedding Predictive Architectures

## Learning to track through representation learning

RICKARD MAUS

Degree Programme in Computer Science and Engineering
Date: October 1, 2023

Supervisors: Patric Jensfelt, Helmut Prendinger
Examiner: Hossain Azizpour
School of Electrical Engineering and Computer Science
Host organization: NII
Swedish title: Spårning genom Prediktiva Arkitekturer med Gemensam Inbäddning
Swedish subtitle: Att lära sig att spåra genom representations inlärning

# Abstract

Multi-object tracking is a classic engineering problem wherein a system must keep track of the identities of a set of a priori unknown objects through a sequence, for example video. Perfect execution of this task would mean no spurious or missed detections or identities, neither swapped identities. To measure performance of tracking systems, the Higher Order Tracking Accuracy metric is often used, which takes into account both detection and association accuracy.

Prior work in monocular vision-based multi-object tracking has integrated deep learning to various degrees, with deep learning based detectors and visual feature extractors being commonplace alongside motion models of varying complexities. These methods have historically combined the usage of position and appearance in their association stage using hand-crafted heuristics, featuring increasingly complex algorithms to achieve higher performance tracking. With an interest in simplifying tracking algorithms, we turn to the field of representation learning. Presenting a novel method using a Joint-Embedding Predictive Architecture, trained through a contrastive objective, we learn object feature embeddings initialized by detections from a pre-trained detector. The results are features that fuse both positional and visual features.

Comparing the performance of our method on the DanceTrack and MOT17 datasets to that of the most performant heuristic-based alternative, Deep OC-SORT, we see a significant improvement of 66.1 HOTA compared to the 61.3 HOTA of Deep OC-SORT on DanceTrack. While the method lags behind the state of the art, which follows the tracking-by-attention paradigm, it presents a novel approach and brings with it a new avenue of possible research. **WIP**

## Keywords

Contrastive learning, Joint-Embedding predictive architectures, Multi-object tracking, Representation learning

# Sammanfattning

Leaving this for later.

## Nyckelord

Kontrastiv inlärning, Prediktiva arkitekturer med gemensam inbäddning, Spårning av flera objekt, Representations inlärning

# Acknowledgments

I would like to thank Augustin and Nicolas for the great conversations discussing this topic of research during my stay at NII. Nicolas in particular was instrumental in giving me new perspectives on the process that underlined this thesis. I would also like to thank the National Institute of Informatics, who provided me with both funding and the hardware needed to conduct my research. Furthermore, I would like to acknowledge the freedom my supervisor at NII, Professor Helmet Prendinger, gave me in pursuing my own topic of research. Everyone at Prendinger's lab made me feel very welcome. Thank you all!

Stockholm, October 2023
Rickard Maus

# Contents

# Chapter 1

# Introduction

Multi-Object Tracking has long stood as an active field of research, with numerous applications ranging from surveillance of people and vehicles to medical imaging [1]. Due to its wide range of applications, multi-object tracking has been applied in various settings such as: radar, stereo vision, monocular vision etc. This thesis concerns itself more specifically with the task of monocular vision-based multi-object tracking, which has broadly seen an increase in performance with the introduction of deep learning based approaches. Henceforth, for the sake of brevity, we will only refer to monocular vision-based multi-object tracking as multi-object tracking.

## 1.1 Problem Background

Difficult scenarios pose challenges to current multi-object tracking (MOT) methods through uniform appearance of objects and increasingly advanced levels of occlusion, leading to potential issues of ID switching and prematurely initializing new or removing existing ones. The current leading paradigm of MOT is that of tracking-by-detection [2], in which the task is separated into two subtasks of i) object detection and ii) association of detections to known tracks or initialization of new ones. While newer deep learning-based methods have shown increasing robustness to difficult tracking conditions, a performance gap between the detection and association part of the tracking-by-detection pipeline still remains on some benchmarks [2].

To find better methods for association we turn to the field of representation learning, which attempts to learn some function that takes some signal, such as a picture, and transform it into some useful vector-space representation. While many prior works have integrated representation learning to varying degrees,

for example through re-identifying objects through their visual appearance, the implementations often apply techniques in combination with non-trivial heuristics for association. This thesis aims to introduce a novel approach to the problem of MOT by reframing the association task as a whole as one of representation learning, and in doing so, produce a performant *and* conceptually simple tracking method.

## 1.2 Objective

Prior to the great advances of deep learning, machine learning methods often relied on hand-crafted algorithms to extract features of data such as images and text. With the advent of performant deep learning methods, the designed methods were almost entirely supplanted by deep neural networks that learned to extract meaningful features, through various forms of learning.

Practically all except a few very recent MOT methods fall into the hand-crafted category when it comes to performing associations, more or less, building on increasingly complex heuristics for association. It is our view that this presents an opportunity to re-evaluate the current methods of MOT, and try to create a method that forgoes the complexities of engineered methods in favor of simpler, learning-based methods. As such, the objective of this thesis is to investigate if employing contemporary representation learning methods can improve over existing more "hand-crafted" tracking methods.

## 1.3 Research Question (WIP)

The research question for this thesis can be summarized as follows:

- *How well does a multi-object tracking method trained through contemporary representation learning methods perform compared to other more heuristic-based methods?*

## 1.4 Research Methodology

For this thesis, we first performed a literature study to better understand the history and current landscape of MOT and representation learning. We then decided on what datasets and metrics to use when evaluating our method by following what other contemporary publications have chosen, to enable broad comparisons with existing methods. Finally, we iterated on the design of the

novel method, building from what was learned in the literature study, and evaluating on the chosen datasets; admittedly, the process of designing was performed in an iterative, yet ad-hoc fashion.

## 1.5 Delimitations

In this thesis we will be presenting and evaluating a novel tracking method on chosen datasets. We will not be running experiments validating results of other methods, and will instead opt to use those reported by the original authors, or possibly those of re-implementations. Due to time and resource constraints, we will not be performing ablation studies on our novel method; it is possible that this will be done in a future publication. As MOT is a highly crowded research field, we will not be covering every relevant method in the related work section, rather choosing to pick a few important methods that highlight the progress of the field as a whole.

## 1.6 Structure of the thesis

The structure of this thesis follows a conventional format. In chapter 2 we will give you some background on MOT and some of the representation learning methods that will be used. In addition we will give an overview of some of the related work over the past decade or so in MOT. Chapter 3 will discuss MOT metrics used for evaluation, detail the novel method proposed in this thesis and how we train and perform tracking with it, along with presenting what experiments and datasets we will be evaluating on. After this, Chapter 4 will show the results of the experiments detailed in Chapter 3. Finally, we end on discussion of the results and conclusions of the thesis, in chapter 5 and 6 respectively.

# Chapter 2

# Background

This chapter provides a brief overview of the problem of multi-object tracking (MOT) and the tracking-by-detection paradigm. In addition, it covers Joint-Embedding (Predictive) Architectures and representation learning, methods of which are applied by this thesis to the task of MOT. Later, in the related work section, landmark and related research in the areas of MOT are presented.

## 2.1 Multi-Object Tracking

Multi-object tracking is a significant and longstanding research problem in the field of computer vision and artificial intelligence. Broadly speaking, it involves the identification and tracking of multiple objects over time in a sequence of images or videos. The "objects" could be anything, ranging from vehicles in traffic videos to cells in biomedical images, and the application domains for MOT are similarly wide-ranging, encompassing areas such as surveillance, autonomous vehicles, sports analytics, and cellular biology, among others.

The MOT problem is challenging due to several complicating factors. For instance, object occlusions, where one object obscures another from view, can make it hard to keep track of individual objects. Changes in object appearance due to alterations in lighting and perspective can also pose difficulties. In addition, when the number of objects is not known a priori or changes over time, the tracking system must be able to handle the dynamic creation and deletion of tracks.

A fundamental task in MOT is data association, which refers to the problem of matching detected objects across different frames. A correct data association implies that each object's identity is correctly maintained

over time. The data association problem becomes increasingly complex as the number of objects and the number of frames increase, leading to a combinatorial explosion in potential matchings. Various strategies have been proposed to solve the data association problem in MOT, including optimization methods and learning-based approaches.

### 2.1.1 Tracking-by-detection

Among tracking methods, one of the more popular approaches is called tracking-by-detection. This method tackles the MOT problem in a two-step process: first detecting the objects in each frame and then associating those detections across frames to form tracks.

The first stage, object detection, involves the use of an object detector to identify instances of the object of interest in each frame. Advances in deep learning over the past decade have led to the development of increasingly accurate object detectors, such as the family of two-stage detector R-CNN [3] models (Region-based Convolutional Neural Networks) and single-stage YOLO [4] (You Only Look Once) models. These models are trained to recognize a variety of objects from bounding box-annotated images, and can thus be used to find objects in video frames.

The second stage, data association, matches the detected objects across different frames. This involves solving the correspondence problem, i.e., determining which object in frames $\{1, \ldots, t\}$ corresponds to which object in frame $t+1$. The association problem can be approached as an optimization problem, where the goal is to minimize the overall cost of matching, based on a cost function that measures the dissimilarity between objects. This cost function could incorporate multiple factors, such as the spatial distance between objects, the similarity of their appearances, and the consistency of their motions.

Traditional tracking-by-detection methods often treat the detection and association stages as separate problems, and solve them sequentially. However, more recent approaches increasingly adopt an end-to-end learning framework, where the detection and tracking tasks are jointly optimized in a unified model. These methods have the potential to further improve the performance of tracking-by-detection by exploiting the interplay between detection and tracking.

However, despite these advancements, tracking-by-detection still faces a number of challenges. For instance, false negatives and false positives can lead to track terminations and spurious tracks, respectively. It can also be sensitive

to the choice of the cost function in the data association stage, with methods resorting to increasingly complex heuristics to alleviate the aforementioned issues.

## 2.2 Joint-Embedding (Predictive) Architectures

In the seminal paper "A Path Towards Autonomous Machine Intelligence", LeCun [5] describes his vision of how one might build intelligent machines that are capable of planning. Among the many requirements laid out by the author, one highlights the need for strong, often self-supervised, representation learning methods.

Self-supervised representation learning has seen extensive research in the past few years as these methods, which do not require explicit labels and rather have the models learn from the structure of the data itself, have been shown to yield high quality representations that match or beat those of fully supervised methods [6]. A couple notable examples showcasing the breadth of self-supervised methods are: the GPT-series of large language models from OpenAI [7], and DINO and DINOv2 vision models from Meta's FAIR [6, 8]. In particular, GPT uses a generative pre-training objective of predicting the next token in a text sequence to learn the underlying semantic structure of text. Working in the domain of vision, DINO and DINOv2 devise a self-distillation approach that learns visual representations by providing two views of an image, with differing global and local crops, to a student and an exponential moving average teacher.

This thesis builds on a general class of representation learning methods highlighted in LeCun [5] called Joint-Embedding Architectures (JAE). The goal of JEA is to learn one or more encoders `Enc` that take a set of signals, say $\mathbf{x}$, $\mathbf{y}$, and map them to some embedding $\mathbf{s_x}$, $\mathbf{s_y}$ such that it minimizes an energy function $E$ for related signals. In other words, the energy function should have a low energy for embeddings of related signals and a large energy for unrelated signals. An example would be that the embedding of a picture of a dog and the embedding of the sound of a dog barking should yield a low energy while an image of a cat and the sound of a dog barking should yield a relatively higher energy. By learning encoders that minimize $E$, we are able to perform various forms of optimization at inference time to find the signal that makes the most "sense" in the context of another. This idea is not new and has been leveraged to great effect by Radford *et al.* [9] for learning shared

image-text embedding spaces, and more recently by Girdhar *et al.* [10] to learn a shared embedding space across six modalities: images, videos, text, audio, depth, thermal, and IMU data.

A further extension of JEA is Joint-Embedding Predictive Architectures (JEPA) which, in addition to the encoders used in JEA, consist of one or more predictors `Pred`. In a simple case, say we have a shared encoder `Enc` and two signal $\mathbf{x}_t$, $\mathbf{x}_{t+1}$ such that $\mathbf{x}_{t+1}$ follows $\mathbf{x}_t$ temporally. Much like in JEA, the encoder maps the two signals to embeddings $\mathbf{s}_{\mathbf{x}_t}$, $\mathbf{s}_{\mathbf{x}_{t+1}}$ in a shared embeddings space. However, unlike in JEA, JEPA applies `Pred` to project $\mathbf{s}_{\mathbf{x}_t}$ to some prediction $\hat{\mathbf{s}}_{\mathbf{x}_{t+1}}$. `Enc` and `Pred` are then learned by optimizing their parameters such that $\mathbf{s}_{\mathbf{x}_{t+1}}$ and $\hat{\mathbf{s}}_{\mathbf{x}_{t+1}}$ minimize some energy function $E$ if they are related. In addition, `Pred` may take a complementary conditioning variable $\mathbf{z}$, to help guide the prediction. Relating back to the example given, $\mathbf{z}$ may contain information regarding how large the temporal distance is between $\mathbf{x}_t$ and $\mathbf{x}_{t+1}$, facilitating a more expressive predictor.

Note that the example given considers signals that are temporally related, but that signals may be related in some other manner, i.e. spatially, without loss of generality. For example, Assran *et al.* [11] recently proposed Image-based Joint-Embedding Predictive Architecture (I-JEPA), where signals are patches of an image and are thus spatially related, rather than temporally. In essence, the predictor in the JEPA architecture can be thought of as the "world model" which learns the relationships between signals. In the context of signals that are temporally related, the predictor then learns to model causality.

Why not perform the prediction in signal space instead of in latent space? The former can be seen as modelling the problem using a Generative Architecture. There are a few advantages to JEPA as when compared to Generative Architectures [11] such as the proposed method of Masked Auto Encoders by He *et al.* [12], which learns embeddings by masking random patches of data and trying to fill in what is missing. The main reason is that signal space is usually noisy and contains a lot of randomness. A good example is Masked Auto Encoders when applied to video and used to predict future frames. Assume a video of a golfer just having started a swing with the driver. How they may swing, hit, or miss the ball is down to variability, making it difficult to predict the video in pixel-space due to uncertainty. What we are able to prognosticate is that the golfer will most likely keep swinging, but not necessarily how or with what consequences. In essence then, JEPA encourages learning an embedding space that is robust to features of the signal that are inappropriate for prediction. Be that prediction of what comes next in time, or what lies close in space.

### 2.2.1 Contrastive predictive coding

When training JEA and JEPA models, care must be taken with what objective is used. Clearly, the energy function must be part of the objective, but this is insufficient to hinder the models from collapsing the embedding space into a constant and equal output. LeCun [5] outlines two approaches to solving this "representation collapse". The first are regularization-based methods. The underlying intuition for these is that we should make the embedding space as information dense as possible by pushing down on the energy function where the representations lie, while at the same time pushing up on the energy function everywhere. One such method is VICReg by Bardes *et al.* [13], which consists of a squared euclidean distance loss for the energy function, a variance loss for regularization to avoid representations collapsing to a constant output, and a covariance loss for regularization to avoid what the authors call "dimensional" collapse. For this thesis, we opt not to use regularization-based methods.

The other category of methods are contrastive methods. These frame the representation learning task with the objective of pushing the embeddings of positive pairs, i.e. related signals, to have low energy, while negative pairs, i.e. unrelated signals, have high energy. A notable example of such a method is SimCLR by Chen *et al.* [14], which introduced the Normalized Temperature-scaled Cross-Entropy (NT-Xent) objective that will be used later in the work of this thesis. Given that $\text{sim}(\mathbf{u}, \mathbf{v})$ is the cosine similarity between two vectors $\mathbf{u}$, $\mathbf{v}$, the NT-Xent loss for a positive pair $(i, j)$ can be expressed as:

$$\ell_{i,j} = -\log \frac{\exp\left(\text{sim}\left(\mathbf{z}_i, \mathbf{z}_j\right)/\tau\right)}{\sum_{k=1}^{2N} 1_{[k \neq i]} \exp\left(\text{sim}\left(\mathbf{z}_i, \mathbf{z}_k\right)/\tau\right)} \tag{2.1}$$

where $1_{[k \neq i]} \in [0, 1]$ is the indicator function that is equal to one when $k \neq i$ and zero otherwise. The temperature-scaling parameter $\tau$ controls the spread of the embedding distributions. A small $\tau$ leads to a denser embedding space, while a large $\tau$ results in a more spread out distribution.

In this thesis we opt to use the contrastive methods, rather than regularization-based methods. Intuitively we choose this as we want our learned encodings to be pushed far away from each other and become as separable as possible. Combining the objective of contrastive learning with the predictive modelling of JEPA, one arrives at a method that can be described as Contrastive Predictive Coding. Note, however, that the method and formulation of Contrastive Predictive Coding in this thesis differs somewhat from that of the paper *Representation Learning with Contrastive Predictive*

*Coding* by Oord *et al.* [15], who present a probabilistic formulation different from that of the energy-based JEPA.

## 2.3 Related work

In this section we will touch on some of the notable developments in vision-based MOT throughout the past decade of research, starting off with the methods first to integrate deep learning based approaches. We will then cover recent methods that make use of an "object query" formulation to solve MOT.

### 2.3.1 Beginnings of deep learning-enhanced MOT

"Simple Online and Realtime Tracking" (SORT) [16] was one of the first methods to integrate deep learning approaches into the MOT pipeline. Bewley *et al.* [16] employed a Convolutional Neural Network-based (CNN) detector to identify objects in frames and simple linear Kalman Filters combined with the Hungarian Algorithm for detection association. The Kalman Filters emit predictions of the bounding box for the corresponding object in the next frame, after which the Intersection-over-Union (IoU) between detected and predicted bounding boxes is calculated to yield the complement of the cost matrix. In other words, the IoU is used as the association metric. Assignments are then made by solving the linear assignment with the Hungarian Algorithm by minimizing the cost with regards to the cost matrix.

Deep SORT by Wojke *et al.* [17] later introduced a deep learning-enhanced association metric that, in addition to SORT's original IoU, added visual feature similarity into the association stage by performing nearest neighbor search at inference time. This allowed Deep SORT to better deal with longer periods of occlusion and achieve lower amounts of identity switching on tracks, which posed an issue for the purely motion-based method of the original SORT. Deep SORT marks one of the first methods to make use of deep learning-based represention learning to improve the association-step in MOT.

Other methods, such as QDTrack by Pang *et al.* [18] and SMILEtrack by Wang *et al.* [19], make use of representation learning for MOT through the similarity learning paradigm. QDTrack employs similarity learning to learn a visual feature extractor, which is later used with nearest-neighbour search in the association stage of a regular tracking-by-detection pipeline, similar to that of Deep SORT. QDTrack learns the extractor by sampling visual patches densely between two frames, with a contrastive loss as the objective. Similarly,

SMILETrack reaches SOTA performance on MOT17 and MOT20 by learning a visual feature extractor through contrastive similarity learning on extracted visual object patches.

OC-SORT [20] takes a step back and moves away from representation learning, and focuses on heuristics for improving the performance of the Kalman Filter when occlusion occurs. In doing so OC-SORT achieved SOTA across multiple benchmarks, showing that motion-models still had room for improvement, especially when object visuals are uniform. In that vein, some works have experimented with replacing the standard Kalman Filter with more advanced deep learning-based motion models, such as simple RNNs, LSTMs and Transformer networks. An analysis by the authors of the DanceTrack dataset Sun *et al.* [21] show that LSTM-based motion models can yield marginal improvements in tracking accuracy over the Kalman Filter counterpart, concluding that further investigations into alternative motion models could be justified.

### 2.3.2 Query-based tracking methods

There exist a newer family of tracking methods that make use of what are called "object queries" and sometimes "detection queries", where each detected or tracked object is associated with a corresponding detection or track query that encodes the object in some form of vector representation. Note that the nomenclature of "query-based tracking" isn't standard, but that this is an umbrella-term I chose to summarize related works that frame the problem around using queries in some manner. The methods that make use of such a formulation are numerous, and the individual implementation details differ. I will here go through some notable query-based methods. All of the methods presented below make use of an architecture introduced by DETR [22], a Transformer-based detector. As the work in this thesis also makes use of DETR, this architecture will be discussed in chapter 3.

Both TransTrack [23] and TrackFormer [24] present joint-detection-tracking methods, employing object queries that encode spatio-visual features, and using these to regress detections. However, their association methods differ. TransTrack uses its queries from the prior frame to query new detection queries, after which a matching is performed through bipartite matching of bounding boxes, optimizing for high IoU. TrackFormer introduces what the authors call the *tracking-by-attention* paradigm. Track queries from the prior frame with a confidence score higher than a specified threshold are fed back into the DETR model to update the queries and regress the detections for the

corresponding objects in the new frame. To better deal with duplicate and lost tracks, TrackFormer employs Non-Maximal Suppression (NMS) and Re-Identification (ReID) heuristics.

Continuing the iterative nature of TrackFormer, MOTR, and it's followup MOTRv2 [25, 26], present SOTA performance by extending the learning context. Track queries, like in TrackFormer, are updated frame-by-frame to perform iterative prediction over time. Unlike TrackFormer, MOTR and MOTRv2 calculate their loss over a longer context of 5 frames, as compared to the 2 frames of TrackFormer. According to Zeng *et al.* [25], this leads to better temporal learning and allows them to forgo the NMS and ReID heuristics employed by TrackFormer. MOTRv2 [26] goes back to a tracking-by-detection paradigm. By including a YOLOX detector [27] for object detection priors, MOTRv2 improves on its detection accuracy while keeping the entire system end-to-end trainable and establishing a new SOTA on several benchmarks. Interestingly, both TrackFormer and MOTR/MOTRv2 perform tracking without explicit association.

## 2.4 Summary

The field of vision-based MOT has seen many different methods in the prior decade, with performance increases largely brought on by the advent of deep learning. One of the key takeaways when reviewing recent SOTA tracking methods is that the use of representation learning in the association step is highly prevalent. The motivation behind the work in this thesis is to further explore using representation learning for MOT, and in doing so, formulate the association task in it's entirety as one of representation learning. More specifically, this thesis will be applying contemporary representation learning methods, JEPA and contrastive learning, to the field of MOT. As such, the goal is to introduce a simple novel tracking method that performs well while forgoing the more complicated association heuristics used in other recent association-based trackers.

Note that while our novel method incorporates techniques developed from self-supervised learning research, it is actually trained in a supervised manner. We are applying these self-supervised techniques to a task that has traditionally been solved with supervised learning.

# Chapter 3

# Methods

This chapter discusses the choice of datasets and metrics that are be used for evaluation. In addition, the function, design and architecture of our novel tracking method *Ultima* is presented. Later, experiments to elucidate the performance of the method are discussed.

## 3.1 Research Process

As is standard in the field of MOT, the novel method presented in this thesis are evaluated with current standard tracking benchmarks. For evaluation, the metrics discussed in section 3.3 are used. These are chosen as they are standard within the field and give a good overview of how any given tracking method performs in each part of the MOT process.

## 3.2 Datasets

The datasets chosen to evaluate the method later presented in this thesis are that of the DanceTrack dataset by Sun *et al.* [21] and the MOT17 dataset by Dendorfer *et al.* [28]. For examples of both datasets, see figure **??**.

The DanceTrack dataset consists of 100 labeled video-camera group-dance videos publicly available on YouTube: 40 and 25 in the training and validation sets respectively with publicly available labels, with the remaining 40 in the test set whose labels are private for competitive reasons. The authors of DanceTrack propose this dataset as they note that other contemporary datasets are too easily solvable by object feature extraction due to objects being easy to distinguish visually and lacking complex motion, e.g. videos of a pedestrian


Figure 3.1: Examples from the DanceTrack and MOT17 datasets. Top left and right are the DanceTrack 26 and 81 sequences, respectively. Bottom left and right are the MOT17 sequences 04 and 11, respectively.

street. By comparison, the group-dance videos that constitute DanceTrack offer objects in scenes that are highly similar in visual appearance (due to matching clothing) with complex motions and frequent occlusions. As a result, prior SOTA methods performed comparatively worse on DanceTrack. Combined, the DanceTrack training and validations set consists of some 65k annotated frames, running at 20 frames per second.

The MOT17 dataset is part of a collection of datasets known as MOTChallenge. MOT17 in particular features primarily videos of pedestrianized streets in both indoor and outdoor settings, with moving and stationary cameras. Both the training and test set consist of 7 videos each, lacking a specified validation set. While the movement and visual features of objects in MOT17 are relatively simple compared to DanceTrack, the dataset can be difficult as it consists of scenes with many concurrent tracks in combination with frequent occlusion. The MOT17 dataset consists of around 5.3k annotated frames in the training set at 30 frames per second.

## 3.3 Evaluation Metrics

Throughout the long history of MOT there have been many metrics proposed for effectively evaluating the performance of tracking methods. In this

subsection, I will cover those that are commonly found in contemporary MOT research as well as their strengths and weaknesses.

### 3.3.1 Multi Object Tracking Accuracy

Multi Object Tracking Accuracy (MOTA) [29] was for sometime the go-to metric for evaluating tracking methods. It is quite easy to comprehend and is calculated as follows:

$$\text{MOTA} = 1 - \frac{\text{FP} + \text{FN} + \text{ID\_SW}}{\text{GT}}$$

- FP (False Positive): The number of bounding boxes that are incorrectly identified as objects.
- FN (False Negatives): The number of ground truth objects that are not detected.
- ID_SW (Identity Switches): The number of times an object changes its identity during tracking.
- GT (Ground Truth): The total number of ground truth object instances in the dataset.

The MOTA metric goes from 1 (best) to negative infinity (worst). Due to its relatively simple and aggregate nature, MOTA does not lend itself well to precise analysis of how well a tracker is performing each of tracking's individual tasks. One important point is that MOTA does not elucidate how well a method's detections are localized; optimally, a detection should perfectly overlap with it's ground truth. In addition, it does not balance well the performance of the detection and association, being rather biased toward detections.

### 3.3.2 Identity

The Identity is a collection of commonly used metrics for measuring how well a tracker maintains consistent identities for objects over time. The metrics are as follows:

- IDTP (Identity True Positives): The number of ground-truth objects that are correctly detected and identified, across all frames.

- IDFN (Identity False Negatives): The number of ground-truth objects that are missed by the tracker across all frames.
- IDFP (Identity False Positives): The number of spurious identities that do not correspond to a ground-truth object. E.g. incorrectly assigning new identity to an existing object or falsely declaring a new object.
- IDR (Identity Recall): A measure of how well ground-truth identities are correctly recalled.

$$\text{IDR} = \frac{\text{IDTP}}{\text{IDTP} + \text{IDFN}}$$

- IDP (Identity Precision): A measure of how well ground truth identities are correctly predicted.

$$\text{IDP} = \frac{\text{IDTP}}{\text{IDTP} + \text{IDFP}}$$

- IDF1 (Identity F1 Score): The harmonic mean of the IDR and IDP.

$$\text{IDF1} = \frac{2 \times \text{IDP} \times \text{IDR}}{\text{IDP} + \text{IDR}}$$

### 3.3.3 Higher Order Tracking Accuracy

The Higher Order Tracking Accuracy (HOTA) [30] metric is the preferred choice of metric in contemporary MOT research. The authors of HOTA motivate its introduction through showing that the previously preferred MOTA [29] is biased towards detection performance, and does not accurately reflect a tracker's association performance. As is evident by its name, HOTA is a higher order metric that makes use of several component metrics. Notably, the HOTA metric is calculated through the composition of the following components:

- LocA (Location Accuracy): A measure of how accurate a detection is in terms of localization; used when deriving the following two.
- DetA (Detection Accuracy): A measure of how accurate the detections are in terms of TP, TN, FP, FN.
- AssA (Association Accuracy): A measure of how accurate the association of detections are to their corresponding tracks.

The final HOTA score is a geometric mean of DetA and AssA, placing importance on both areas of MOT. In turn, DetA and AssA are decomposed into individual precision and recall components: DetPr, DetRe, AssPr, AssRe. This decomposition allows for both a holistic view of a tracking method's performance, as well as a more detailed view permitting analysis of where a method performs well and where there is room for improvement. I will not detail here the formulas for calculating HOTA as they are quite numerous and require quite some explanation. Curious readers are referred to the original paper on HOTA by Luiten *et al.* [30].

## 3.4 Proposed Method (Design)

In this section we describe our proposed tracking method Ultima. We first give an overview of the method, before describing in detail the component encoder `Enc` and a predictor `Pred`. Figure 3.2 gives an overview of how our method works, and figures 3.3 and 3.4 show the design of `Enc` and `Pred`, respectively. Following that, training and inference details will be provided.

### 3.4.1 Ultima overview

Ultima is trained through a contrastive objective detailed in 2.2.1 and learns to model an object's positional and visual appearance in a shared embedding space. The method builds on the Joint-Embedding Predictive Architectures detailed in 2.2 and consists of two main components, an encoder `Enc` and a predictor `Pred`. `Enc` takes detections from a pre-trained detector and encodes them while enriching with visual features to create what we call encoded proposal queries, which is a latent representation of each detected object. For each currently tracked object, `Pred` uses the object's past encoded proposal queries to predict its latent representation in the next time-step. Tracking is then performed by matching the predictions of `Pred` to the encoded proposals of `Enc` through cosine similarity.

### 3.4.2 The encoder

The encoder `Enc` is tasked with learning object-level representations and is essentially the same architecture as DETR [22], part of the family of Transformer-based detectors. The DETR design consists of a ResNet-50 convolutional neural network backbone for image-feature extraction, followed by a Transformer encoder to enrich image features. In practice we use the

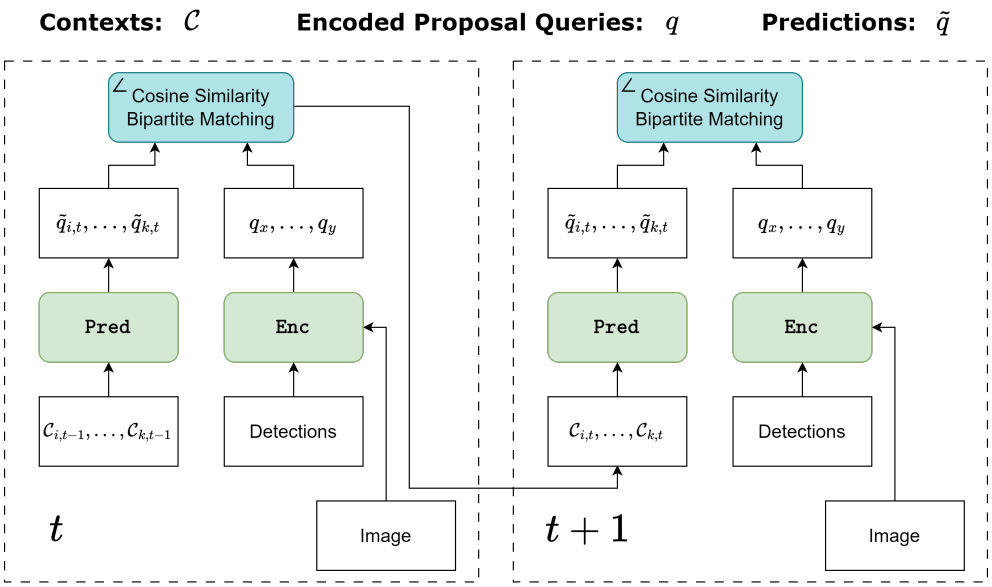


Figure 3.2: Overview diagram of the Ultima method across two time-steps.

features output from stage-4 of the ResNet. A Transformer decoder then takes learned detection queries and performs cross-attention with the image features from the encoder to output final detection predictions. Note, however, that rather than the learned detection queries used for detection by DETR, we use the bounding boxes and detection confidences from a detector to initialize what we designate as *proposal queries*. Proposal queries are fed to the decoder to produce *encoded proposal queries*, which encode both spatial and visual features. This is in line with the design choices of MOTRv2 [26]. As such, the tracking method presented in this thesis follows the tracking-by-detection paradigm.

Normalized bounding boxes and their corresponding confidences are first embedded before being fed as proposal queries to the decoder. For both bounding box and confidence score we use learnable Fourier features, introduced by Li *et al.* [31], which perform well when embedding sparse multidimensional spatial features. Similarly, we create a mesh grid of pseudo bounding boxes that are embedded with the same learned Fourier features module and added to the image-features before being passed to the image Transformer encoder. We add this positional encoding as the Transformer is an otherwise permutation invariant architecture.

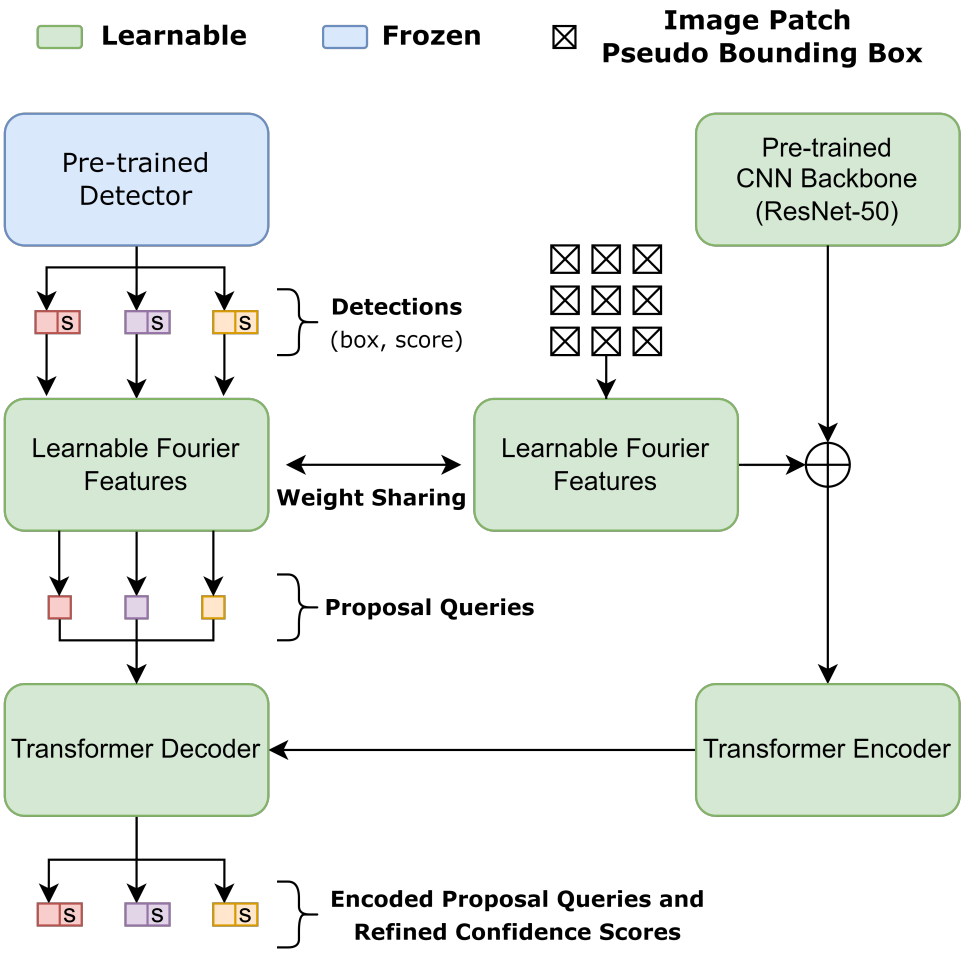


Figure 3.3: Diagram of the architecture of `Enc`. `Enc` produces encoded proposal queries for each frame using the image, and proposed detections from a pre-trained detector.

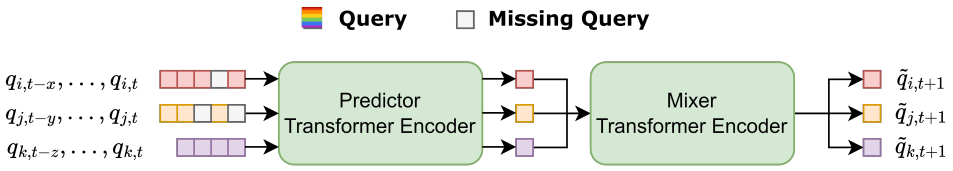


Figure 3.4: Diagram of the architecture of `Pred`. `Pred` predicts the future latent representation of objects from their prior context.

### 3.4.3 The predictor

The predictor `Pred` is comparatively more simple. Modelled by a Transformer encoder, its task is predicting an object's representation in the next time-step. More formally, for each object $i$, `Pred` accepts a set of sequential time-position encoded proposal queries $\{q_{i,0}, \ldots, q_{i,t}\}$ from time $0$ to $t$, $t \in [0, \text{max\_context}]$, and makes a prediction $\tilde{q}_{i,t+1}$ of it's representation $q_{i,t+1}$ at time $t+1$. Time-position encoding is accomplished by adding sine-positional encodings [32]. Since the predictions of each object at time $t$ are otherwise unaware of each other, we add an additional "mixing" Transformer encoder between predicted object representations to encourage more discriminative predictions.

### 3.4.4 Training specifics

As the work in this thesis focuses on establishing a novel tracking method and follows the tracking-by-detection paradigm, we opt to use detections from a pre-trained detector. To be specific, we use the exact same detections as MOTRv2 [26], who in turn borrow pre-trained YOLOX [27] detector weights from DanceTrack [21] and ByteTrack [33] for the DanceTrack and MOT17 datasets respectively. To maintain a high detection recall, we also follow Zhang *et al.* [26] in keeping any detection with a confidence score greater than 0.05.

For learning `Enc` and `Pred` we employ the contrastive loss NT-Xent, with $\tau = 0.07$, detailed in 2.2.1. Formally, for each time-step $t$ we apply NT-Xent over the set of detected objects $\{j, \ldots, k\} = \mathcal{O}_{t,\text{vis}}$ between encoded proposal queries $\{q_{j,t}, \ldots, q_{k,t}\}$ and predicted representations $\{\tilde{q}_{j,t}, \ldots, \tilde{q}_{k,t}\}$. The set $\mathcal{O}_{t,\text{vis}}$ is constructed by performing a bipartite matching between detected and ground truth annotated bounding boxes, using IoU as the scoring criterion and allowing only assignments that have a minimum of 0.5 IoU. The proposal queries at time $t$ and those prior, used to derive the predictions, are initialized with the matched detected bounding boxes and their corresponding confidence scores. For the context provided to the predictor, we set a max context $M_{mc}$ of 24.

To enable a more discriminative selection of proposal queries, we add an additional binary classification task. Following the encoding of proposal queries, a multilayer perceptron outputs a refined confidence score indicating how likely an encoded proposal is to belong to a true positive detection, i.e. would have a ground truth match. This turns the detection-encoding pipeline into what is essentially a two-stage detector, where `Enc` serves as a discriminator. Thus, in practice, we also encode unmatched detections that go

unused during the contrastive objective. To get more robust classification, we employ a focal loss [34] to give difficult samples more weight. To be precise, we use $\alpha = 0.5$ (both classes are equally weighted) and $\gamma = 2.0$.

As the work in this thesis builds off the MOTRv2 codebase we follow their data-augmentations, which include Hue-Saturation-Value (HSV) augmentations as well as random-resize and crop. In addition, clips are constructed with a uniformly sampled stride of between 1 and 5 for both DanceTrack and MOT17, to simulate varying frame rates. For more details on these augmentations, please check the provided code repository. Images are resized during testing to have a size of 800px on the smallest dimension while maintaining aspect ratios up to a maximum of 1333px on the largest dimension. As the image sizes vary across the sequences of the datasets used, we refer curious readers to Dendorfer *et al.* [28] and Sun *et al.* [21] for specifics.

We use the Adam optimizer with a learning rate of $2 \times 10^{-5}$ and $2 \times 10^{-4}$ for the CNN backbone and rest of the model respectively. We train the model for 10 epochs on the DanceTrack dataset, during the last of which we drop the learning rate by a factor of 10. For the MOT17 dataset, we train for a total of 50 epochs, dropping the learning rate by a factor of 10 beginning at epoch 40. All training is performed with causal masking of the predictor and at half-precision with the `bfloat16` format to lower memory usage and allow for longer contexts. If there is no corresponding detection for an object at time $t$, and therefore no encoded proposal query, we pad the context with a zero-vector at time $t$ to indicate a missed detection. All training was performed on a single RTX 4090 graphics card, using gradient accumulation and a batch size of 8.

### 3.4.5 Inference

For each frame we collect detections with a confidence greater than 0.05 from the same pre-trained detector used during training. These are then embedded and encoded using `Enc` to produce encoded proposal queries. Using the refined proposal scores we discard detections, and their corresponding encoded proposal queries, that fall below a threshold $\mu$. Following this, for each frame, we perform the following simple algorithm:

1. If there are no current tracks, surviving encoded proposal queries are used to initialize new tracks.

2. In the case that any current track $i$ exists, we use the track's context

$\{q_{i,0}, \ldots, q_{i,t}\}$ to predict $\tilde{q}_{i,t+1}$ with `Pred`.

3. Assignments of detections are made by maximizing a bipartite matching between predictions and encoded proposal queries, using the cosine similarity as the criterion with a minimum similarity threshold $\theta$.

4. Encoded proposals are used to update their corresponding matched tracks.

5. Unmatched tracks have a zero-vector added to their context to indicate a missed detection.

6. If unmatched detections remain, these are used to initialize new tracks.

7. Tracks that have not received any matches within the last $M_{mc}$ frames are culled.

Those with prior familiarity of MOT literature will note that this is a relatively simple association algorithm.

### 3.4.6 Model architecture summary

We introduce, train and test only a *base* version of the model discussed so far. With more time and resources, we would have wished to provide models of varying scales and performance. To get a better overview of the *base* model, please see table 3.1 for a summary of model components and capacity.

Table 3.1: Overview of *base* model architecture, layers and parameter count.

| Module | Type | Layers | Parameters |
|---|---|---|---|
| `Enc` ResNet-50 Backbone | CNN | 50 | 23,454,912 |
| `Enc` Learnable Fourier Feat. | MLP+Fourier | 3 | 49,408 |
| `Enc` Image Encoder | Transformer | 6 | 4,738,560 |
| `Enc` Proposal Decoder | Transformer | 6 | 6,567,809 |
| `Pred` Predictor Encoder | Causal Transformer | 6 | 4,738,560 |
| `Pred` Mixer Encoder | Transformer | 3 | 2,369,280 |
| Total: | | 73 | 42,393,665 |

## 3.5 Experimental Design

To effectively compare the performance of our novel method to that of others, we follow convention and report the metrics specified in 3.3 on the test set of both DanceTrack and MOT17. We test models trained according to what is detailed in 3.4.4, by running the inference algorithm mentioned in 3.4.5. The thresholding parameters $\mu$ and $\theta$ are chosen following a grid search on the DanceTrack validation set, optimizing for the highest HOTA metric. For evaluation on the validation set, we use `TrackEval` by Jonathon Luiten [35].

To gauge how well the method presented in this thesis generalizes across domains, we also test the model trained on DanceTrack against the test set of MOT17, and vice versa. While this practice is non-standard in MOT research, we believe that cross-domain generalization performance is of great interest. Before testing on the opposite test set, we use the opposite validation set to find the optimal values of $\mu$ and $\theta$. In the case of MOT17, since there is no designated validation set, we use the training set instead. We do this as we believe it to be prudent to tune the hyperparameters before applying to any new domain. Note that we do **not** fine-tune or train the model further.

In addition to the quantitative experiments, we also perform qualitative comparisons of the learned object-level representations, i.e. encoded proposal queries, to that of pre-trained ReID features. More specifically, we compare the object-level representations of our method to that of those produced by OSNet-AIN by Zhou *et al.* [36, 37], using the `torchreid` library [38]. For a fairer comparison, we opt to compare features from the dataset each model was *not* trained on, i.e. the MOT17 features are produced by `Enc` of the model trained on DanceTrack. To more easily grasp the high-dimensional space, we employ the non-linear dimensionality reduction technique t-SNE [39] to project representations to $\mathbb{R}^2$, while aiming to preserve local structure. For this we use the GPU-accelerated `tsnecuda` by Chan *et al.* [40]. So as to not clutter the plots, we limit ourselves to the embeddings of at most 20 objects in the first 200 frames of each sequence, which corresponds to 10 seconds of DanceTrack and 6.67 seconds of MOT17. To simulate more realistic tracking conditions, we use detections from the same pre-trained YOLOX detector used during training. We choose detections by performing a bipartite matching between ground truth annotations and detections, with a minimum of 0.5 IoU.

# Chapter 4

# Results WIP

We first present what hyperparameters were found and used for testing, as well as the quantitative results of benchmarking the proposed method on the DanceTrack and MOT17 test set, while comparing to prior and current SOTA methods. The tables also include the results of the cross-domain experiments. Subsequently, the qualitative analysis of the learned object embeddings is presented.

## 4.1 Hyperparameter Search

For the model trained on the DanceTrack dataset, we found that the best values for $\mu$ and $\theta$ on the validation set were 0.5 and 0.25, respectively. When testing the same model on the ... **to be continued**

## 4.2 Quantitative Results

When comparing our method to others we choose to also include methods that have as of yet only appeared in pre-print, to better take into account recent MOT research. As can be seen in table 4.1, the method presented in this thesis performs quite well on the DanceTrack dataset, though it does not reach SOTA. The largest differentiating factor on the DanceTrack dataset, among all methods, is the Association Accuracy. For this metric our method falls 6 points short of the leading method MOTRv2, 53.0 vs 59.0 points. Comparing to that of Deep OC-SORT, a method that combines the Kalman filter motion models and heuristics of OC-SORT with deep learning-based ReID features to perform assocation, we see that our less engineered method outperforms it by a wide margin of 7.2 points on AssA.

In addition, the IDF1 score also seems to vary greatly between the methods presented. This is not surprising, taking the AssA results into account, as IDF1 generally measures how well objects are tracked in terms of identity. On this metric our method falls 5.2 points behind the leading method CO-MOT.

Despite being trained on an order of magnitude less data (in terms of frames), and a dataset which is quite different from that of DanceTrack, the model trained on MOT17 delivers compelling cross-domain performance. While our cross-domain model lags behind that of Deep OC-SORT by 3.9 AssA and 3.7 IDF1 score on DanceTrack, we outperform other prior deep learning-based methods such as MOTR, QDTrack and TransTrack on both metrics.

Table 4.1: Results of prior and current SOTA methods on the DanceTrack test set. Results of TransTrack, Bytetrack, QDTrack, MOTR, and OC-SORT are provided by DanceTrack [21]. * indicates the model trained on the MOT17 training set. For the sake of readability, all values are increased by a factor of 100.

| Methods | HOTA | DetA | AssA | MOTA | IDF1 | Extra Data |
|---|---|---|---|---|---|---|
| TransTrack [23] | 45.5 | 75.9 | 27.5 | 88.4 | 35.7 | ✗ |
| ByteTrack [33] | 47.7 | 71.0 | 32.1 | 89.6 | 53.9 | ✗ |
| QDTrack [18] | 54.2 | 80.1 | 36.8 | 87.7 | 50.4 | ✗ |
| MOTR [25] | 54.2 | 73.5 | 40.2 | 79.2 | 51.5 | ✗ |
| OC-SORT [20] | 55.1 | 80.3 | 38.3 | 92.0 | 54.6 | ✗ |
| MOTR [25] | 54.2 | 73.5 | 40.2 | 79.2 | 51.5 | ✗ |
| Deep OC-SORT [41] | 61.3 | 82.2 | 45.8 | **92.3** | 61.5 | ✗ |
| MeMOTR [42] | 68.5 | 80.5 | 58.4 | 89.9 | 71.2 | ✗ |
| CO-MOT [43] | 69.4 | 82.1 | 58.9 | 91.2 | **71.9** | ✓ |
| MOTRv2 [26] | **69.9** | **83.0** | **59.0** | 91.9 | 71.7 | ✗ |
| Ultima (ours)* | 58.5 | 81.5 | 42.1 | 90.3 | 57.8 | N/A |
| Ultima (ours) | 66.1 | 82.6 | 53.0 | 91.1 | 66.7 | ✗ |

**Talk about MOT17 results here.**

## 4.3 Qualitative Results

We choose here to only include a subset of the sequences available in DanceTrack and MOT17. To see the visualizations of all sequences, see appendix A.

Looking at the visualizations in figures 4.1 and 4.2 we can see that the

Table 4.2: Results of prior and current SOTA methods on the MOT17 test set. For the sake of readability, all values are increased by a factor of 100.

| Methods | HOTA | DetA | AssA | MOTA | IDF1 |
|---|---|---|---|---|---|
| TransTrack [23] | 54.1 | 61.6 | 47.9 | 74.5 | 63.9 |
| ByteTrack [33] | 63.1 | 64.5 | 62.0 | 80.3 | 77.3 |
| QDTrack [18] | 53.9 | 55.6 | 52.7 | 68.7 | 66.3 |
| MOTR [25] | 57.8 | 60.3 | 55.7 | 73.4 | 68.6 |
| OC-SORT [20] | 63.2 | / | 63.2 | 78.0 | 77.5 |
| MOTRv2 [26] | 62.0 | 63.8 | 60.6 | 78.6 | 75.0 |

embeddings produced by our method are generally quite dense, a bit more so than those of the ReID model. We can't say whether this local structure is necessarily beneficial or not for tracking performance. Another quality of our embeddings is the apparent wandering trails, which are especially noticeable in the MOT17-02 and DanceTrack 41 sequences. We posit that this attribute of the t-sne feature projections is likely due to the positional information that our embeddings are enriched with, which the ReID features lack.

Comparing the visualizations of DanceTrack and MOT17, we can see that the separability of features seem more difficult in DanceTrack. This should not be surprising as DanceTrack was intentionally constructed with uniform appearance and complex motion in mind. Nonetheless, the learned features of our method seem more easily separable compared to the ReID features in both datasets, with a clearer advantage in the DanceTrack sequences shown in 4.1.

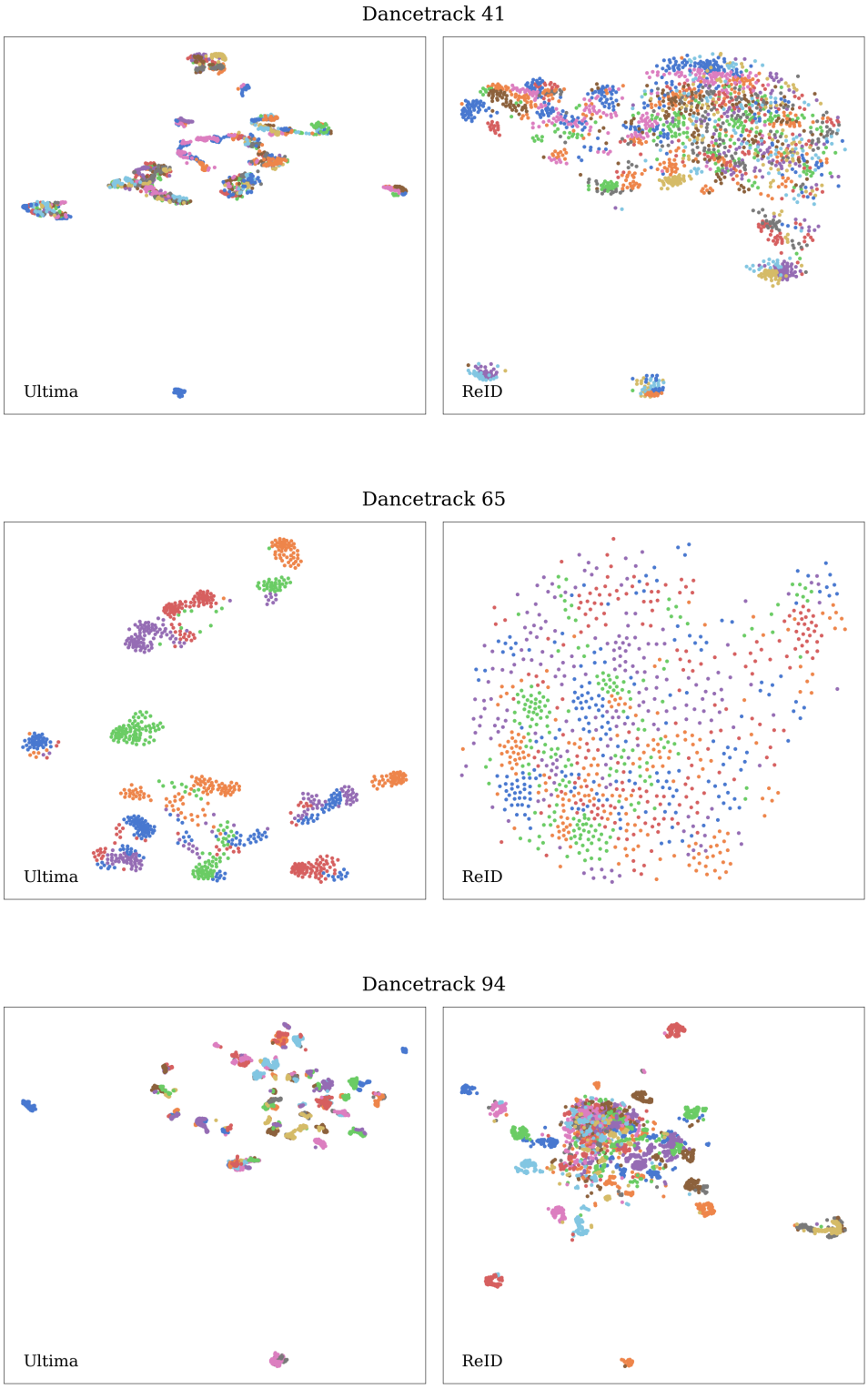


Figure 4.1: A t-sne visualization of DanceTrack validation set embeddings of our proposed method *Ultima*, trained on MOT17, compared to that of a pre-trained ReID model OSNet-AIN (left). Each object has an associated color.

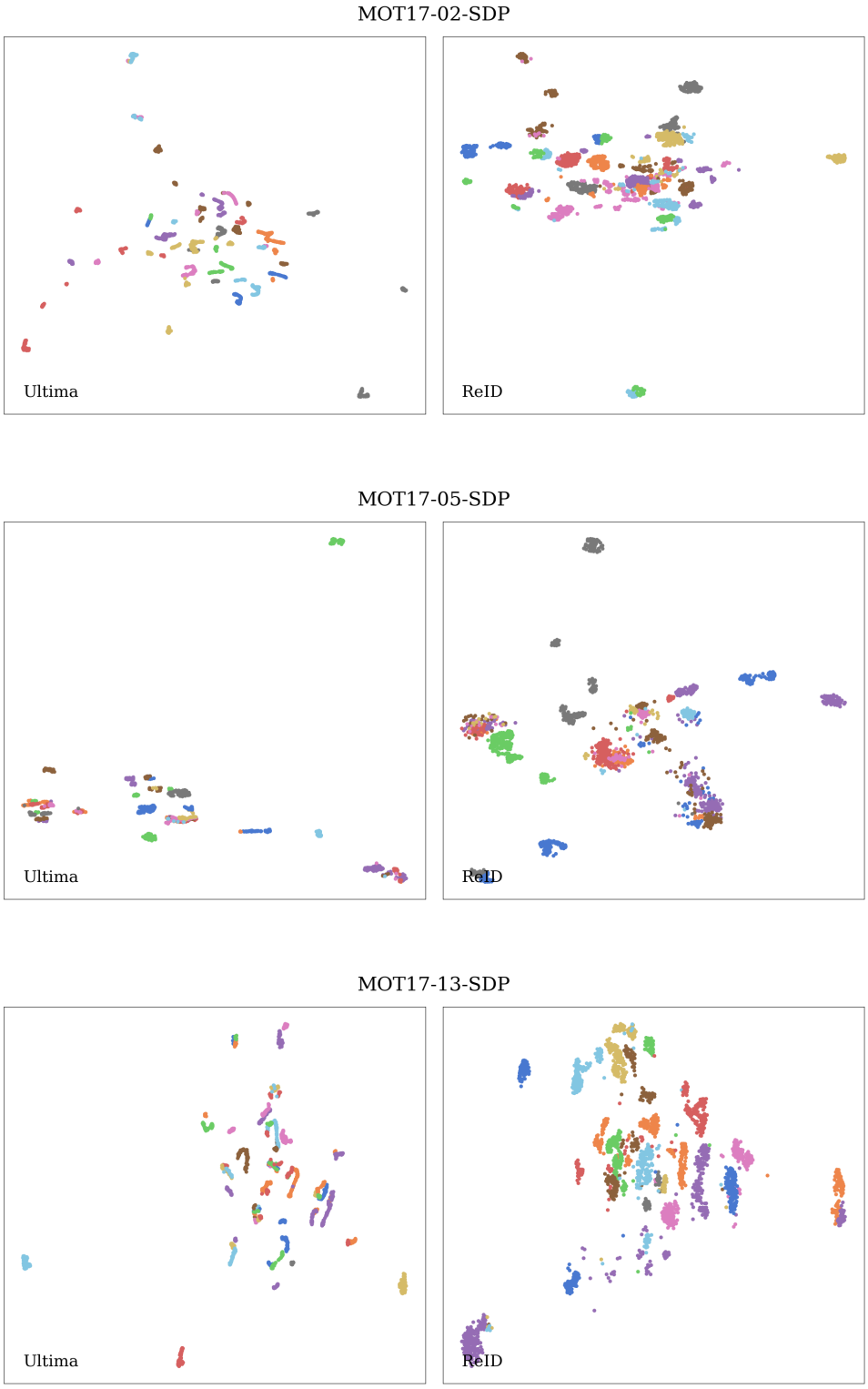


Figure 4.2: A visualization of MOT17 training set embeddings of our proposed method *Ultima* (right), trained on DanceTrack, compared to that of a pre-trained ReID model OSNet-AIN (left).

# Chapter 5

# Discussion WIP

In this chapter we will discuss the results presented in the prior chapter. We will then elaborate on the design choices made in our novel tracking method. Finally, we will touch on ethical and environmental impact considerations.

## 5.1 Analysis of Results

On the whole, we are encouraged by the results shown when benchmarking our method on the DanceTrack and MOT17 datasets. While our method doesn't outperform the more recent competition on the DanceTrack dataset, MOTRv2, CO-MOT and MeMOTR, which follow the tracking-by-attention paradigm, we believe we offer a novel approach for tracking-by-detection. Our cross-domain results are particularly compelling, showing that our method is quite robust in out-of-domain circumstances. On this point, it would be interesting to see other methods present similar results, so as to have another axis of comparison between methods.

In the comparison of learned features to that of ReID, our method performed well in the cross-domain context. The results seem to indicate that including positional information in embeddings may be useful in so much as the representations between objects become easier to discern. Of course, tracking methods that use ReID features, such as Deep SORT/OC-SORT, make use of motion models and heuristics instead. Our method outperforms such methods in the in-domain experiments, indicating that embedding positional information in the feature space, rather than treating it as a separate step in performing associations, may have performance benefits. However, our methods are too different to necessarily make that argument. For example, when encoding proposal queries our method is able to take into account

the context of other objects in the same scene, and thus may produce more discriminant visual features which make up the difference in performance. It would be interesting to perform further experiments wherein we compare purely motion based methods such as OC-SORT against our method sans the visual features.

**Stuff about MOT17 perf here.**

## 5.2 Design Choices

The novel method presented in this thesis performs quite well on both datasets tested, though trails behind other recently published methods. We posit that the performance delta seen in DanceTrack is primarily due to the older architecture that we employ. While the original MOTR used the same DETR-style architecture as we do, MOTRv2, and other methods that are more performant than ours on the DanceTrack dataset, use derivatives of the DETR architecture. To be more precise, MOTRv2, CO-MOT and MeMOTR employ deformable-DETR [44], which uses deformable attention in both the encoder and decoder transformer of the DETR architecture, and incorporate multi-scale features from the CNN backbone. The deformable attention guides learning by letting each image patch in the image transformer encoder attend to only a set number of patches close to itself. Similarly, detection queries are only allowed to attend to a set number of image patches close to their corresponding detection during cross-attention.

Due to deformable attention's increased efficiency, deformable-DETR is able to integrate features from multiple stages, i.e. multiple scales, of the CNN backbone. This is beneficial as it is well known in the field of deep learning-based computer vision that using visual features from multiple scales enhances the performance of many tasks, be that detection or classification [45]. The original DETR, however, uses only the features from the last stage, stage 4, of the CNN backbone, by which time the feature space has been down-sampled by a factor 32. Thus, we posit that our learned features may benefit and become more discriminant if we were to employ the deformable-DETR architecture.

During the initial stages of designing our method we weighed the possibility of using deformable-DETR, but ultimately decided to go with the much simpler standard DETR architecture. Another factor in our choice, other than simplicity, was that of memory. We opted to use standard DETR as existing implementations of deformable-DETR were not readily compatible with the half precision training required to extend the context length. Our chosen context length of 24 was just enough to fit into memory during training,

and this seemed like a reasonable choice as it would allow a bit more than one second of occlusion in DanceTrack, and a bit less than one second in MOT17. For comparison, initial testing using deformable-DETR at single precision only allowed for a context length of 8 in our VRAM budget. If we had access to more graphics cards, or cards with more VRAM than that of the 24GB RTX 4090, this would not have been an issue.

There are certainly other methods that could be employed to extend the context during inference, without having to resort to an increased context during training. We could, for example, sample frames with a variable stride between frames in a clip and pad the context as we usually do when objects lack detection. This would perhaps violate the usual behaviour of the detector, but may ultimately help during difficult portions of prolonged occlusions through allowing increased contexts. Another idea would be to use Rotary-Positional Embeddings [46], which have recently been shown to improve performance in GPT-style language models when compared to sine-positional embeddings. Importantly, the usage of rotations to encode positional information, together with a few tricks [47], have been shown to allow for longer contexts during inference than seen during training.

## 5.3 Ethical Concerns

Tracking is, as previously mentioned, of prime interest for surveillance. However, the topic of surveillance can be considered divisive; while some think surveillance is required for a safe society, others may argue that it presents privacy concerns. It is not difficult to see how improved tracking methods may eventually be deployed in such manners as to infringe on individuals' privacy en masse through surveillance, particularly when deployed by larger operators, such as surveillance states or large technology companies. As such, any improved tracking methods inherently present an ethical conundrum. While we will leave this discussion to philosophers, politicians and the general public, we believe that it is important to make note of this issue.

Another morally dubious area that may benefit from improved tracking methods are military applications. From the late twentieth century forward, we have seen increasingly automated militaries across many countries. Drones, for example, have become a frequently used tool, both in covert and active military operations. Examples of such are drone strikes by the US military on heads of terrorist groups, and makeshift drone bombers in the Ukraine war. While many of these cases still feature a human drone operator, it is

not difficult to see how some may want to automate drones as a military tool for increased efficiency and/or efficacy. Imagine, for example, having a fleet of automated drones continuously scan a region for a specified target, tracking and waiting for the person to be in the clear so as to "safely" take them out when no civilians are present. Even when disregarding the cases where this may go horribly wrong, the thought of such efficient methods of assassination presents a terrible future depending on whom you might ask.

## 5.4 Environmental Impact

The method presented in this thesis is by no means lightweight. While the memory requirements at inference is somewhat reasonable, around 1-2GB at full precision, the compute requirements are quite high. On an RTX 4090 running at full precision, which may be a bit slower than half-precision, the frame-rate is around 60 frames per second. Requiring such heavy hardware to perform tracking, however performant, is hardly environmentally friendly. This goes for both the environmental impact of producing the hardware to begin with, as well as the energy required to run the tracking algorithm. Baring any efforts to minimize the requirements necessary, applications requiring tracking may do better in using simpler methods that make use of lighter models, or no deep learning models at all.

In addition to environmental concerns at inference time, the impact of training should be taken into consideration. Training the DanceTrack model, for example, took around 5 days real-time on the 450W RTX 4090. If we go by approximate numbers, only considering the power of the AI-accelerator running at full power, that equates to 54kWh required. While this is not substantial, it may become if training is scaled up with more data and larger models.

# Chapter 6

# Conclusions WIP

In this thesis we set out to develop and present a novel method for multi-object tracking, using modern approaches from the field of representation learning. Our method, trained through a contrastive objective, learns to model an object's positional and visual appearance in a shared embedding space. The method builds on Joint-Embedding Predictive Architectures and features two main components, an encoder `Enc` and a predictor `Pred`. `Enc` takes detections from a pre-trained detector and encodes them while enriching with visual features to create what we call encoded proposal queries, which is a latent representation of each detected object. For each currently tracked object, `Pred` uses the object's past encoded proposal queries to predict its latent representation in the next time-step. Tracking is then performed by matching the predictions of `Pred` to the encoded proposals of `Enc` through cosine similarity.

We trained and evaluated the proposed method on the DanceTrack and MOT17 datasets, while comparing to other prior state-of-the-art methods using the HOTA, AssA, DetA, MOTA, and IDF1 metrics. On DanceTrack our method does not outperform the SOTA, but is quite competitive, suffering a 3.8 HOTA and 6.0 AssA difference when compared to the best performing model MOTRv2. However, it does out-compete the recent and more traditional tracking-by-detection method Deep OC-SORT by a HOTA and AssA difference of 4.8 and 6.2, respectively. **TODO: More to come about MOT17...**

To see how well our method generalizes in cross-domain settings, we further tested the trained models on the opposite dataset's test set. From this we found that our method seems quite robust to out of distribution situations. With 58.5 HOTA and 42.1 AssA on DanceTrack, Ultima performs just a bit

worse than Deep OC-SORT despite being trained on a magnitude less and out-of-domain data. We are only able to compare to results of methods trained and evaluated in-domain, however.

In addition to the qualitative results, we looked at low-dimensional projections of our learned representations as compared to an off-the-shelf ReID model. In doing so we found that our representations had more easily discernible clusterings compared to the ReID models, which we posit is due to the introduction of positional features.

Taken together, we can answer the research question that we posed in section 1.3. The tracking method presented in this thesis, trained through contemporary representation learning methods, seems to outperform more heuristic, i.e. "hand-crafted", methods.

## 6.1 Limitations

Due to the aforementioned lack of time/resources we were unable to perform any real ablation studies, which we see as a significant limitation of this thesis. Optimally, we would have liked to produce results that back up the design choices made in our method. In addition, we do not train multiple versions of the same model to perform statistical testing of performance variability. This does not seem to be common-practice in the field of MOT, at least from what we have seen during the literature study.

## 6.2 Future Work

As stated, it would be a good idea to perform ablations in the future so to systematically ascertain how design choices affect final performance. For example, the structure of the predictor, how the mixer layers affect results, and what values of $\tau$ should be used in the NT-Xent loss, would be good starting points. Of course, seeing how the scale of our model affects performance would also be interesting. Another point of great interest would be to investigate whether replacing the standard DETR with its deformable counterpart might yield improved metrics, as prior methods have shown. Furthermore, we would like to see this method trained and evaluated on other datasets such as TAO track, which features a much larger variety of different types of objects. This would give a good idea of how well the method performs at tracking when faced with long-tailed distributions.

## 6.3 Reflections

If I were to liken the process of working on this thesis to anything, it would be to a meandering river. While I started out with the impression that there was a project in place, before I would begin my internship, there turned out to be no plan for what I was supposed to work on at the National Institute of Informatics. For a few months I felt despair as I was aimlessly reading papers, trying to find something of importance and interest to work on. While I enjoyed having the opportunity to learn, and I did learn a lot, this was quite a stressful experience. Along the way it became clear that the topic of tracking would be of interest to the lab I was visiting, and so I tried to find a topic related to this that I could work on. I did eventually find a few projects. In fact, I tried out many things, to various degrees of success, but nothing quite captured my attention enough to feel worthwhile writing a thesis about. Perhaps this is a personal failing. I have a very difficult time bringing myself to task when the topic is of little interest to me; when it is of interest, I have a hard time stopping myself.

During my time reading papers, it became increasingly evident that representation learning was something I was truly fascinated with. With this in mind, I started looking at how representation learning had previously been applied in the field of MOT. Using visual features to re-identify objects was pretty much the go-to, but I found the formulation of ReID + movement models + heuristics in such an engineered approach to be unappealing. It felt like there should be some way to remove the need for hand-crafted methods and replace it with something entirely learned and in my mind more "pure". I feel like I've succeeded in this, at least somewhat. After all, the method presented in this thesis is straightforward, only requires two hyper-parameters and a very simple tracking algorithm, yet outperforms the more "hand-crafted" algorithms I was seeking to replace. Who knows, perhaps it may even outperform the tracking-by-attention competitors with a little more work.

Despite the pain of getting here, I don't think I would change the past as it happened even if I were able to. On the whole, I feel that the experience benefited me in ways that I'm not sure a more "structured" approach would have. For example, I think I am now much more familiar with the field of Machine Learning as a whole. I now know better how to engage with prior research, and I feel comfortable discussing my ideas with others. My only regret is that I didn't have more time and resources to do justice in exploring, systematically, the qualities of the method I proposed.

# References

[1] I. Smal, K. Draegestein, N. Galjart, W. Niessen, and E. Meijering, "Particle Filtering for Multiple Object Tracking in Dynamic Fluorescence Microscopy Images: Application to Microtubule Growth Analysis", *IEEE Transactions on Medical Imaging*, vol. 27, no. 6, pp. 789–804, Jun. 2008, ISSN: 1558-254X. DOI: 10.1109/TMI.2008.916964. [Online]. Available: https://ieeexplore.ieee.org/abstract/document/4436039?casa_token=bDp6kjve-N0AAAAA:jR1MNyxJdzWxdDj_XZMfZZI6R3tbGziP4Mbtfwg7Un6Z30tYq1vxA5IC6NQdrsAuBYxUsErGUw (visited on 09/24/2023).

[2] D. M. Jiménez-Bravo, Á. Lozano Murciego, A. Sales Mendes, H. Sánchez San Blás, and J. Bajo, "Multi-object tracking in traffic environments: A systematic literature review", *Neurocomputing*, vol. 494, pp. 43–55, Jul. 2022, ISSN: 0925-2312. DOI: 10.1016/j.neucom.2022.04.087. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0925231222004672 (visited on 09/28/2023).

[3] R. Girshick, J. Donahue, T. Darrell, and J. Malik, "Rich Feature Hierarchies for Accurate Object Detection and Semantic Segmentation", in *Proceedings of the 2014 IEEE Conference on Computer Vision and Pattern Recognition*, ser. CVPR '14, USA: IEEE Computer Society, Jun. 2014, pp. 580–587, ISBN: 9781479951185. DOI: 10.1109/CVPR.2014.81. [Online]. Available: https://doi.org/10.1109/CVPR.2014.81 (visited on 09/09/2023).

[4] J. Redmon, S. Divvala, R. Girshick, and A. Farhadi, "You Only Look Once: Unified, Real-Time Object Detection", in *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, ISSN: 1063-6919, Jun. 2016, pp. 779–788. DOI: 10.1109/CVPR.2016.91.

[5] Y. LeCun, "A Path Towards Autonomous Machine Intelligence", en,

[6] M. Oquab *et al.*, *DINOv2: Learning Robust Visual Features without Supervision*, arXiv:2304.07193 [cs], Apr. 2023. DOI: 10.48550/arXiv.2304.07193. [Online]. Available: http://arxiv.org/abs/2304.07193 (visited on 06/13/2023).

[7] T. Brown *et al.*, "Language Models are Few-Shot Learners", in *Advances in Neural Information Processing Systems*, vol. 33, Curran Associates, Inc., 2020, pp. 1877–1901. [Online]. Available: https://papers.nips.cc/paper/2020/hash/1457c0d6bfcb4967418bfb8ac142f64a-Abstract.html (visited on 06/13/2023).

[8] M. Caron *et al.*, *Emerging Properties in Self-Supervised Vision Transformers*, arXiv:2104.14294 [cs], May 2021. DOI: 10.48550/arXiv.2104.14294. [Online]. Available: http://arxiv.org/abs/2104.14294 (visited on 06/13/2023).

[9] A. Radford *et al.*, *Learning Transferable Visual Models From Natural Language Supervision*, arXiv:2103.00020 [cs], Feb. 2021. DOI: 10.48550/arXiv.2103.00020. [Online]. Available: http://arxiv.org/abs/2103.00020 (visited on 06/13/2023).

[10] R. Girdhar *et al.*, *ImageBind: One Embedding Space To Bind Them All*, arXiv:2305.05665 [cs], May 2023. DOI: 10.48550/arXiv.2305.05665. [Online]. Available: http://arxiv.org/abs/2305.05665 (visited on 06/13/2023).

[11] M. Assran *et al.*, *Self-Supervised Learning from Images with a Joint-Embedding Predictive Architecture*, arXiv:2301.08243 [cs, eess], Apr. 2023. DOI: 10.48550/arXiv.2301.08243. [Online]. Available: http://arxiv.org/abs/2301.08243 (visited on 06/15/2023).

[12] K. He, X. Chen, S. Xie, Y. Li, P. Dollár, and R. Girshick, *Masked Autoencoders Are Scalable Vision Learners*, arXiv:2111.06377 [cs], Dec. 2021. DOI: 10.48550/arXiv.2111.06377. [Online]. Available: http://arxiv.org/abs/2111.06377 (visited on 06/15/2023).

[13] A. Bardes, J. Ponce, and Y. LeCun, *VICReg: Variance-Invariance-Covariance Regularization for Self-Supervised Learning*, arXiv:2105.04906 [cs], Jan. 2022. DOI: 10.48550/arXiv.2105.04906. [Online]. Available: http://arxiv.org/abs/2105.04906 (visited on 06/13/2023).

[14] T. Chen, S. Kornblith, M. Norouzi, and G. Hinton, *A Simple Framework for Contrastive Learning of Visual Representations*, arXiv:2002.05709 [cs, stat], Jun. 2020. DOI: 10.48550/arXiv.2002.05709. [Online]. Available: http://arxiv.org/abs/2002.05709 (visited on 05/20/2023).

[15] A. v. d. Oord, Y. Li, and O. Vinyals, *Representation Learning with Contrastive Predictive Coding*, arXiv:1807.03748 [cs, stat], Jan. 2019. DOI: 10.48550/arXiv.1807.03748. [Online]. Available: http://arxiv.org/abs/1807.03748 (visited on 06/15/2023).

[16] A. Bewley, Z. Ge, L. Ott, F. Ramos, and B. Upcroft, "Simple Online and Realtime Tracking", in *2016 IEEE International Conference on Image Processing (ICIP)*, arXiv:1602.00763 [cs], Sep. 2016, pp. 3464–3468. DOI: 10.1109/ICIP.2016.7533003. [Online]. Available: http://arxiv.org/abs/1602.00763 (visited on 02/08/2023).

[17] N. Wojke, A. Bewley, and D. Paulus, *Simple Online and Realtime Tracking with a Deep Association Metric*, arXiv:1703.07402 [cs], Mar. 2017. DOI: 10.48550/arXiv.1703.07402. [Online]. Available: http://arxiv.org/abs/1703.07402 (visited on 06/14/2023).

[18] J. Pang *et al.*, "Quasi-Dense Similarity Learning for Multiple Object Tracking", en, 2021, pp. 164–173. [Online]. Available: https://openaccess.thecvf.com/content/CVPR2021/html/Pang_Quasi-Dense_Similarity_Learning_for_Multiple_Object_Tracking_CVPR_2021_paper.html (visited on 06/18/2023).

[19] Y.-H. Wang, J.-W. Hsieh, P.-Y. Chen, M.-C. Chang, H. H. So, and X. Li, *SMILEtrack: SiMIlarity LEarning for Occlusion-Aware Multiple Object Tracking*, arXiv:2211.08824 [cs] version: 3, Aug. 2023. DOI: 10.48550/arXiv.2211.08824. [Online]. Available: http://arxiv.org/abs/2211.08824 (visited on 09/02/2023).

[20] J. Cao, X. Weng, R. Khirodkar, J. Pang, and K. Kitani, *Observation-Centric SORT: Rethinking SORT for Robust Multi-Object Tracking*, arXiv:2203.14360 [cs], Mar. 2022. [Online]. Available: http://arxiv.org/abs/2203.14360 (visited on 02/08/2023).

[21] P. Sun *et al.*, *DanceTrack: Multi-Object Tracking in Uniform Appearance and Diverse Motion*, arXiv:2111.14690 [cs], May 2022. DOI: 10.48550/arXiv.2111.14690. [Online]. Available: http://arxiv.org/abs/2111.14690 (visited on 06/18/2023).

[22] N. Carion, F. Massa, G. Synnaeve, N. Usunier, A. Kirillov, and S. Zagoruyko, *End-to-End Object Detection with Transformers*, arXiv:2005.12872 [cs], May 2020. DOI: 10.48550/arXiv.2005.12872. [Online]. Available: http://arxiv.org/abs/2005.12872 (visited on 09/06/2023).

[23] P. Sun *et al.*, *TransTrack: Multiple Object Tracking with Transformer*, arXiv:2012.15460 [cs], May 2021. [Online]. Available: http://arxiv.org/abs/2012.15460 (visited on 05/20/2023).

[24] T. Meinhardt, A. Kirillov, L. Leal-Taixé, and C. Feichtenhofer, "TrackFormer: Multi-Object Tracking With Transformers", en, 2022, pp. 8844–8854. [Online]. Available: https://openaccess.thecvf.com/content/CVPR2022/html/Meinhardt_TrackFormer_Multi-Object_Tracking_With_Transformers_CVPR_2022_paper.html (visited on 09/09/2023).

[25] F. Zeng, B. Dong, Y. Zhang, T. Wang, X. Zhang, and Y. Wei, *MOTR: End-to-End Multiple-Object Tracking with Transformer*, arXiv:2105.03247 [cs] version: 4, Jul. 2022. DOI: 10.48550/arXiv.2105.03247. [Online]. Available: http://arxiv.org/abs/2105.03247 (visited on 06/18/2023).

[26] Y. Zhang, T. Wang, and X. Zhang, "MOTRv2: Bootstrapping End-to-End Multi-Object Tracking by Pretrained Object Detectors", en, 2023, pp. 22 056–22 065. [Online]. Available: https://openaccess.thecvf.com/content/CVPR2023/html/Zhang_MOTRv2_Bootstrapping_End-to-End_Multi-Object_Tracking_by_Pretrained_Object_Detectors_CVPR_2023_paper.html (visited on 06/06/2023).

[27] Z. Ge, S. Liu, F. Wang, Z. Li, and J. Sun, *YOLOX: Exceeding YOLO Series in 2021*, arXiv:2107.08430 [cs], Aug. 2021. DOI: 10.48550/arXiv.2107.08430. [Online]. Available: http://arxiv.org/abs/2107.08430 (visited on 09/06/2023).

[28] P. Dendorfer *et al.*, "MOTChallenge: A Benchmark for Single-Camera Multiple Target Tracking", en, *International Journal of Computer Vision*, vol. 129, no. 4, pp. 845–881, Apr. 2021, ISSN: 1573-1405. DOI: 10.1007/s11263-020-01393-0. [Online]. Available: https://doi.org/10.1007/s11263-020-01393-0 (visited on 09/05/2023).

[29] K. Bernardin and R. Stiefelhagen, "Evaluating Multiple Object Tracking Performance: The CLEAR MOT Metrics", en, *EURASIP Journal on Image and Video Processing*, vol. 2008, no. 1, pp. 1–10, Dec. 2008, ISSN: 1687-5281. DOI: 10.1155/2008/246309. [Online]. Available: https://jivp-eurasipjournals.springeropen.com/articles/10.1155/2008/246309 (visited on 09/09/2023).

[30] J. Luiten *et al.*, "HOTA: A Higher Order Metric for Evaluating Multi-Object Tracking", *International Journal of Computer Vision*, vol. 129, no. 2, pp. 548–578, Feb. 2021, arXiv:2009.07736 [cs], ISSN: 0920-5691, 1573-1405. DOI: 10.1007/s11263-020-01375-2. [Online]. Available: http://arxiv.org/abs/2009.07736 (visited on 08/26/2023).

[31] Y. Li, S. Si, G. Li, C.-J. Hsieh, and S. Bengio, "Learnable Fourier Features for Multi-dimensional Spatial Positional Encoding", in *Advances in Neural Information Processing Systems*, vol. 34, Curran Associates, Inc., 2021, pp. 15 816–15 829. [Online]. Available: https://proceedings.neurips.cc/paper/2021/hash/84c2d4860a0fc27bcf854c444fb8b400-Abstract.html (visited on 09/06/2023).

[32] A. Vaswani *et al.*, "Attention is All you Need", in *Advances in Neural Information Processing Systems*, vol. 30, Curran Associates, Inc., 2017. [Online]. Available: https://proceedings.neurips.cc/paper_files/paper/2017/hash/3f5ee243547dee91fbd053c1c4a845aa-Abstract.html (visited on 09/21/2023).

[33] Y. Zhang *et al.*, "ByteTrack: Multi-object Tracking by Associating Every Detection Box", en, in *Computer Vision – ECCV 2022*, S. Avidan, G. Brostow, M. Cissé, G. M. Farinella, and T. Hassner, Eds., ser. Lecture Notes in Computer Science, Cham: Springer Nature Switzerland, 2022, pp. 1–21, ISBN: 9783031200472. DOI: 10.1007/978-3-031-20047-2_1.

[34] T.-Y. Lin, P. Goyal, R. Girshick, K. He, and P. Dollár, *Focal Loss for Dense Object Detection*, arXiv:1708.02002 [cs], Feb. 2018. DOI: 10.48550/arXiv.1708.02002. [Online]. Available: http://arxiv.org/abs/1708.02002 (visited on 09/07/2023).

[35] A. H. Jonathon Luiten, *TrackEval*, 2020. [Online]. Available: https://github.com/JonathonLuiten/TrackEval.

[36] K. Zhou, Y. Yang, A. Cavallaro, and T. Xiang, "Omni-Scale Feature Learning for Person Re-Identification", 2019, pp. 3702–3712. [Online]. Available: https://openaccess.thecvf.com/content_ICCV_2019/html/Zhou_Omni-Scale_Feature_Learning_for_Person_Re-Identification_ICCV_2019_paper.html (visited on 09/21/2023).

[37] K. Zhou, Y. Yang, A. Cavallaro, and T. Xiang, "Learning Generalisable Omni-Scale Representations for Person Re-Identification", *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 44, no. 9, pp. 5056–5069, Sep. 2022, ISSN: 1939-3539. DOI: 10.1109/TPAMI.2021.3069237.

[38] K. Zhou and T. Xiang, *Torchreid: A Library for Deep Learning Person Re-Identification in Pytorch*, arXiv:1910.10093 [cs], Oct. 2019. DOI: 10.48550/arXiv.1910.10093. [Online]. Available: http://arxiv.org/abs/1910.10093 (visited on 09/21/2023).

[39] L. v. d. Maaten and G. Hinton, "Visualizing Data using t-SNE", *Journal of Machine Learning Research*, vol. 9, no. 86, pp. 2579–2605, 2008, ISSN: 1533-7928. [Online]. Available: http://jmlr.org/papers/v9/vandermaaten08a.html (visited on 09/11/2023).

[40] D. M. Chan, R. Rao, F. Huang, and J. F. Canny, "GPU accelerated t-distributed stochastic neighbor embedding", *Journal of Parallel and Distributed Computing*, vol. 131, pp. 1–13, Sep. 2019, ISSN: 0743-7315. DOI: 10.1016/j.jpdc.2019.04.008. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S074373151830875X (visited on 09/11/2023).

[41] G. Maggiolino, A. Ahmad, J. Cao, and K. Kitani, *Deep OC-SORT: Multi-Pedestrian Tracking by Adaptive Re-Identification*, arXiv:2302.11813 [cs] version: 1, Feb. 2023. DOI: 10.48550/arXiv.2302.11813. [Online]. Available: http://arxiv.org/abs/2302.11813 (visited on 09/23/2023).

[42] R. Gao and L. Wang, *MeMOTR: Long-Term Memory-Augmented Transformer for Multi-Object Tracking*, arXiv:2307.15700 [cs] version: 2, Jul. 2023. DOI: 10.48550/arXiv.2307.15700. [Online]. Available: http://arxiv.org/abs/2307.15700 (visited on 09/23/2023).

[43] F. Yan, W. Luo, Y. Zhong, Y. Gan, and L. Ma, *Bridging the Gap Between End-to-end and Non-End-to-end Multi-Object Tracking*, arXiv:2305.12724 [cs] version: 1, May 2023. DOI: 10.48550/arXiv.2305.12724. [Online]. Available: http://arxiv.org/abs/2305.12724 (visited on 09/23/2023).

[44] X. Zhu, W. Su, L. Lu, B. Li, X. Wang, and J. Dai, *Deformable DETR: Deformable Transformers for End-to-End Object Detection*, arXiv:2010.04159 [cs], Mar. 2021. DOI: 10.48550/arXiv.2010.04159. [Online]. Available: http://arxiv.org/abs/2010.04159 (visited on 09/25/2023).

[45] C. Szegedy *et al.*, "Going Deeper With Convolutions", 2015, pp. 1–9. [Online]. Available: https://www.cv-foundation.org/openaccess/content_cvpr_2015/html/Szegedy_Going_Deeper_With_2015_CVPR_paper.html (visited on 09/25/2023).

[46] J. Su, Y. Lu, S. Pan, A. Murtadha, B. Wen, and Y. Liu, *RoFormer: Enhanced Transformer with Rotary Position Embedding*, arXiv:2104.09864 [cs] version: 4, Aug. 2022. DOI: 10.48550/arXiv.2104.09864. [Online]. Available: http://arxiv.org/abs/2104.09864 (visited on 09/25/2023).

[47] Y. Sun *et al.*, *A Length-Extrapolatable Transformer*, arXiv:2212.10554 [cs] version: 1, Dec. 2022. DOI: 10.48550/arXiv.2212.10554. [Online]. Available: http://arxiv.org/abs/2212.10554 (visited on 09/25/2023).

# Appendix A

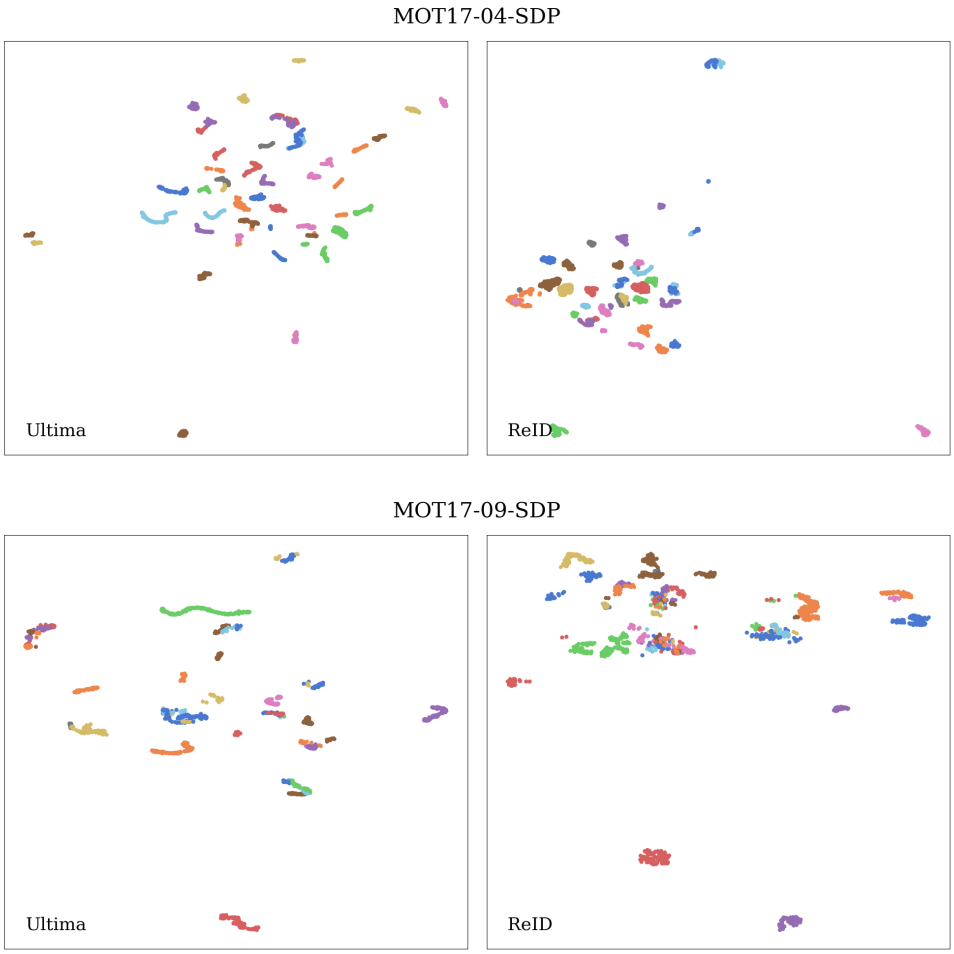


Figure 6.1: A t-sne visualization of MOT-17 training set embeddings of our proposed method *Ultima*, trained on DanceTrack, compared to that of a pre-trained ReID model OSNet-AIN (left). Each object has an associated color.

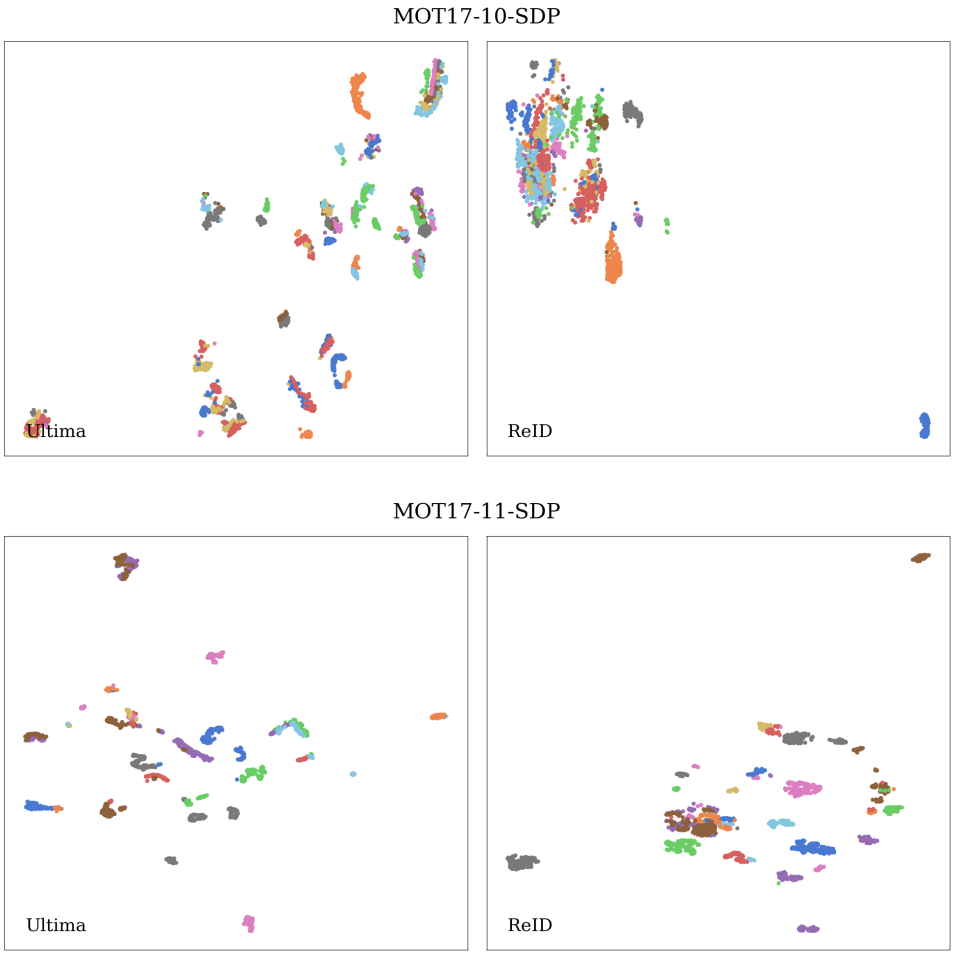


Figure 6.2: A t-sne visualization of MOT-17 training set embeddings of our proposed method *Ultima*, trained on DanceTrack, compared to that of a pre-trained ReID model OSNet-AIN (left). Each object has an associated color.

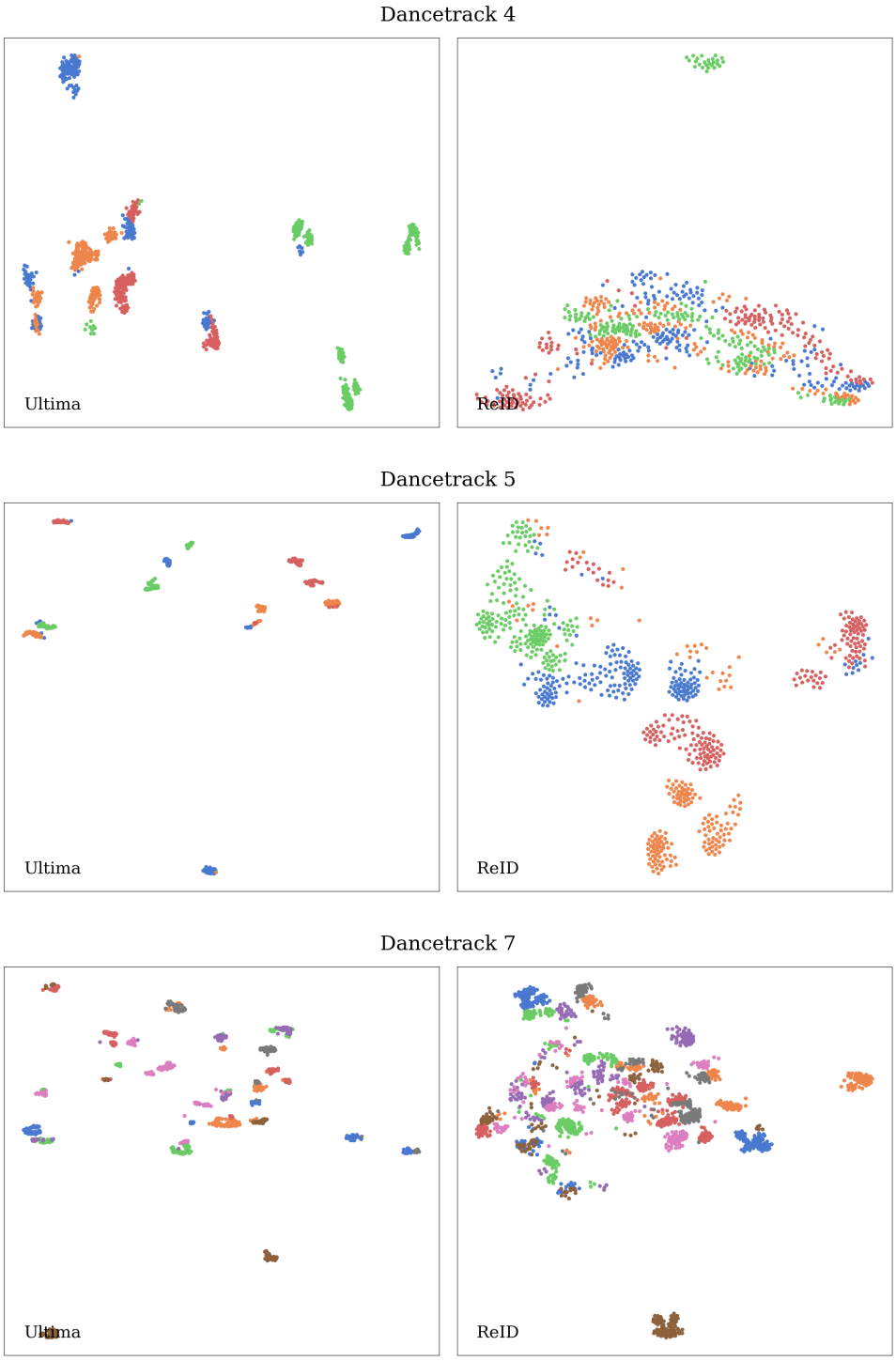


Figure 6.3: A t-sne visualization of DanceTrack validation set embeddings of our proposed method *Ultima*, trained on MOT17, compared to that of a pre-trained ReID model OSNet-AIN (left). Each object has an associated color.

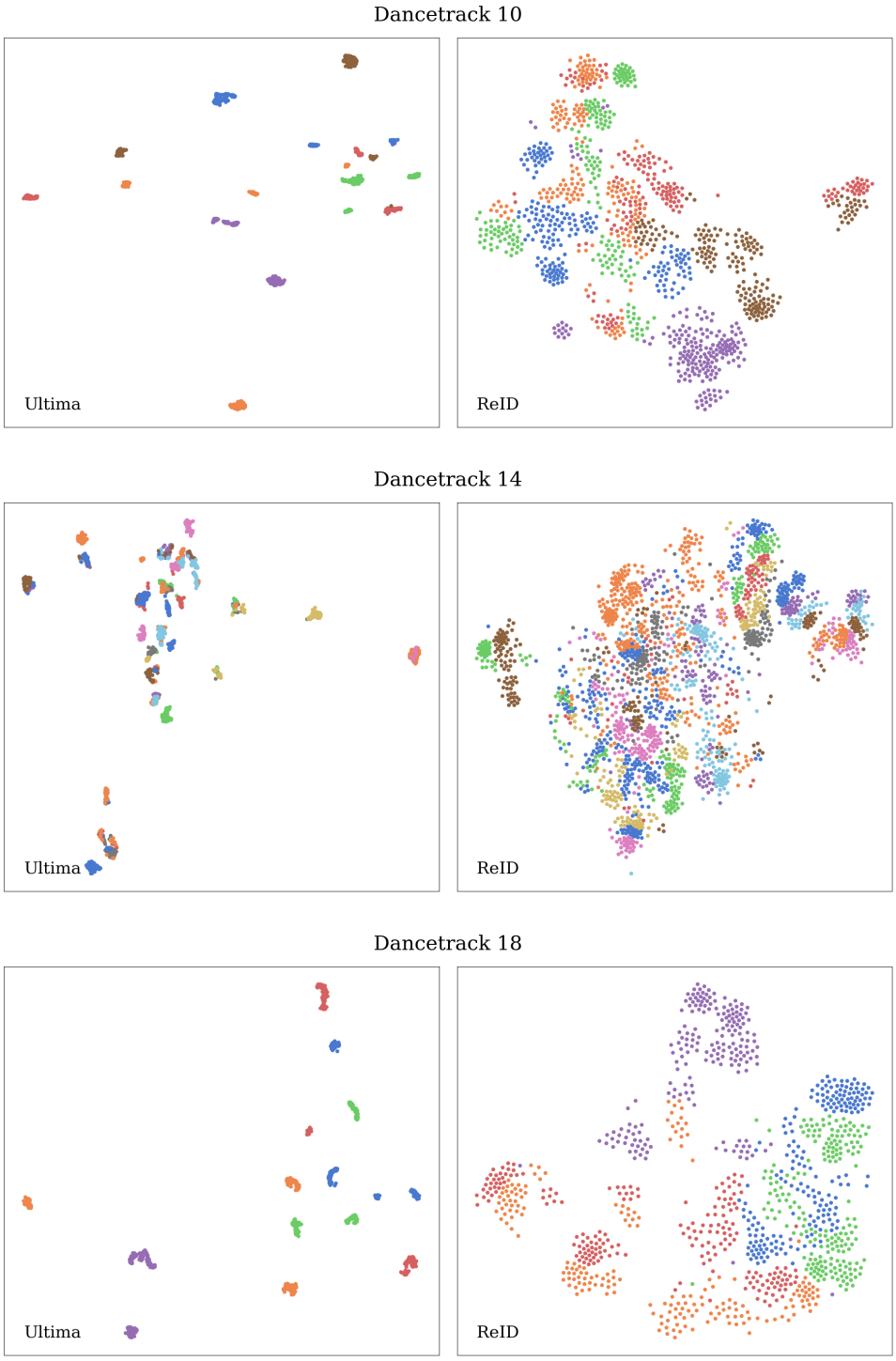


Figure 6.4: A t-sne visualization of DanceTrack validation set embeddings of our proposed method *Ultima*, trained on MOT17, compared to that of a pre-trained ReID model OSNet-AIN (left). Each object has an associated color.

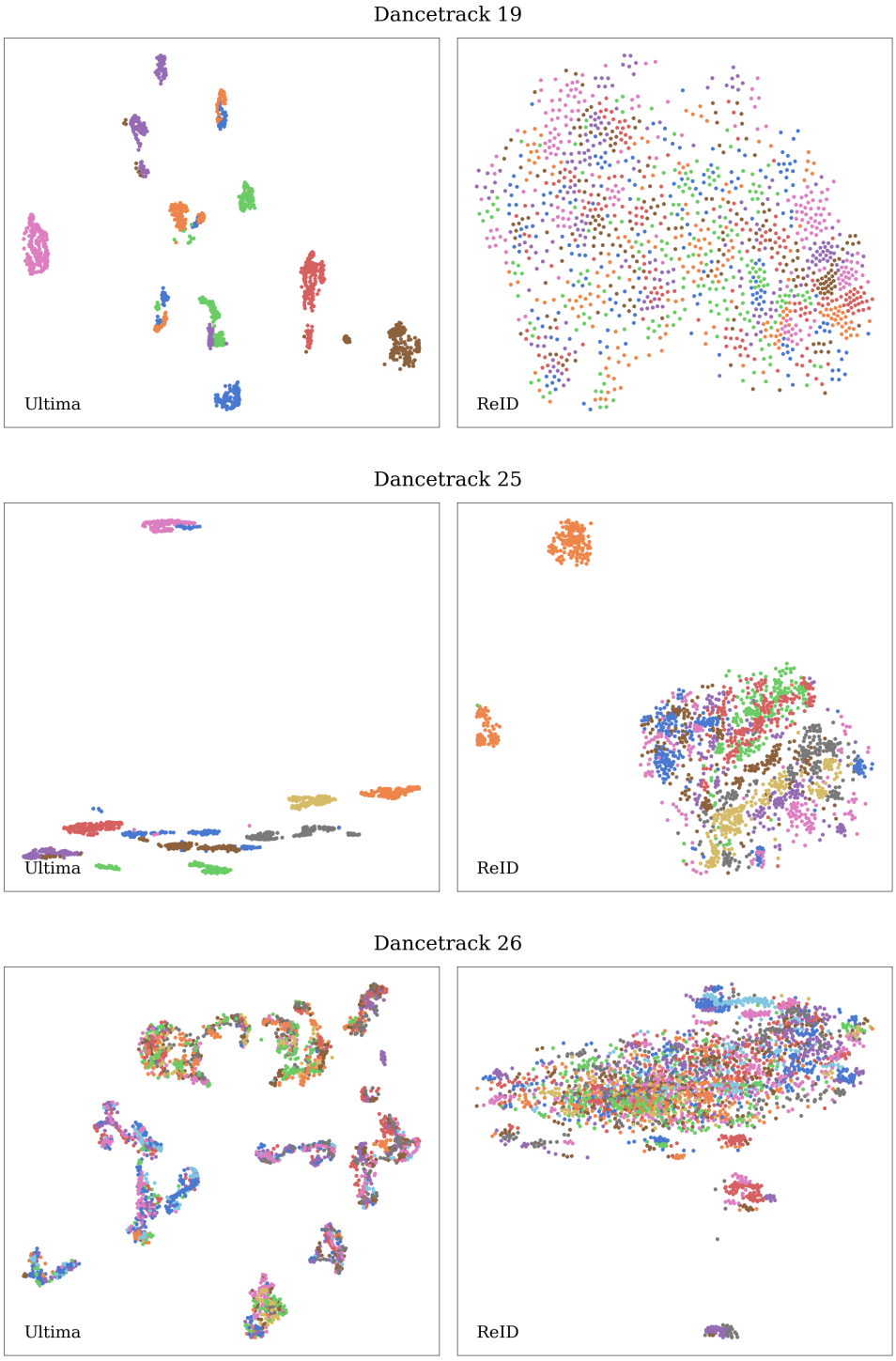


Figure 6.5: A t-sne visualization of DanceTrack validation set embeddings of our proposed method *Ultima*, trained on MOT17, compared to that of a pre-trained ReID model OSNet-AIN (left). Each object has an associated color.

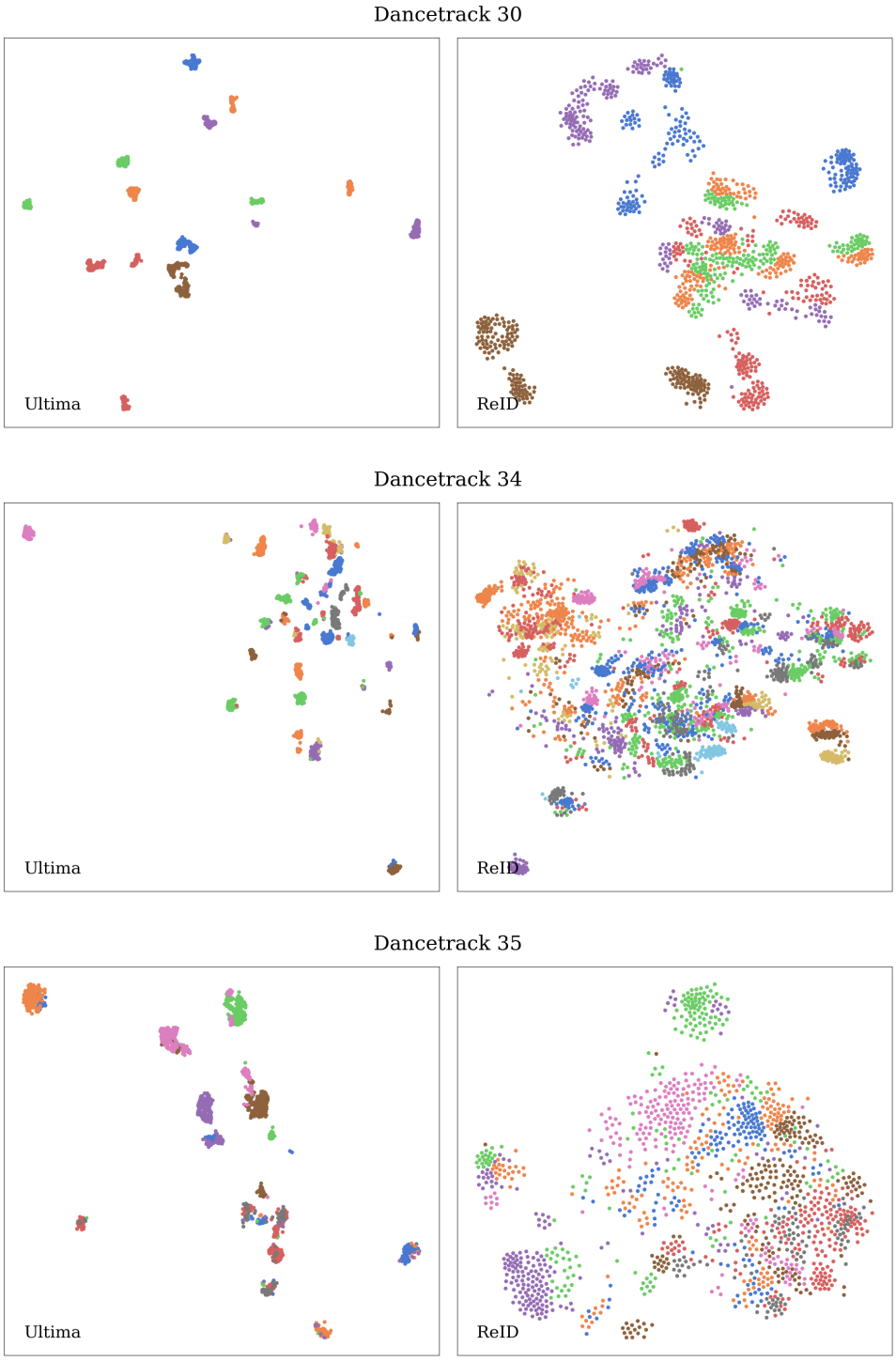


Figure 6.6: A t-sne visualization of DanceTrack validation set embeddings of our proposed method *Ultima*, trained on MOT17, compared to that of a pre-trained ReID model OSNet-AIN (left). Each object has an associated color.

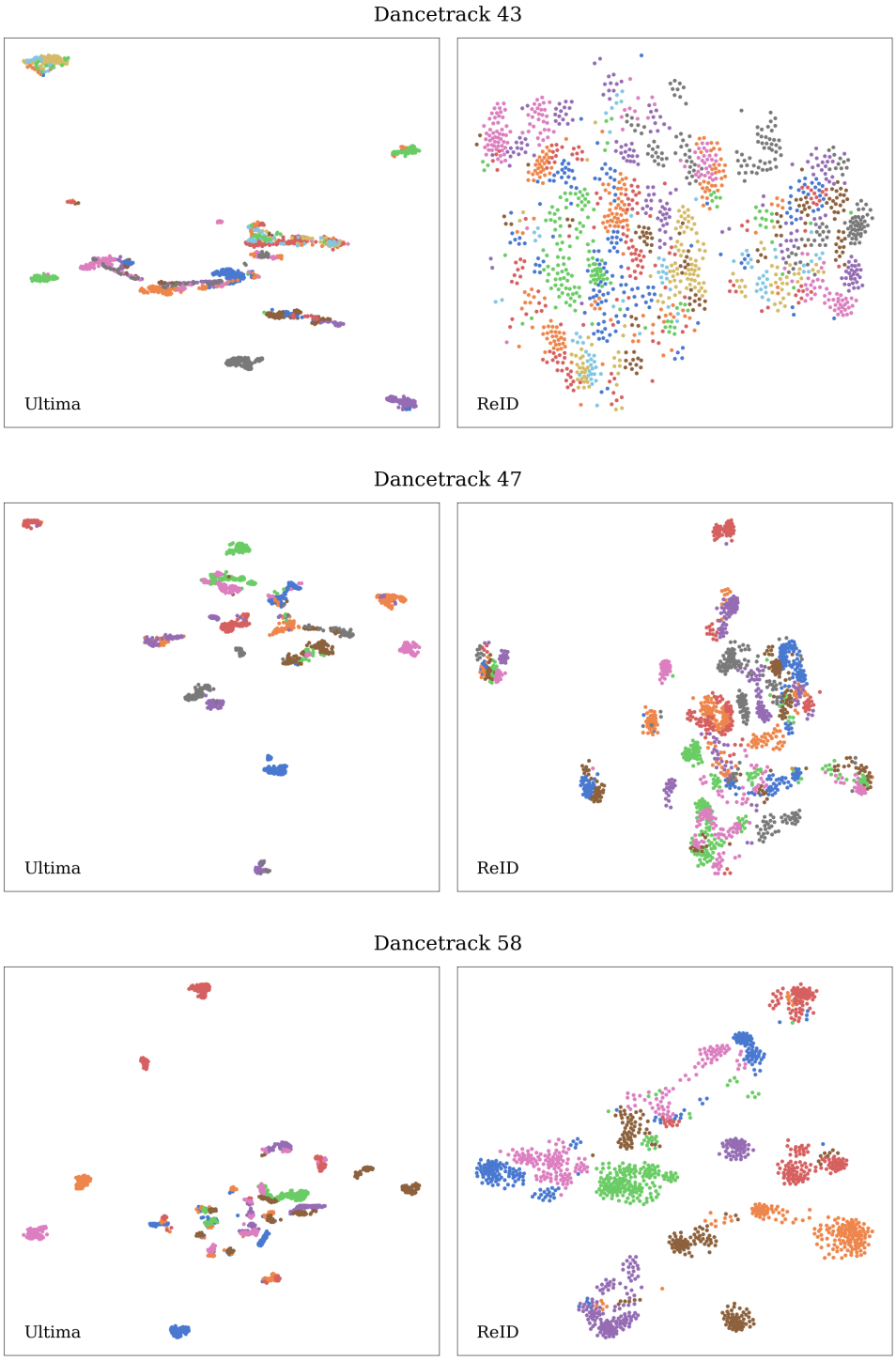


Figure 6.7: A t-sne visualization of DanceTrack validation set embeddings of our proposed method *Ultima*, trained on MOT17, compared to that of a pre-trained ReID model OSNet-AIN (left). Each object has an associated color.

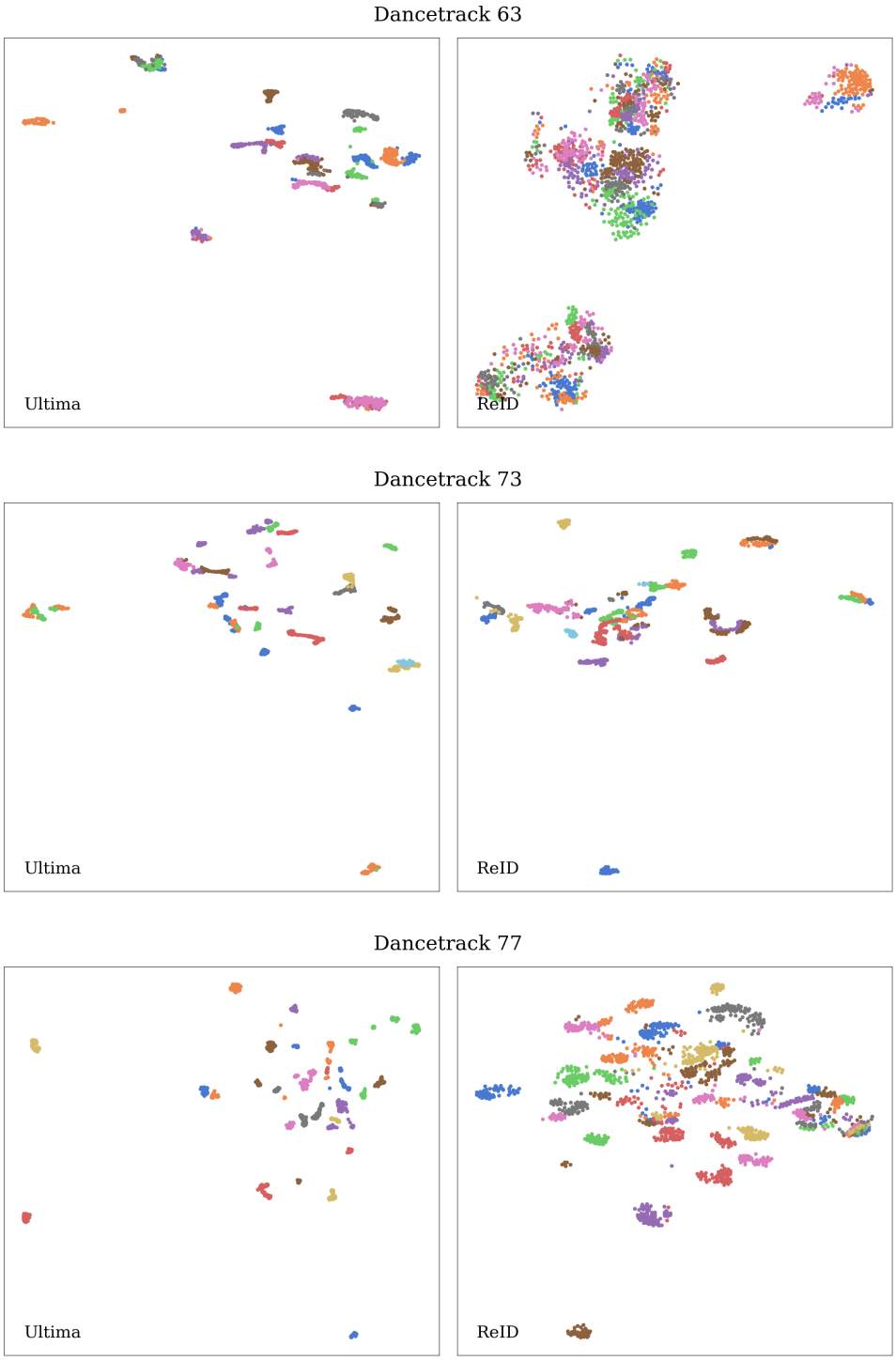


Figure 6.8: A t-sne visualization of DanceTrack validation set embeddings of our proposed method *Ultima*, trained on MOT17, compared to that of a pre-trained ReID model OSNet-AIN (left). Each object has an associated color.


Figure 6.9: A t-sne visualization of DanceTrack validation set embeddings of our proposed method *Ultima*, trained on MOT17, compared to that of a pre-trained ReID model OSNet-AIN (left). Each object has an associated color.

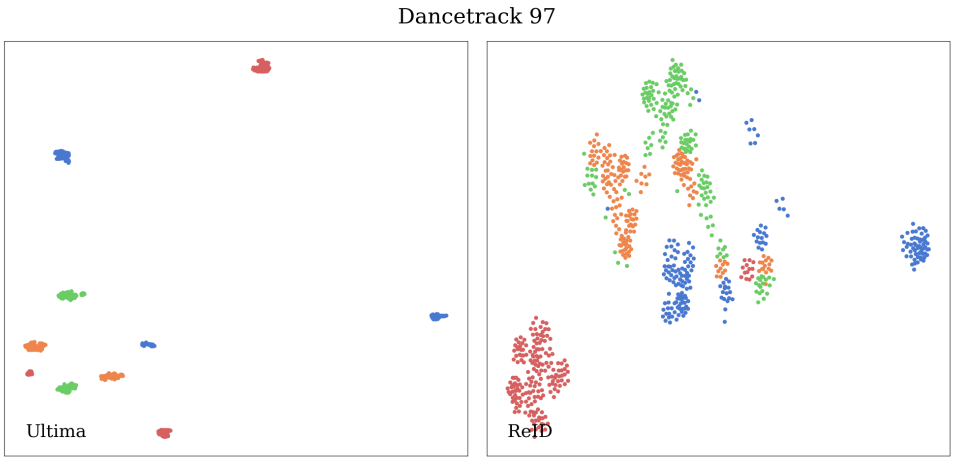


Figure 6.10: A t-sne visualization of DanceTrack validation set embeddings of our proposed method *Ultima*, trained on MOT17, compared to that of a pre-trained ReID model OSNet-AIN (left). Each object has an associated color.

TRITA-EECS-EX- 2023:0000

# €€€€ For DIVA €€€€


{
"Author1": { "Last name": "Maus",
"First name": "Rickard",
"Local User Id": "udunno",
"E-mail": "maus@kth.se",
"organisation": {"L1": "School of Electrical Engineering and Computer Science",
}
},
"Cycle": "2",
"Course code": "DA233X",
"Credits": "30.0",
"Degree1": {"Educational program": "Degree Programme in Computer Science and Engineering"
,"programcode": "CDATE"
,"Degree": "Degree of Master of Science in Engineering"
,"subjectArea": "Computer Science and Engineering"
},
"Title": {
"Main title": "Tracking with Joint-Embedding Predictive Architectures",
"Subtitle": "Learning to track through representation learning",
"Language": "eng" },
"Alternative title": {
"Main title": "Spårning genom Prediktiva Arkitekturer med Gemensam Inbäddning",
"Subtitle": "Att lära sig att spåra genom representations inlärning",
"Language": "swe"
},
"Supervisor1": { "Last name": "Jensfelt",
"First name": "Patric",
"Local User Id": "u100003",
"E-mail": "patric@kth.se",
"organisation": {"L1": "School of Electrical Engineering and Computer Science",
"L2": "Computer Science" }
},
"Supervisor2": { "Last name": "Prendinger",
"First name": "Helmut",
"E-mail": "helmut@nii.ac.jp",
"Other organisation": "NII National Institute of Technology"
},
"Examiner1": { "Last name": "Azizpour",
"First name": "Hossain",
"Local User Id": "u1d13i2c",
"E-mail": "azizpour@kth.se",
"organisation": {"L1": "School of Electrical Engineering and Computer Science",
"L2": "RPL" }
},
"National Subject Categories": "10201, 10206, 10207",
"Other information": {"Year": "2023", "Number of pages": "1,43"},
"Copyrightleft": "copyright",
"Series": { "Title of series": "TRITA-EECS-EX" , "No. in series": "2023:0000" },
"Opponents": { "Name": "A. B. Normal & A. X. E. Normalè"},
"Presentation": { "Date": "2022-03-15 13:00"
,"Language":"eng"
,"Room": "via Zoom https://kth-se.zoom.us/j/ddddddddddd"
,"Address": "Isafjordsgatan 22 (Kistagången 16)"
,"City": "Stockholm" },
"Number of lang instances": "2",
"Abstract[eng ]": €€€€
Leaving this for later.  €€€€,
"Keywords[eng ]": €€€€
Contrastive learning, Joint-Embedding predictive architectures, Multi-object tracking, Representation learning  €€€€,
"Abstract[swe ]": €€€€
€€€€,
"Keywords[swe ]": €€€€
Kontrastiv inlärning, Prediktiva arkitekturer med gemensam inbäddning, Spårning av flera objekt, Representations inlärning  €€€€,
}