5-2023

# Dynamic scene understanding: Pedestrian tracking from aerial devices.

Abdelhamid Bouzid
*University of Louisville*

# DYNAMIC SCENE UNDERSTANDING:
# PEDESTRIAN TRACKING FROM AERIAL DEVICES

By

Abdelhamid Bouzid
M.S. Computer Science, University of Louisville, 2023

A Dissertation
Submitted to the Faculty of the
J.B. Speed School of Engineering
in Partial Fulfillment of the Requirements
for the Degree of

Doctor of Philosophy in Computer Science and Engineering

Department of Computer Science and Engineering
University of Louisville
Louisville, Kentucky

May 2023

# DYNAMIC SCENE UNDERSTANDING:
# PEDESTRIAN TRACKING FROM AERIAL DEVICES

By

Abdelhamid Bouzid
M.S. Computer Science, University of Louisville, 2023

A Dissertation Approved On

4/24/2023

By the following Dissertation Committee:

Adel S. Elmaghraby, Dissertation Director.

Ibrahim N. Imam.

Hui Zhang.

Michael Losavio.

Daniel Sierra-Sosa.

# ACKNOWLEDGEMENTS

First of all, I would like to thank The University of Louisville for offering me this opportunity to have this memorable experience and to work in these encouraging conditions.

I would like also to express my sincere gratitude to my advisor Prof. Adel Elmaghraby for the continuous support of this research, for his patience, motivation, and immense knowledge. His guidance helped me throughout this research and in the writing of this report.

I would also like to thank Industrial Research and Innovation laboratory members for their support during my Ph.D. Laboratory members were a great support for my Ph.D thesis, offering me help and guiding me with valuable advices.

Last but not least, I would never forget the encouragement and the unconditional support I have always had from my parents, my siblings and my friends.

# ABSTRACT

DYNAMIC SCENE UNDERSTANDING:
PEDESTRIAN TRACKING FROM AERIAL DEVICES

Abdelhamid Bouzid

May 1, 2023

Multiple Object Tracking (MOT) is the problem that involves following the trajectory of multiple objects in a sequence, generally a video. Pedestrians are among the most interesting subjects to track and recognize for many purposes such as surveillance, and safety. In the recent years, Unmanned Aerial Vehicles (UAV's) have been viewed as a viable option for monitoring public areas, as they provide a low-cost method of data collection while covering large and difficult-to-reach areas. In this thesis, we present an online pedestrian tracking and re-identification from aerial devices framework. This framework is based on learning a compact directional statistic distribution (von-Mises-Fisher distribution) for each person ID using a deep convolutional neural network. The distribution characteristics are trained to be invariant to clothes appearances and to transformations. In real world scenarios, during deployment, new pedestrian and objects can appear in the scene and the model should detect them as Out Of Distribution (OOD). Thus, our frameworks also includes an OOD detection adopted from [16] called Virtual Outlier Synthetic (VOS), that detects OOD based on synthesising virtual outlier in the embedding space in an online manner. To validate, analyze and compare our approach, we use a large real benchmark data that contain detection tracking and identity annotations. These targets are captured at different viewing angles, different places, and different times by a "DJI Phantom 4" drone. We validate the effectiveness of the proposed framework by evaluating their detection, tracking and long term identification performance as well as classification performance between In Distribution (ID) and OOD. We show that the the proposed methods in the framework can learn models that achieve their objectives.

# TABLE OF CONTENTS

# LIST OF FIGURES

# LIST OF TABLES

# LIST OF ALGORITHMS

# CHAPTER 1
# INTRODUCTION

In artificial intelligence, Multiple Object Tracking (MOT) refers to the task of locating objects in a scene and maintaining their trajectories throughout the entire video. It is an essential task for a broad range of computer vision applications such as surveillance, and autonomous driving. Pedestrians are among the most interesting subjects to track in public scene for many purposes such as safety and security. Therefore, lately, there have been a lot of research attention to pedestrian tracking [42, 11].

In general, MOT follows the tracking-by-detection paradigm. The state of the art of this paradigm is divided into two main categories; online tracking [18, 34, 43, 45, 47], and batch tacking [21, 46, 10, 51]. Online tracking is about associating object instances between the past frame and the current frame, whereas batch tracking uses the entire video frames to form a unique trajectory per ID.

The association task requires a similarity measure between the object instances. This similarity is based on two types of features; motion properties and visual features. The visual features are extracted from the visual appearance of the object instance in the bounding box using a feature extractor. Thus, the feature extractor plays an important role in tracking performance. Recently, MOT proposed approaches [34, 47, 46] heavily relies on training a neural network base extractor. These approaches make use of advances in Deep Metric Learning (DML) [9, 38, 36, 53] area in learning powerful visual representation. These base neural networks embed an input image into a sphere with a unit norm. The choice of the directional space is motivated by the assumption that the type of data is sometimes referred to as directional data, because the direction of the data provides more information than the magnitude. However, the learning method for patch representational learning is largely based on contrastive supervised learning in most MOT algorithms. These approaches rely on time-consuming mini-batch formats like triplets or N-pairs, which are incompatible with the MOT learning algorithm.

Machine learning-based person re-identification applications aim to successfully retrieve pedestrians by their identity for many security and safety tasks. Most of these application aims to learn a mapping function that embeds images into compact euclidean space (unit sphere mostly) [44, 23]. These embeddings should have the characteristic that two images of the same person map to adjacent feature points, while images of two distinct persons map to feature points that are far apart. These mapping approaches should, in an ideal world, be resistant to real-world situational changes like position, orientation, and occlusion in the same scene. It also should not rely on the clothes appearances, since pedestrians wear different clothes at different lapses of time (e.g,. days/weeks).

Most machine learning tracking and re-identification applications are heavily affected by image acquisition systems such as static cameras and the cost of collecting data. Unmanned aerial vehicles (UAV's), which enable a low-cost way of data collection while covering broad and difficult-to-reach regions, have recently been recognized as a feasible alternative for monitoring public spaces [8, 20, 27, 39]. The advancements in UAV's have benefited MOT, particularly pedestrian tracking and re-identification, since it gives a viable solution to solve numerous challenges such as occlusion, moving cameras, and difficult-to-reach locations. Compared to static cameras UAV's are flexible enough to adapt their emplacement location and direction in the 3D space.

Self-Supervised Learning (SSL) in machine learning refers to the task of learning from data without requiring labels. In order to take advantage of supervised learning frameworks, SSL problems are framed as a supervised problems. Recently, many researchers benefit from the success of DML and proposed approaches that are based on learning a representational feature in a supervised manner [31, 32, 17, 52, 12].

In open world environment, when a model trained and pushed to deployment, certainly, it will encounter new pedestrians that the model did not learn to recognize which called In Distribution (ID). Moreover, it will encounter objects different from humans. This new data points are called Out Of Distribution (OOD). The OOD data points are shifts from the ID that can be divided into types, non-semantic shifts (new pedestrian) or semantic shifts such as objects different from humans. The latter definition is adopted from [22]. Therefore, OOD detection is needed so that the trained models can work perfectly in real world scenarios.

To address the OOD detection problem, a straightforward idea is train a deep learning classifier on classifying ID vs OOD using real world OOD data points. However, sampling or generating real world OOD data points is difficult and intractable in the high dimensional pixel space. To overcome this challenge, [16] proposed a framework based on synthesising virtual outlier in the embedding space in an online manner. This framework is named Virtual Outlier Synthetic (VOS). The latter relies on the compact embedding of the samples from the same object to generate virtual outliers using class-conditional multi-variate Gaussian distribution. This assumption is consistent with the objective of the DML method.

Motivated by these facts, and to address the aforementioned issues, in this work, we investigate the application of directional statistics and SSL to MOT from aerial devices. First, we adapt the approach proposed by [53] for image classification and retrieval to the pedestrian tracking and identification from aerial devices problem. This approach is a simple and efficient method for learning a von-Mises-Fisher (vMF) distribution for each ID in the directional space. For spherical data, this distribution can be considered as a gaussian distribution. Learning a vMF for each ID helps simultaneously in measuring the similarity between object instances and identifying the person ID. Second, we used SSL to train a feature extractor, that is integrated in an online tracking system to measure the similarity between objects.

The remainder of this report is laid out as follows. We start by giving a background and an overview of related previous work in section $II$. Our proposed online pedestrian tracking and identification is presented in section $III$. In section $IV$, we present the dataset case study. Penultimate, we present our preliminary results. Finally, we outline potential future work and summarize our conclusions.

# CHAPTER 2
# RELATED WORK

## 2.1 Pedestrian Tracking

The application of deep learning in pedestrian tracking has been making significant strides in recent years, becoming a crucial aspect of urban safety systems for pedestrians. These algorithms have the ability to detect and monitor multiple individuals at the same time, leading to enhanced security measures and improved pedestrian management.

The integration of deep learning in pedestrian tracking has the potential to revolutionize the way we ensure the safety of people in public spaces. By employing advanced sensors and deep learning algorithms, it is now possible to accurately detect and track individuals in real-time, thereby providing a safer environment for everyone. Moreover, this technology can be used to optimize traffic systems by gaining deeper insights into pedestrian behaviors and patterns.

Pedestrian tracking systems that leverage deep learning are now being used to enhance public safety. These systems can detect and track pedestrians through unmanned aerial vehicles (UAVs) or other sensors, and can be used for a range of purposes such as traffic flow monitoring, suspicious behavior detection, and providing real-time alerts for potential hazards. Additionally, the use of deep learning in pedestrian tracking can help to reduce the number of false alarms generated by conventional security systems.

In machine learning, pedestrian tracking refers to the task of detecting and tracking humans from video scenes. It plays an important role in many safety systems such as video surveillance and autonomous vehicle driving [42, 11]. However, pedestrian tracking from a fixed camera suffers from several issues such as occlusion, access to area, and covering large areas. To overcome the aforementioned issues, unmanned aerial vehicles (UAV) such as drones provide a cheap and effective way for collecting data. The UAV can record the scene from different angles and follow the flow of objects. In recent years, tracking from UAV has gained a lot of attention from researchers. [8, 20, 27, 39].

## 2.2 Tracking Approaches

The task of finding objects and retaining their identities over all video frames is known as Multiple Object Tracking (MOT). It is a vital problem for a broad range of computer vision applications such as surveillance, and autonomous driving, as it requires tracking multiple objects simultaneously in real-time. The main challenge with MOT is to keep track of the objects while they move around and occlude each other. To address this challenge, researchers have developed various algorithms that use deep learning and other techniques to accurately track multiple objects over time.

In the recent years, MOT has been dominated by detection followed by tracking paradigm, where first detections are obtained, then, instances of the same object are linked together. An object instance from frame $t-1$ is at maximum associated with a single object instance from frame. $t$. The state of the art of this paradigm can be divided into two main sub-categories:

### 2.2.1 Online Tracking Mode

Online tracking mode [18, 34, 43, 45, 47] is a technique used to track objects in a video or image sequence. The latter is based on associating the detections between the past frame and the current frame. Usually, these methods are applied in real-time tracking scenarios tasks such as surveillance. Generally, this association problem is formulated as a bipartite graph problem [33]. The disjoint sets of vertices are the objects instances in frame $t-1$ and frame $t$. The edges between the nodes represent the cost (similarity) between their corresponding instances.

### 2.2.2 Batch Tracking Mode

Batch tracking is a method that can be used to track multiple objects in a video frame over time. This mode is based on processing all the detections available in the video frames, which can include individual objects or groups of objects. The goal is to make this process faster and more efficient by relying heavily on computer vision algorithms to generate the necessary data. Batch tracking methods use all possible frame detections [21, 46, 10, 51] to create a unique tracking of several objects from one frame to another. By doing so, the related object trajectories are established and used for long-term decisions or short-term predictions for further analysis or action.

Methods that are based on batch tracking mode use all the entire video frame detections to build a unique track (trajectory) per ID This is accomplished by associating past, present, and future directions. The association is generally formulated as a graph optimization problem such as maximum flow [2], or minimum cliques [50].

## 2.3 Pedestrian Re-identification

Pedestrian re-identification (Re-ID) is a crucial yet challenging problem in the field of computer vision. It requires the capability to recognize individuals across different cameras, which is of great significance in surveillance applications. To tackle this challenge, researchers have proposed various approaches over the years, ranging from traditional feature-based methods to state-of-the-art deep learning-based methods.

Feature-based approaches make use of manually designed features and traditional machine learning techniques to perform pedestrian recognition. In comparison, metric-based approaches employ deep learning models to learn discriminative features directly from images or videos of pedestrians. Each of these approaches possesses its own strengths and limitations in terms of accuracy and computational efficiency, making the choice of approach dependent on the specific problem requirements.

Re-ID research has a broad range of aspects spanning from feature-based [54] to metric-based [15] and from hand-crafted features to deeply learned features [28, 48]. In this report, we review three of the recent and relevant sub-research areas related to pedestrian re-identification problems.

Open-world person re-identification is a computer vision task aimed at establishing one-to-one correspondences between individuals in two distinct sets of images. It finds its applications in a broad range of scenarios where the accurate recognition of people is essential, including security and surveillance, tracking of individuals' movements, and analysis of behavior patterns in public spaces such as airports, shopping malls, and other similar environments.

To achieve the goal of accurately recognizing individuals across different images, researchers have proposed various techniques, including feature extraction, image processing, and deep learning algorithms. Open-world person re-identification is based on a one-to-one set matching approach, where the goal is to match individuals present in both the probe and gallery sets. The challenge in this problem lies in the need to accurately match the same person across images taken from different angles or locations, and the pedestrian set is assumed to be known beforehand.

Generalized-view re-identification approached is mainly based on discriminative learning from two different views acquired by two different fixed cameras [1, 25]. However, in practice this problem is costly since it requires data collected, annotated and matched from two different cameras.

Pedestrian re-identification from drone is a new tool that has been developed to help with the identification of pedestrians in public places. By using drones, it is possible to detect and recognize pedestrians in an efficient manner. This technology can be used for various applications such as security and surveillance, crowd management, and pedestrian counting. It also helps with the retrieval of lost persons in case of emergency situations. With this new tool, it is now possible to identify individuals quickly and accurately from a distance.

Recently, pedestrian re-identification from drones gained a lot of attention and new benchmark datasets are published [26, 27]. Drones provide a new tool for data acquisition, especially for video surveillance and analysis. With this new tool, problems such as pedestrian detection, tracking, and re-identification can be taken to a next challenges as it helps overcome some of the static camera issues.

### 2.4 Deep Metric Learning

Deep learning is a machine learning technique that can be applied to many different problems. It's a great way to gain an understanding of the meaning of data and how it can be used to make predictions and automate tasks. Deep learning uses multiple layers of deep neural networks, which are trained using back-propagation methods. Many machine learning tasks require learning a measure of similarity between data objects such as MOT. The goal of metric learning is to learn a mapping function that quantifies the similarity between data points. The metric learning objective is to minimize the similarity between data points from the same category and maximize the distance between data points from different categories.

In recent years, deep learning has gained enormous success in a variety of machine learning tasks such as image classification, image embedding, and multiple object tracking. Deep learning brought revolutionary advances due to its representational power in extracting high abstract non-linear features. This fact led to a new research area known as Deep Metric Learning (DML) [9, 38, 36, 53].

MOT has been revolutionized by the success of deep metric learning (DML) models. By training a neural network feature extractor, MOT has benefitted from the ability to learn a similarity measure between different objects in an image or video. This similarity measure can be used to track objects over time, allowing for more accurate and robust tracking of multiple objects in a scene.

### 2.5 Directional Statistics in Machine Learning

Directional data is defined as points in a Euclidean space with norm $\|\mathbf{x}\|_2 = 1$, where $\|.\|_2$ is the Euclidean norme 2. In other words corresponding to points on the surface of the unit sphere. Thus, statistics that deal with directional data are called directional statistics.

The topic of directional statistics has gained a lot of attention due to demands from fields such as machine learning or the availability of big data sets that necessitate adaptive statistical methodologies, as well as technical improvements. Recently, directional statistics method has led to a tremendous success in many computer vision tasks, such as image classification and retrieval [53], pose estimation [35] and Face Verification [19]. It has also been introduced to other machine learning fields such as text mining [40].

### 2.5.1 Von Mises-Fisher Distribution

Von Mises-Fisher Distribution (vMf) is a probability distribution function for directional data. It can be seen as a Gaussian distribution since they have very similar properties. It can be used to model data that has an underlying direction or orientation, such as wind direction or geographic coordinates. The vMf is useful for analyzing directional data in applications such as image processing, geostatistics, and weather forecasting.. In a directional data space $\mathbb{S}^{P-1}$, the probability distribution density function defined as:

$$f_p(x; \mu, \kappa) \quad = \quad Z_p(\kappa) \exp(\kappa \mu^T x). \tag{1}$$

where, $\mu$ is the mean direction of the distribution, $\kappa \geq 0$ is a concentration parameter which can be seen as the standard deviation for Gaussian distribution, $p$ is the space dimension, $Z_p(\kappa) = \frac{\kappa^{p/2-1}}{(2\pi)^{p/2} I_{p/2-1}(\kappa)}$ is a normalization term and $I_v$ is the modified Bessel function of the first kind with order $v$.

Given $N$ samples from a vMF distribution, we can estimate its parameters as follows:

$$\hat{\mu} \quad = \quad \frac{\sum_{i=1}^N x_i}{\| \sum_{i=1}^N x_i \|_2}, \tag{2}$$

and

$$\hat{\kappa} \quad = \quad \frac{\overline{R}(p - \overline{R}^2)}{1 - \overline{R}^2}. \tag{3}$$

In (16), $\overline{R} = \frac{\| \sum_{i=1}^N x_i \|_2}{N}$.

### 2.5.2 Learning von-Mises Fisher Distribution

The learning problem is defined as follows. Given $C$ identities, the goal is to learn a vMF distribution for every ID parameterized by $\{\kappa_i, \mu_i\}$, where $i = 1..C$.

Given a point $x$ in the mapping space, the normalized probability of x belonging to a chosen class $c$ is defined as

$$P(c | x, \{\kappa_i, \mu_i\}_{i=1}^C) = \frac{Z_p(\kappa_c) \exp(\kappa_c \mu_c^T x)}{\sum_{i=1}^C Z_p(\kappa_i) \exp(\kappa_i^T \mu_i x)} \tag{4}$$

Equation (4) can be used to increase the likelihood that the sample belongs to the correct class while decreasing the likelihood that it belongs to other classes. Given a mini-batch with $N$ samples and for a $C$ identity, we can maximize the following objective function:

$$\begin{aligned} P(Y|X, \Theta, \cup, \kappa) &= \prod_{n=1}^{N} P(c|x, \{\kappa_i, \mu_i\}_{i=1}^{C}) && (5) \\ &= \prod_{n=1}^{N} \frac{Z_p(\kappa_c)\exp(\kappa_c \mu_c^T x)}{\sum_{i=1}^{C} Z_p(\kappa_i)\exp(\kappa_i \mu_i^T x)}, && (6) \end{aligned}$$

Where $X$ and $Y$ represent the data points in the mini-batch and their ID labels, $\Theta$ contains the deep model parameters, and $\cup = \{\mu_i\}_{i=1}^{C}$, $\kappa = \{\kappa_i\}_{i=1}^{C}$. For a simplification purpose, we assumed $\kappa$ to be a constant for all IDs, and by applying the negative likelihood, equation (6) can be simplified to:

$$\underset{\Theta, \cup}{\arg\min}\ L = -\sum_{n=1}^{N} \log\left(\frac{\exp(\kappa \mu_c^T x)}{\sum_{i=1}^{C}\exp(\kappa \mu_i^T x)}\right) \tag{7}$$

Since it difficult to simultaneously optimize the neural network parameters $\Theta$ and the vMF mean direction distributions $\cup$, in [53] they proposed a learning algorithm (algorithm 1) that is based on alternative learning. In this algorithm, the mean directions are fixed, while training the neural network parameters for several iterations, then updating them using the training data set. The mean direction update is based on the estimation using all training data points. The algorithm converges when the mean directions and loss are stagnant. Given a class $i$ with $N$ training data points. Let $x_n$ denotes the mapping of the $n^{th}$ sample using the current mapping function, where $n = 1..N$. The mean direction of class $i$ can be updated as follows:

$$\hat{\mu_i} \quad = \quad \frac{\sum_{n=1}^{N} x_n}{\|\sum_{n=1}^{N} x_n\|_2}, \tag{8}$$

**Algorithm 1** vMF learning algorithm

```
1. Initialize CNN parameters $\Theta$.
2. Repeat:
    (a) Estimate mean directions using (8) and all the training data.
    (b) Train CNN for several iterations and update $\Theta$.
3. Until convergence.
```

During the inference phase, we can predict the ID of a given object by measuring cosine similarity with the learned mean directions. The object will be assigned with the ID label of its nearest mean vector.

### 2.6 Self Supervised Learning

Self-supervised learning (SSL) is a cutting-edge approach in machine learning that harnesses unsupervised techniques to automatically extract valuable information from data, eliminating the need for manual labeling or annotation. This innovative method leverages the inherent structure in data to uncover patterns and gain insights, while also improving the generalizability of the model on unseen data. In recent years, self-supervised learning has gained significant traction due to its ability to perform complex tasks with minimal human intervention. This approach works by introducing a proxy task, such as predicting the next frame in a video sequence or guessing the missing words in a sentence, to train the model and develop robust representations of the data. By learning about the structure and context of the data through this proxy task, self-supervised learning enables machines to acquire a deeper understanding of the information without the dependency on explicitly labeled data.

The SSL problems are framed as a supervised learning problem in order to apply supervised learning algorithms to solve it. In the recent years, many researchers benefit from the success of DML and proposed approaches that are based on learning a representational feature in a supervised manner [31, 32, 17, 52, 12].

The self-supervised learning approach of SimClr presents a novel way of enabling machines to learn from unlabeled data. This approach utilizes the concept of contrastive learning, where representations of the data are learned in an unsupervised manner. The use of SimClr has been demonstrated to enhance the performance of various natural language processing (NLP) tasks such as text classification, sentiment analysis, and question answering. Additionally, this method provides a new way of pre-training deep neural networks for computer vision tasks, including image recognition and object detection. Through the utilization of self-supervised learning, SimClr enables machines to better comprehend visual and textual data without the need for human-annotated labels.

SimClr is a simple self-learning framework proposed by[12] to learn visual representational. This framework combines advances in deep metric learning with data augmentation operations to train a neural network base encoder that embeds a data point into a directional data. Directional data is defined as the Euclidean space with norm $\|\mathbf{x}\| = 1$. It has garnered enormous success in a short period regarding its simple and solid idea. The four primary components of this framework are illustrated in figure 1.

1. A data augmentation module that takes as input a data point $x$ and outputs two different augmented, yet correlated data points.

2. A CNN based encoder $f(.)$ that maps the augmented data points to the representational space.
3. A neural network projector $g(.)$ that maps representational to the unit sphere.
4. The contrastive loss function that minimizes the distance between positive pairs (augmented pairs of the same input $x$) and maximizes the distance between negative pairs.

$N$ samples are randomly selected from the dataset to form a mini-batch, then using the augmentation module, $2N$ samples are derived from them. Each augmented pair of the same input $x$ is treated as a positive pair, while the rest $2 * (N - 1)$ are treated as negative samples. More details about the loss function and the training algorithm are in the original paper [12].

Motivated by the success achieved by SimClr, we chose it as a framework to train the feature extractor in order to measure the similarity between object instance patches.

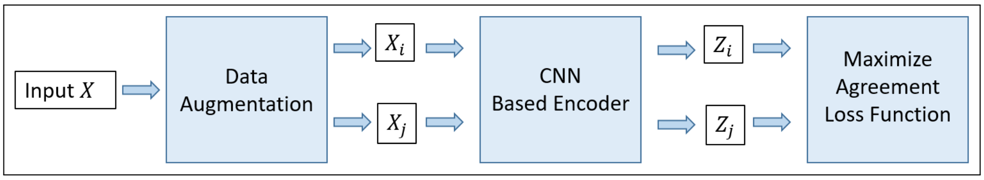
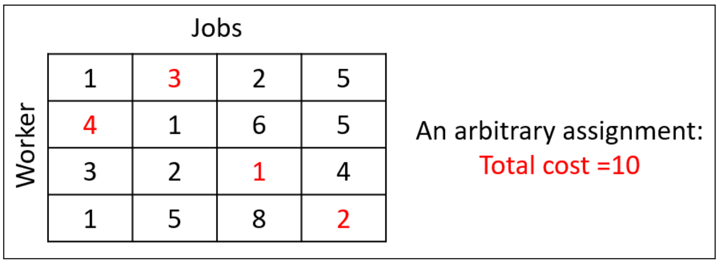


Figure 1: Illustration of SimClr framework for Contrastive learning of visual representations. Two different images of the same input are generated using a data augmentation module. The two augmented images are mapped through the CNN based encoder. A loss function designed to maximize the agreement between the image projections.

### 2.7 Hungarian Algorithm

The Hungarian Algorithm is a widely utilized solution in Artificial Intelligence for solving optimization problems, specifically the assignment problem. This algorithm is a means of finding the optimal assignment of tasks to agents while minimizing costs, making it a powerful tool in various fields including transportation, resource allocation, and scheduling. Developed by Harold Kuhn in 1955, the Hungarian Algorithm has since been widely adopted and applied in various industries, due to its ability to maximize profits or minimize costs when assigning tasks to agents. It operates by iteratively finding the best solution to the problem at hand, resulting in an optimal assignment of tasks to agents that minimizes costs or maximizes profits.

The Hungarian algorithm solves the assignment problem using a combinatorial optimization technique. Given a non-negative cost matrix in the matrix formulation, with the element in the $i^{th}$ row and $j^{th}$ column representing the cost of allocating the $j^{th}$ task to the $i^{th}$ worker. We must discover a method of assigning jobs to employees in which each job is allocated to one worker, each worker is assigned one job, and the overall cost of assignment is kept to a minimum. The Hungarian algorithm can optimally solve this problem. Figure 2 illustrates this problem. This algorithm has been used in online MOT literature review [18, 34, 43, 45, 47] to associate between object detections, where object from past frame can be seen as workers and object from current frame can be seen as jobs.



| | | | |
|---|---|---|---|
| 1 | 3 | 2 | 5 |
| 4 | 1 | 6 | 5 |
| 3 | 2 | 1 | 4 |
| 1 | 5 | 8 | 2 |

An arbitrary assignment:
Total cost =10

Figure 2: Illustration of cost assignment matrix that can be solved optimally by Hungarian Algorithm.

### 2.8 Out Of Distribution Detection

Out of Distribution Detection (OOD) is a crucial aspect of artificial intelligence and machine learning systems. It refers to the ability of the algorithm to recognize when it is presented with data points that are not representative of the training data. The detection of OOD instances is vital for ensuring accurate classification and prediction outcomes, as well as avoiding overfitting of the model. In this sense, OOD detection is a critical factor in maintaining the robustness and reliability of AI systems by avoiding false positives and false negatives. The ability to distinguish between in-distribution and out-of-distribution samples plays a crucial role in avoiding generalization errors and maintaining the predictive performance of machine learning models.

The OOD problem can be considered as classification task, with $D_{in} = \{(X_i, y_i)\}_i^N$ denoting the ID, $X_i \in \mathcal{R}^k$, and $y_i \in \{1..C\}$ for $C$ classes. A distribution $p_{in}$ is used to generate $D_{in}$. To infer class probabilities, most classification functions, such as deep neural networks, are trained on $D_{in}$ datasets. When the model is deployed in production in the open world, it encounters data derived from a different distribution $p_{out}$ during inference. $p_{in} \neq p_{out}$ These two distributions are not the same. A straightforward idea is sampling from $p_{out}$ distribution. However, sampling from the high dimensional pixel space is very difficult and intractable.

The difference between out $p_{in}$ and $p_{out}$ can be categorized into two main categories. A semantic shift and non-semantic shift. A semantic shift means a new type of class appears in the $p_{out}$, while a non-semantic shift means objects instances from the same class that exist in the $p_{in}$ appears differently comparing to the training instances. The latter is more similar to the anomaly detection setup. In order for model to work in the production environment, it needs to detect these types of shift in the data.

To address the aforementioned issue, recently, many studies has tackled this problem in a classification task setup. The majority of these studies built a binary scoring function $\mathcal{S}(x)$, where high score is assigned to data points from the ID and low score is assigned to OOD samples. This scoring function can be learned based on energy models [TO ADD], linear transformation using deep neural network, or both [VOS].

### 2.9 Anomaly Detection

Anomaly detection, in its broadest definition, is the detection of unusual events, items, or observations that are suspicious because they deviate dramatically from expected behaviors or patterns. Outliers, noise, and new objects are detected as anomalies in computer vision when compared to the distribution of the known objects. It occurs in many industrial contexts where obtaining images of normal samples is simple, but specifying the expected defect variations in detail is costly and difficult. This problem is sometimes referred to as an out-of-distribution (OOD) detection problem, in which a model must discriminate between samples obtained from the training data distribution and those outside its support. Existing work in anomaly detection is based on learning a compact visual representational features in a latent space via auto-encoder, GANs. There are also the unsupervised methods that are based on pretrained CNNs [Patchcore SPADE, PaDIM], which are mostly used in the industrial application.

Anomaly detection is a key element of pedestrian re-identification. It is the process of identifying and classifying abnormal or unusual behavior in data sets. Anomaly detection can be used to identify pedestrians who are out of place or have suspicious behaviors, helping to improve security and reduce crime. By detecting anomalies, it is possible to identify potential threats before they happen and take preventive measures. Another use case, pedestrians being re-identified are from the training data support and pedestrian that are out of the defined list of re-identification can be seen as outliers (anomalies).

### 2.10 Visual Outlier Synthetic

Visual Outlier Synthetic is a machine learning technique that uses computer vision to detect anomalies in a dataset. It works by analyzing the visual features of the data and then using these features to identify patterns that are not typical of the dataset. This technique can be used for a variety of purposes, such as detecting fraud or identifying outliers in large datasets. Visual Outlier Synthetic can also be used to improve accuracy in machine learning models, as it can help identify which features are most important for predicting outcomes.

Outlier synthesis is a powerful technique used to generate synthetic data points. In computer vision, synthesising images has been tackled by many different approaches. The most straightforward and widely used approach is using Generative Adversarial Networks GANs [37]. However, synthesising images in the high dimensional pixel space is difficult to optimize and to track. To address this issue [16] proposed a framework named "Visual Outlier Synthetic" (VOS) that synthesises virtual outliers in the embedding space on an online manner. Their approach relies on the model learning the embedding of the ID objects to generate hard virtual outlier.

VOS assumes that the features representation in the embedding space of the object instances follows a class-conditional multi-variate Gaussian distribution. Where objects from the same class forms a multi-variate Gaussian distribution in the latent representational space.

### 2.10.1 Learning of class-conditional multi-variate Gaussian distribution

Computer vision is a subfield of artificial intelligence that deals with the ability of computers to interpret and analyze digital images. It encompasses a wide range of techniques aimed at extracting valuable information and insights from digital images, including objects, faces, and text. Coreset of objects, which is an important tool in computer vision, plays a critical role in the feature extraction process by allowing machines to efficiently recognize different objects in images. This is achieved by selecting a representative subset of data points from the dataset and using it to train machine learning models for object recognition tasks. By utilizing the coreset of objects, machine learning algorithms are able to achieve greater accuracy and efficiency in identifying and classifying various objects present in an image.

Given a coreset that represents the embeddings of the objects, a class-conditional Gaussian distribution can be learned from the coreset by estimating its parameters. Thus from the coreset they computed the empirical class mean $\hat{\mu}$ and covariance $\hat{\Sigma}$ of training samples $\{(x_n, y_i)\}_{i=1}^{N}$ as follows:

$$\hat{\mu_k} \quad = \quad \frac{1}{N_k} \sum_{n:y_n=k} x_n. \tag{9}$$

$$\hat{\Sigma_k} \quad = \quad \frac{1}{N} \sum_{k} \sum_{n:y_n=k} (x_n - \hat{\mu_k})(\hat{\mu_k} - x_n)^T. \tag{10}$$

Where $N_k$ is the number of objects in class $k$, and $N$ is the total number of objects.

### 2.10.2 Sampling from the features representational space

The exploration of the feature representational space through sampling has gained significant interest in recent times as it provides a deeper understanding of the interactions and relationships between various data points. The ability to sample from the feature representational space provides valuable insights into the creation of a more accurate representation of the data.

One approach that has been proposed by researchers is the sampling of virtual outliers using multivariate distributions from the feature representational space. This approach enables the detection of outliers that may not be visible in traditional univariate analysis, thereby providing a deeper understanding of the inter-feature relationships and their impact on the results.

The implementation of this sampling technique leads to a better understanding of the data and enhances the ability to make informed decisions regarding the models. It also identifies potential areas for improvement and optimization, thereby elevating the performance of the models.

Using the above-mentioned multivariate distributions, they suggested sampling the virtual outliers from the feature representation space. These virtual outlier are generated in an online manner. The more the learning progresses, the more the embeddings of each class are compact. These virtual outliers aligns with this objective and helps in learning a more compact embedding. This accomplished by sampling virtual outliers from the learned class-conditional distribution:

$$f(x_n) = \frac{1}{(2\pi)^{\frac{m}{2}} |\hat{\Sigma}|} \exp\left( -\frac{1}{2}(x_n - \hat{\mu_k})^T |\hat{\Sigma}|^{-1} (\hat{\mu_k} - x_n) \right). \tag{11}$$

$$\nu_k \quad = \quad \{\nu_k | f(\nu_k) < \epsilon\}. \tag{12}$$

Where $\nu_k \sim \mathcal{N}(\nu_k, \hat{\Sigma})$ denotes the sampled virtual outliers for class $k$, which are in the sublevel set based on the likelihood. $\epsilon$ is sufficiently small so that the sampled outliers are near class boundary.

### 2.10.3 Out of Distribution Detection

VOS framework used linear transformation to learn to distinguish between virtual outliers. This was achieved using Fully Connected Pooling (FCP) layer that learn the tight boundaries around the ID. More details about the classification between ID and OOD and the learning algorithm exists in the original paper.

# CHAPTER 3
# PRELIMINARIES

Computer vision can be defined as the science field that deals with digital images and sensors to derive high abstract level understanding from images and videos content. From an engineering standpoint, it aims to comprehend and automate operations that the human visual system can do. In artificial intelligence and machine learning, computer vision is the the subfield in which engineers teach computers to learn, understand and interpret the world around them through the acquired images.

Computer vision tasks encompass approaches for acquiring, processing, analyzing, and understanding digital images as well as the extraction of high-dimensional data from the actual world in order to provide numerical or symbolic information, such as decisions. Large-scale formalization of challenging problems into well-known, defensible problem statements was a hallmark of the development of machine vision. Researchers from all over the world were able to recognize issues and effectively address them thanks to the topical division into well-defined categories with appropriate nomenclature. The most popular well defined computer vision tasks are: image classification, object detection, object segmentation, tracking and recognition. These tasks can be divided to two categories: first, tasks to derive decisions on single image and second tasks to derive decisions on a sequence of images such as video clips.

In this chapter, we describe the computer vision tasks used in our work. The reminder of this chapter is outlined as the following: first, we explain the definition of image classification. Second, we give an overview of object detection by focusing on detecting pedestrians. Third, we explain object tracking. Finally, we describe the pedestrian re-identification.

## 3.1 Classification

Classification is a process of categorizing data into different classes based on certain criteria. It is an important task in machine learning and computer vision, as it helps to identify objects, activities, and trends in large datasets. Classification algorithms can be used to classify images, videos, audio files, text documents, and other types of data. By using classification algorithms, we can gain insights about the data that would otherwise be difficult or impossible to obtain. Despite its apparent simplicity, image classification is the most common computer vision task undertaken by both beginners and experts, in computer vision and has a wide range of real-world applications.

To concertedly understand image classification, figure 3 contains an example that illustrates image classification task. Given a set of pre-defined classes dog, cat, person and an input image (dog), the task is to assign a one of the classes to the input image. The image example contains a dog, ideally, the class dog should be assigned.

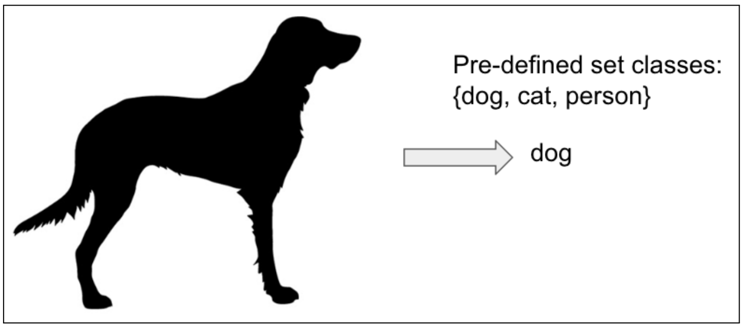


Figure 3: An illustration of image classification problem. Given a set of pre-defined classes dog, cat, person and an input image (dog), the task is to assign a one of the classes to the input image. The image example contains a dog, ideally, the class dog should be assigned.

### 3.2 Object Detection

Image classification mainly assumes that the image must contains only a single object of the predefined categories, while in real world scenario, the image can contain more than object from the categories that exist in our life. Thus, a more complex tasks is needed to localize and classify the object. Object detection is an important application of machine learning and computer vision. It involves the use of algorithms to detect, classify, and localize objects in images or videos. Object detection can be used in a variety of applications such as autonomous vehicles, facial recognition systems, security systems, medical imaging, and robotics. With advances in machine learning techniques such as deep learning and convolutional neural networks, object detection has become increasingly accurate and efficient. In this article we will discuss the fundamentals of object detection and its various use cases. Object detection is one of the most studied topics in computer vision, where the task is to localize the objects using a bounding boxes. Then, each object is classified to one of the pre-defined classes. Figure 4, shows an example that explain what the output of a trained detection model should look like.

During the recent years, deep learning model achieved a tremendous success in solving many object detection problems in different fields. Methods like YOLO, RCNN, and SSD demonstrated a great capabilities in detecting and recognizing a wide range of objects spanning from daily seen objects to objects exist only in industrial environments.

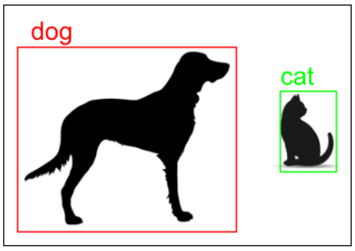


Figure 4: An illustration of object detection problem. Given a set of pre-defined classes dog, cat, person and an input image (contains a dog and a cat), the task is to localize all object in the input image. The image example contains a dog and a cat, ideally, the trained model should localize both the cat and the dog with a bounding box for each and assign the correct class for them.

### 3.3 Object Tracking

The task of finding objects and retaining their identities over all video frames is known as Multiple Object Tracking (MOT). It is a vital problem for a broad range of computer vision applications such as surveillance, and autonomous driving. Pedestrian are among the most interesting objects to track. Figure 5 illustrates an example of online pedestrian tracking. Given two successive frames $t-1$ and $t$. First an object detection model is applied on each frame to detect the objects. Then, given the detected locations in the two frames, the tracking algorithm is applied to match between the object instances in the two frames. In the example we have two object instances in each frame. Therefore there are two IDs 1 and 2. In other words, the object tracking task is the task that studies the movement of the objects in a scene such as pedestrians

Opposite to image classification and object detection, object tracking might face several challenges at different aspects. First, challenges related to object movement in the scene such as occlusion background clutter. Occlusion is the phenomenon of interference between objects while they are moving in the scene which can cause the tracking algorithm to lose track of the occluded objects. More the objects get closer and occlude more the tracking gets harder. Clutter in the background can cause issues to the detection and tracking algorithm.

The effect of the aforementioned issues differs between fix and moving cameras. For moving cameras such as used in drones, the speed and direction of the drone controlled by the operator can help overcome some of the challenges like occlusion by flying in the same direction on top of the pedestrian while they are walking. However, this can introduce a new challenge for tracking algorithms such as the estimation of the next object motion.

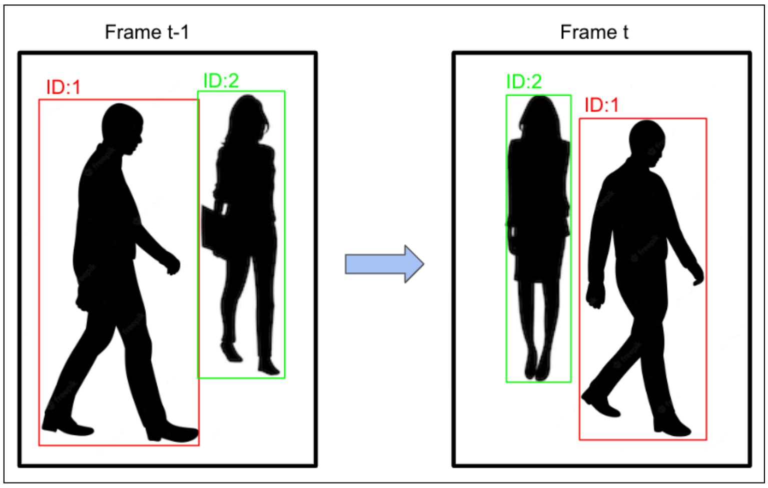


Figure 5: An illustration of pedestrian tracking problem. Given two consecutive frame each contains two instances of the same pedestrians, the task is to localize two instances, then, match them. The localization happens through object detection model first. Then, the association is done by the tracking algorithm.

### 3.4 Pedestrian re-identification

In computer vision, matching the same pedestrian across different acquired camera views is referred to pedestrian re-identification. Open-world person re-identification is defined as one-to-one set matches. Given two sets of pedestrian, the first called probe and the second called gallery, every person appears in both sets, and the task is to match between them [24]. In this problem, the pedestrian set must be known.

Recently, pedestrian re-identification from drones gained a lot of attention and new benchmark datasets are published [26, 27]. Drones provide a new tool for data acquisition, especially for video surveillance and analysis. With this new tool, problems such as pedestrian detection, tracking, and re-identification can be taken to a next challenges as it helps overcome some of the static camera issues.

### 3.5 Pedestrian tracking and Re-identification Deep Learning Framework

In daily real world scenario, humans take decisions over the environment around them using their vision. In order to model the decision using the current research state in machine learning vision, complex tasks in computer vision are designed. These complex tasks are usually frameworks that combines several trained machine learning model, where each model is trained under one of the basic tasks. In an inference model, these tasks are pipe-lined together to address certain complex task. Figure 6 shows an example of pedestrian tracking and re-identification framework, where we need three models trained to do the task in an inference model.

There are approaches, that tries to simplify these frameworks by merging some of the tasks under one single algorithm. This merging not only helps in training an end-to-end complex frameworks, but also having more fast inference.

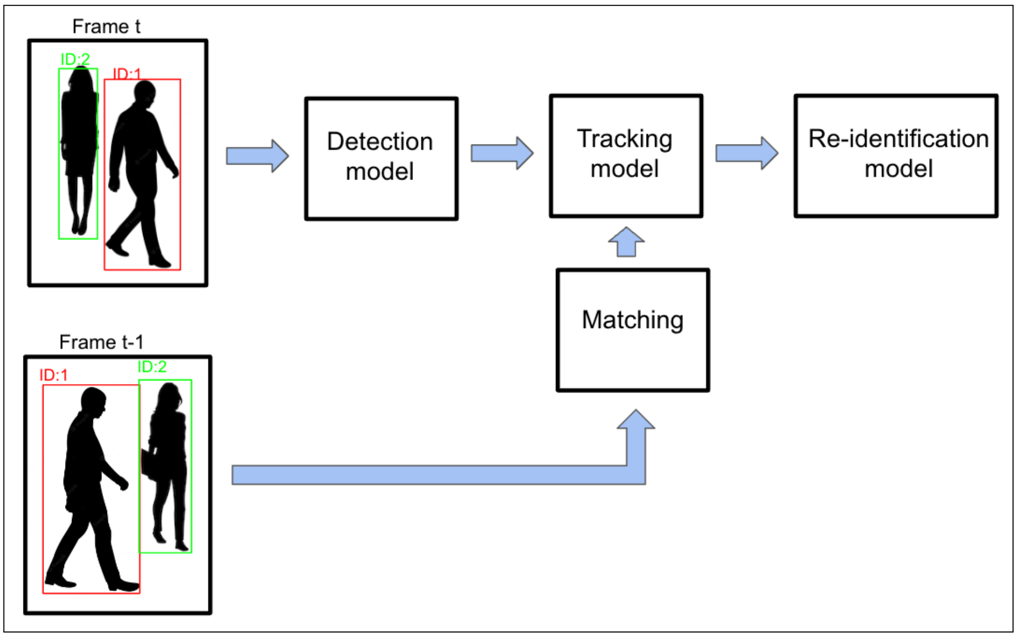


Figure 6: An illustration of pedestrian tracking and re-identification problem. Three models needed to be trained first. Then, they are pipe-lined together to work in inference mode to do the task. First, a detection model to detect pedestrians locations. Second, a tracking algorithm to track the movement of the pedestrians. Third, a pedestrian recognizer to re-identify persons.

# CHAPTER 4

# DATASET DESCRIPTION

Over the recent years, pedestrian tracking and re-identification topic has gained a lot of interest due to its application, such as surveillance systems and traffic control. However, the acquisition system is limited by the stationary camera, which represents a main issue for the tracking and especially the person re-identification. In recent years, Unmanned Aerial Vehicles (UAV's) have been viewed as a viable option for monitoring public areas, as they provide a low-cost method of data collection while covering large and difficult-to-reach areas.

The development and improvement of machine learning and computer vision algorithms require robust and extensive data sources. Pedestrian tracking and re-identification datasets play a crucial role in this regard by offering a comprehensive platform for training and evaluating models. These datasets provide a wealth of information captured through aerial images taken by Unmanned Aerial Vehicles (UAVs) that can be utilized to train models to recognize pedestrians.

Through the utilization of these datasets, researchers can effectively develop algorithms capable of accurately tracking and identifying individuals from aerial images taken by UAVs. This, in turn, leads to the advancement of pedestrian tracking technology and its widespread application in various domains. Furthermore, these datasets provide valuable insights into the influence of environmental factors such as lighting, terrain, and weather conditions on the performance of pedestrian tracking algorithms, which is crucial information for developing and improving these algorithms.

The **P-DESTRE** [26] dataset is a comprehensive collection of data for machine learning and computer vision applications, specifically related to Unmanned Aerial Vehicles (UAVs). The dataset contains images from different UAVs, with annotations that provide information on the objects in the images. This data can be used to train algorithms for object detection and tracking, as well as other computer vision tasks. The **P-DESTRE** dataset is a fully annotated dataset for detecting, tracking, re-identification, and searching pedestrians from aerial devices. A group of researchers from the University of Beira Interior (Portugal) and the JSS Science and Technology University gathered this data (India). They recorded packed sights on both institutions campuses using drones called "DJI Phantom 4"[1]. These drones, as shown in Figure 1, are piloted by humans to fly and collect data from a volunteer audience walking at altitudes ranging from 5.5 to 6.7 meters. In total, 75 videos with a Frame Per Second (fps) equal to 30 are collected. In these videos there are $318,745$ annotated instances of 269 different IDs. These statistics are summarized in table $I$.

[1] https://www.dji.com/phantom-4

| | |
|---|---|
| Total number of videos | 75 |
| Frame Per Second (fps) | 30 |
| Total number of identities | 269 |
| Total number of annotated instances | $318,745$ |
| Camera range distance | $[5.5 - 6.7] metres$ |

Table 1: P-DESTRE Dataset Statistics Summary.

The pedestrian search challenge, in which data is collected over long periods of time (e.g., days/weeks), with constant ID labels across observations, is the primary distinguishing characteristic between the P-DESTRE and comparable datasets. The re-identification techniques in this problem cannot rely on clothing appearance-based features, which is a key property that distinguishes search from the (less difficult) re-identification problem, in which the consecutive observations of each ID are assumed to have been taken in short intervals of time and clothing appearance features can be reliably used.

This dataset, we believe, is an excellent case study for training and evaluating our frameworks for pedestrian tracking and re-identification from aerial devices. This dataset was used for all of the experiments. We plan to look at more aerial datasets in the future.

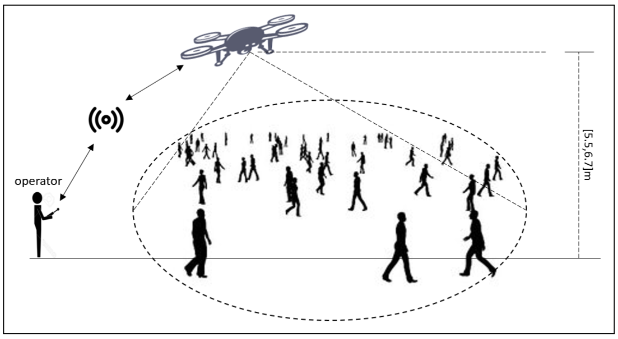


Figure 7: The P-DESTRE datasets were obtained using a consistent data gathering technique. Human operators flew "DJI Phantom 4" aircraft at altitudes ranging from 5.5 to 6.7 meters to mimic autonomous surveillance of urban scenes. The gimbal pitch angle ranged from 45 to 90 degrees [26].

# CHAPTER 5

# TRACKING and RE-IDENTIFICATION METHODOLOGY

In this chapter, we present the proposed pedestrian tracking and re-identification framework. The latter consists of three main consecutive steps: 1) detection 2) tracking 3) re-identification. The first step, is about detection in which we used YOLO V4[7]. The second and third steps are about tracking and re-identification for which, we propose an approach to couple them together under one single algorithm. The synergy between tracking and re-identification is used to improve the framework overall performance. To validate and analyze the proposed approaches, we designed several experiments to gauge a variety of metrics ranging from detection performance to tracking and re-identification of pedestrian. In the following, we first describe the framework pipeline. Then, we describe the proposed object association and re-identification approach. Finally, we illustrate the designed experiments used to validate and analyze the framework.

## 5.1 Multiple Object Tracking and Re-identification Framework

Many deep learning approaches have become increasingly popular in tracking and re-identification over the past few years. These approaches rely on a consecutive manner where tracking is performed first then re-identification is performed second. In this work, we propose a single framework that uses deep learning to perform these tasks together. The proposed framework is able to accurately track and re-identify objects in a given sequence of frames. The pipeline of our suggested approach that is different from the generic MOT task paradigm of object detection followed by tracking. In the generic MOT task paradigm, the re-identification task is performed after tracking. In our framework, we propose an approach that merges both tracking and re-identification. This merging benefits the framework at many performance aspects spanning from inference time to tracking and identification. In summary, the framework can be seen as a two consecutive models, a detector model and recognized model.

### 5.1.1 Framework in Inference Phase

For the purpose of illustrating how the framework works, in this phase we assume that we have the necessary neural network models are trained (section 5.1.2 provides the necessary details for training these models). Figure 2 illustrates the structure diagram of this framework in the inference phase. It comprises of 4 major steps:

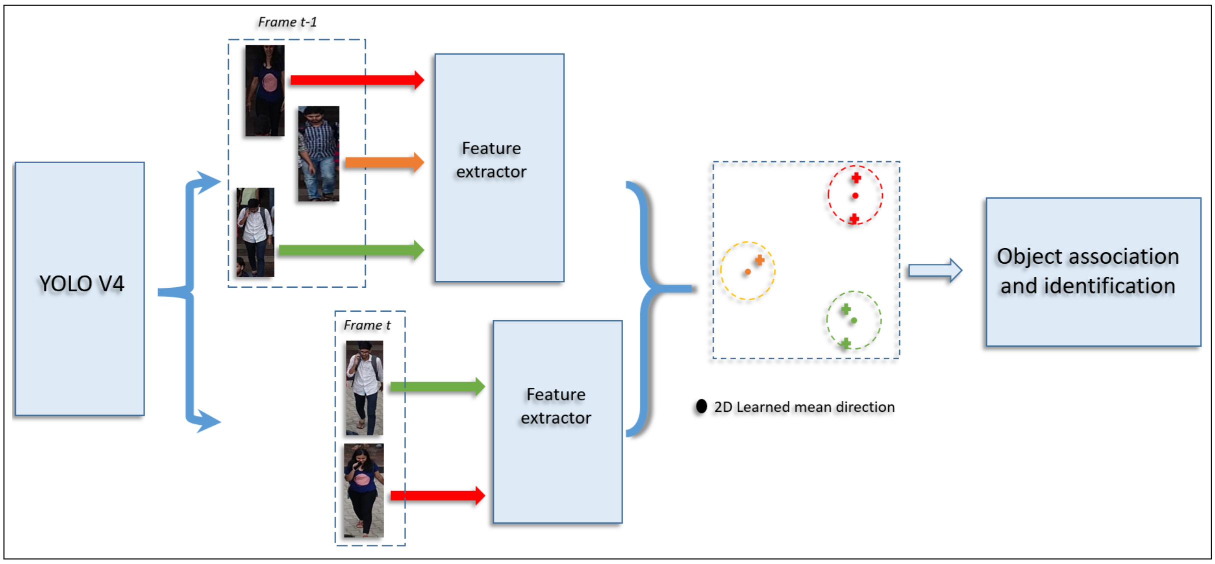

Frame t-1
Feature extractor
YOLO V4
Object association and identification
Frame t
Feature extractor
● 2D Learned mean direction

Figure 8: Multiple object tracking online framework. First, patches from frame $t$ are extracted and resized to the same shape. Second, these patches are mapped through the trained feature extractor network to obtain the visual descriptors. Third, a pairwise similarity measure is conducted between the object descriptors of frame $t$ and others of frame $t-1$. Finally, an association algorithm is performed to match object instances based on their similarity.

#### 5.1.1.1 Detection and Localization

We used YOLO (You Only Look Once) V4 for real-time object detection. The YOLO V4 detector receives input videos frame by frame. As an output, the detection bounding boxes in each frame are obtained. YOLO is widely used to solve object detection problems in research and industrial fields due to its tremendous success in accurately locating objects of intersect in images. YOLO is reportedly to be among the fastest real-time object detectors. YOLO V4 is a remarkable breakthrough in deep learning and object detection technology. It has demonstrated promising results in its ability to perform better accuracy, faster speed, and good performance on pedestrian detection. YOLO was created with two primary goals: providing a rich user experience while not compromising on accuracy or compute resources. Through its advanced architecture and optimizations, YOLO V4 successfully achieves these goals while pushing the boundaries of object detection to new heights.

#### 5.1.1.2 Pedestrian Object Extraction

The bounding box has been a key tool in object detection. It consists of a rectangle that encloses the object of interest. With the help of artificial intelligence and deep learning, bounding boxes have become even more powerful, allowing us to accurately detect objects in an image. Patches are cropped from each frame using the bounding boxes that have been detected using YOLO V4.

#### 5.1.1.3 Shape Resizing

Resizing images is an important task in digital image processing, especially when working on images of different sizes and shapes such as the detected objects of different sizes in the scene (a frame at a time). Resizing to a unified shape allows for easier comparison and utilization of the images. Therefore, these patches are resized to the unified shape $(H, W, C)$, where $H$ denotes height, $W$ denotes width, and $C$ denotes channel count. A feature vector is then generated for each detected object using the trained features extractor.

#### 5.1.1.4 Association and Re-Identification

The extracted feature vectors from frames $t$ and $t-1$ are important for the object association and re-identification. Based on these feature vectors and the positions, the algorithm matches and recognizes the objects

### 5.1.2 Framework in Training Phase

Two neural network architectures, YOLO V4 and features extractor, must be trained for this framework. Using P-DESTRE dataset, we trained these two architectures independently, using the $10-$fold learning/validation/test splits provided in the dataset web page. The data is randomly divided into 60% for learning, 20% for validation, and 20% for testing.

- YOLO V4: YOLO V4 is a pre-trained architecture that can be used to detect objects in images and videos. This architecture has been fine-tuned for each fold to detect only pedestrians using the train set. The detector performs well in terms of accuracy and speed, making it an ideal choice for pedestrian detection applications. Furthermore, YOLO V4 is also capable of detecting multiple objects simultaneously, allowing for efficient object recognition tasks. For each fold, we fine-tuned YOLO V4 pre-trained architecture to detect only pedestrian using the train set. The detector performance is monitored using the error on the validation set.

- Feature extractor: the feature extractor architecture is composed of a base model and a header. We used Wide ReseNet-50 (WRN) [49] as a base model, and header composed of two Fully connected Pooling Layers (FPL) of sizes $[4096, 128]$ respectively. From each fold a train, validation, and test patches dataset is created from train, validation, and test sets correspondingly. The model is trained using the vMF learning algorithm (algorithm 1). Figure 3 illustrates the training mechanism of the features extractor.

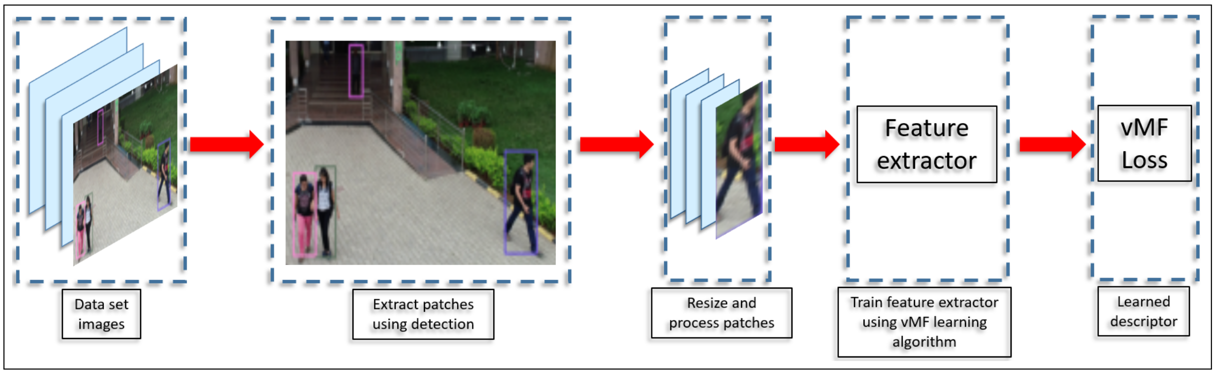

Data set images
Extract patches using detection
Resize and process patches
Feature extractor
Train feature extractor using vMF learning algorithm
vMF Loss
Learned descriptor

Figure 9: Object Visual Representation Learning Framework. First, all patches are extracted from the dataset frames. Second, the patches are resized to the same input shape. Finally, these patches are used by the learning framework to train the feature extractor.

### 5.2 Object Association and Re-identification

Learning a vMF distribution for each ID class has the ultimate objective of predicting the ID of an object instance during inference. These learned distributions, on the other hand, can be used to compare the similarity of any two object instances. As a result, learned distributions can be used to predict ID and assess similarity at the same time. We integrate the feature extractor into the online multiple object tracking and re-identification system once it has been trained in an offline way using the suggested framework. Current detections at frame $t$ are cropped and scaled to fit the feature extractor's input form in this framework. Then, these patches are mapped to the embedding space. Since the system is online, it only retains the previous frame objects and their IDs at $t-1$. The pairwise similarity measure between the current object instances at frame $t$ and the objects instances at frame $t-1$ is the next stage in the system. The object detection association is the final phase.

#### 5.2.1 Similarity Measure

Since the detection patches are mapped to a euclidean space with unit norm $\|\mathbf{x}\|_2$, the similarity can be measured using cosine similarity or Euclidean distance. We opted to utilize both of them since the two metrics are inversely connected and provide the same exact results.

### 5.2.2 Object Detection Association

The final step in the online tracking and re-identification system is the object association and re-identification of the IDs. For object association, the current object detections in frame $t$ needed to be matched with the others in frame $t-1$. However, a pair of matched objects needs to belong to the same identity. In other words, we cannot match two object instances, while they are assigned to two different identities. Therefore, a consistency between the data association and the identity prediction needs to be established.

Prior to the start of the data association step, data preparation is completed. Let $D_t$ denotes the set of object detection in frame $t$, where $d_t^i$ denotes the $i^{th}$ detected object. Let $\mu_c$ denotes the learned mean direction for ID $c$. The data association can be done in two different ways. First, we only rely on ID prediction by assigning the class, of which the mean direction is the nearest to the object representation:

$$\arg\max_c \quad = \quad \cos{(d_t^i, \mu_c)}, c = 1..C, \tag{13}$$

where cos is the cosine similarity function. This technique is simple and straightforward, where no data association algorithm is needed. It can be seen as a recognition problem.

A second way to correlate targets from the previous frame $D_{t-1}$ with detections produced by the current frame $D_t$, we utilize the product result of the two appearance and position criteria to indicate the amount of similarity of the target-detection pair of $i^{th}$ target and $j^{th}$ detection. This product is defined as follows:

$$\arg\max_j = \frac{\cos{(d_t^i, d_{t-1}^j)} + 1}{2} * \frac{Max_d - \|d_t^i - d_{t-1}^j\|_2}{Max_d - Min_d} \tag{14}$$

where $\|d_t^i - d_{t-1}^j\|_2$ is the Euclidean distance between the two object centroides in the pixel space. $Max_d$ and $Min_d$ are the maximum and the minimum distances respectively, between the two sets $D_t$ and $D_{t-1}$. Each of these criteria is normalized to [0..1] range, so that we have a normalized score. This similarity is motivated by the fact that, the similarity between two object instances is not enough; it has to be reinforced by a position confidence score. We suppose that two detections belonging to the same ID have a similarity measure that is higher than a predetermined threshold $T$. The user must define this threshold, which is a hyper-parameter. We assign IDs to the unmatched subset of objects in frame $t$, which is denoted as $D_t^u$, using equation 9.

## 5.3 Experiments

In this section we report the performance results of the proposed framework. In order to provide a comparable results, we follow the same experimental protocol provided by the P-DESTRE web page[1]. The experiments were divided into three categories: pedestrian 1) detection 2) tracking 3) re-identification. We evaluated our framework in a cross-validation manner to measure its accuracy and robustness. We also compared our results with other existing state-of-the-art methods. The results of the state-of-the-art are directly reported from the original paper.

### 5.3.1 Pedestrian Detection

A crucial step in optimizing any machine learning model is hyper-parameter tuning. Particularly with YOLO V4, rigorous hyper-parameter adjustment is necessary for optimum accuracy and performance. Non Maximum Suppression (NMS), confidence threshold, learning rate, and maximum number of positive predictions are some of the important hyper-parameters used during YOLO V4 hyper-parameter tuning. For the fine tuned YOLO V4 detector, we set the NMS hyper-parameter intersection over union between the predictions to 0.5 and the threshold score for pre-filtering to 0.05. The maximum number of positive detections (predicted pedestrian per image) is set at 100. During the inference phase, the score threshold is set to 0.3. To train the detector, 300 epochs with a learning rate of 0.0001 were sufficient. In order to align with the reported results from the original paper, we used the same development kit[2] to evaluate YOLO V4 detection performance in P-DESTRE dataset. Following the $10-$fold cross validation scheme provided[TODO: look at VMF paper], in which each fold the data was randomly divided into 60% for learning, 20% for validation, and 20% for testing.

The results of all detection methods are summarized in Table $II$. The approaches were assessed using the Average Precision (AP) measure with an Intersection over Union (IoU) of 0.5 ($AP@IoU = 0.5$). Using this table, it is possible to assess how each method performs in different settings such as object categories and sizes. Furthermore, by observing the AP score of all these algorithms, one can better understand which model may provide the best accuracy in terms of localizing objects with higher degree of overlap between ground truth and predicted values. As can be observed, the YOLO V4 outperformed the other detection methods considerably. It's also built for online real-time detection, making it simple to combine with other online tracking and re-identification systems as a detector.

[1] http://p-destre.di.ubi.pt/experiments.html
[2] http://host.robots.ox.ac.uk/pascal/VOC/voc2012/devkit

| Method | Backbone | AP |
|---|---|---|
| RetinaNet[29] | ReseNet-50 | 63.10 % ± 1.64 % |
| R-FCN[13] | ReseNet-101 | 59.29 % ± 1.31 % |
| SSD[30] | Inception-V2 | 55.63 % ± 2.93 % |
| YOLO V4[7] | CSPResNext50 | 65.70 % ± 2.40 % |

Table 2: The Average Precision (AP) results obtained by 4 detection methods in the P-DESTRE dataset.

We overall qualitatively analyzed the performance of YOLO V4 and other approaches presented in the P-DESTRE paper on crowded scenes. By changing various parameters and additional features, we examined how well each approach could detect objects in challenging scenarios, such as when a significant part of the scene was occluded.

Overall, our evaluation showed that all approaches faced difficulties with crowded situations and partial viewability. The results indicated that the performance of all the methods was affected by occlusions and small amounts of area visible in the scene.

### 5.3.2 Pedestrian Tracking

As pedestrian tracking experiment settings are important to optimize the performance of tracking methods. Tracking experiment settings can be used to find optimal parameters for pedestrian tracking and test the corresponding implications. Through these experiments, we can observe key performance indicators such as speed, trajectory accuracy, detection failures, etc., to properly optimize the algorithm parameters related to pedestrian tracking. With proper optimization and suitable parameter settings, we can gain maximum accuracy in pedestrian movements. Therefore, in our case empirical experiments are used to find the best hyper-parameters for the tracking algorithm. We trained the feature extractor, using the $10-$fold cross validation scheme of the P-DESTRE set. In this scheme, each split randomly divided into 60% for learning, 20% for validation and 20% for test, i.e., 45 videos were used for learning, 15 for validation and 15 videos for test purposes. From each frame, the bounding boxes are cropped and scaled to patches of dimension $(48, 64, 3)$. The input patch is normalized to the $[0, 1]$ range by dividing by 255 for the prepossessing. We set the concentration parameter $\kappa$ to 15 for the learning algorithm hyper-parameter. This number produced the best outcomes experimentally. The feature extractor was trained using 50 epochs with a batch size of 64. 128 is the embedding space dimension. The similarity threshold is set to 0.2.

As detailed in [4], we assessed the performance of our proposed tracking system using three metrics: Multiple Object Tracking Accuracy (MOTA), Multiple Object Tracking Precision (MOTP), and F1 score. Table $III$ summarizes the performance results. As it can be seen, our proposed tracking method slightly outperformed the other tracking methods in terms of MOTA, but provides comparable results in terms of MOTP to TracktorCV. In terms of $F-1$ score, our method is significantly better.

From a qualitative standpoint, our approaches struggle with severe occlusions. Normally, it is difficult to reestablish the person ID in online tracking after occlusion happens, but our method is able to restore the person ID in several cases due to re-identification based on vMF mean direction (9).

| Method | MOTA | MOTP | F-1 |
|---|---|---|---|
| TracktorCv[3] | $56.00\,\% \pm 3.70\,\%$ | $55.90\,\% \pm 2.60\,\%$ | $87.40\,\% \pm 2.00\,\%$ |
| V-IOU[6] | $47.90\,\% \pm 5.10\,\%$ | $51.10\,\% \pm 5.80\,\%$ | $83.30\,\% \pm 8.40\,\%$ |
| IOU[5] | $38.27\,\% \pm 8.42\,\%$ | $39.68\,\% \pm 4.92\,\%$ | $74.29\,\% \pm 6.87\,\%$ |
| Ours (based on vMF) | $59.30\,\% \pm 3.50\,\%$ | $56.10\,\% \pm 5.61\,\%$ | $92.10\,\% \pm 3.20\,\%$ |

Table 3: Comparison between 3 tracking algorithms from the state-of-the-art and our tracking algorithm in the P-DESTRE dataset.

We designed an experiment to display the tracking results in inference mode on a chosen video as a proof-of-concept for our tracking plus detection approach. We picked one of the $5-$folds at random, we trained the algorithm, and then apply it to a particular video from the test set in inference mode. The results of the detection and tracking at three distinct timestamps are shown in Figure 4. As can be noticed, the tracking algorithm was successful in keeping track of the majority of the pedestrians in the scenario. Our investigation revealed that the majority of switched identities occur as a result of miss-detection at particular frames, which results in two cases: 1) assign new ID one the pedestrian when it is detected again, 2) two pedestrian switch their identities.

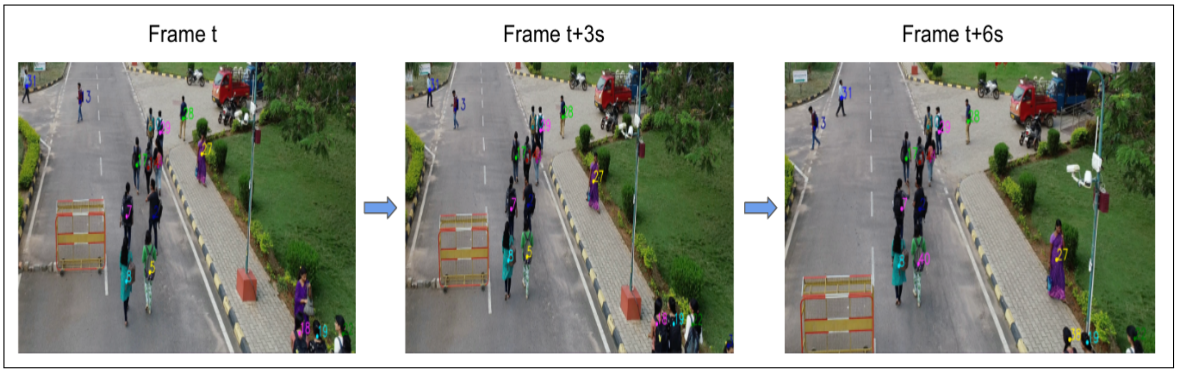


Figure 10: Example of applying our detection plus tracking in inference mode on one of the videos. The selected video is part of the test data. We show the results at three different timestamps. The tracking algorithm was able to keep track of most of the identities in the scene.

### 5.3.3 Long-Term Pedestrian Re-Identification

Long-term pedestrian re-identification is one of the main reasons for the development of tracking and re-identification based on vMF distribution. In a long-term real world scenario pedestrian clothing appears to differ between lapses of time (e.g.,. days/weeks). Therefore, a reliable identifier must rely on alternative features rather than clothes appearances such as face or body features.

In this experiment, the data was randomly divided into 5−folds. The ratios of these folds are 50% learning, 10% gallery, and 40% query. The detailed information about this split is provided in[TODO: look at VMF paper].

Our approach to re-identification of persons can be assessed in two ways. First, we utilize equation (9) to get the nearest mean direction of the IDs. Second, top-N recall performance can be used. We report using the same metrics as the state-of-the-art results in order to be consistent. We also present the results of the re-identification using the initial measure we presented (9). The results are summarized in table $IV$. Our method significantly outperformed other methods under different metrics. This proves that a feature extractor based on vMF is able to learn reliable features that help in recognizing the person, rather than the person's clothes appearances.

| Method | mAP | Rank-1 | Rank-20 | Mean direction |
|---|---|---|---|---|
| ArcFace[14] + COSAM[41] | 34.90 % ± 6.43 % | 49.88 % ± 8.01 % | 70.10 % ± 11.25 % | — |
| vMF identifier | 37.85 % ± 3.42 % | 53.81 % ± 4.50 % | 74.61 % ± 8.50 % | 64.45 % ± 3.90 % |

Table 4: COMPARISON BETWEEN THE RE-IDENTIFICATION PERFORMANCE ATTAINED BY the STATE-OF-THE-ART METHODS and ours based on vMF IN THE P-DESTRE

#### 5.3.4 Analysis

In order to demonstrate the capability of the proposed framework to learn a von Mises-Fisher (vMF) distribution for each pedestrian identity, a visual representation of the learned embeddings was created through an experiment. To conduct the experiment, 7 pedestrian identities were randomly selected, and the vMF learning algorithm was applied to their patch dataset. The embedding dimensions were set to 3 for visualization purposes and the concentration parameter $\kappa = 15$ was set to 15. The results of the experiment, as illustrated in figure 11, revealed the success of the algorithm in learning vMF distributions for each identity. The patches belonging to the same identity were grouped together into a single cluster, which was clearly distinguishable from the embeddings belonging to other identities. This supports the proof of concept that the proposed framework can effectively learn a vMF distribution for each pedestrian identity.

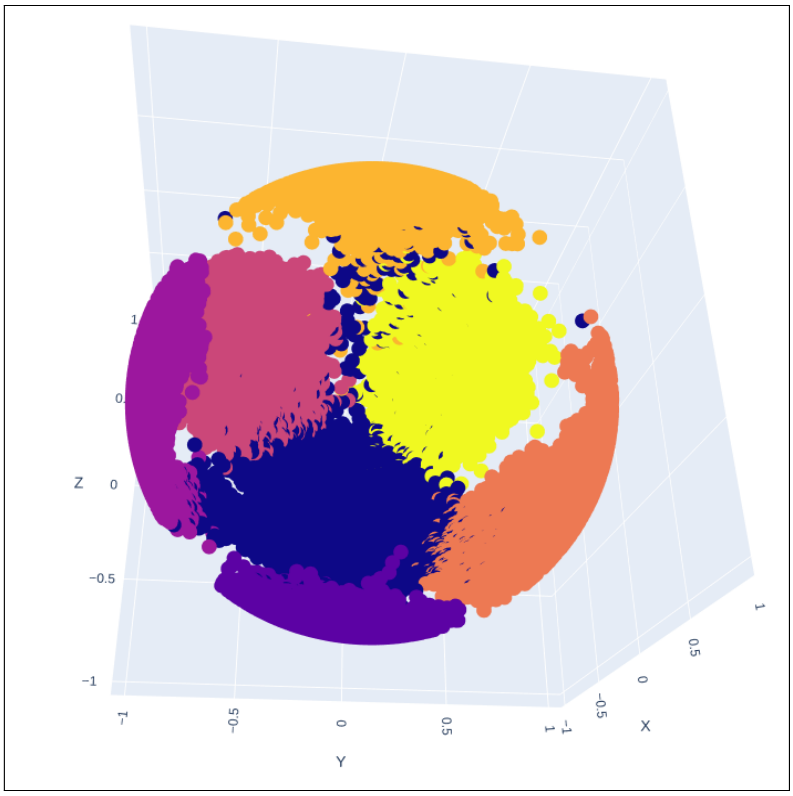


Figure 11: The learned embedding of 7 randomly selected pedestrian identities, where each color is associated with one single identity. For the purpose of visualization we set the embedding dimensions to 3. We also set the concentration $\kappa = 15$.

In order to understand the impact of the concentration parameter ($\kappa$) on the learned embedding space distribution, an experiment was conducted, where the only variable manipulated was the concentration value. The experiment followed the same procedure outlined previously. The results were visualized in Figure 12, which compares two concentration values: a low value of $\kappa = 1$ and a high value of $\kappa = 20$. The results of the experiment indicated that a high concentration value aided in aggregating data points with the same identity towards the mean. However, this also resulted in the model learning von Mises-Fisher (vMF) distributions that were closely grouped together. Conversely, with a low concentration value, the data points belonging to the same identity were less tightly clustered, while the different identities were more separated in the embedding space. This highlights the trade-off between clustering data points within identities and separating different identities in the embedding space.

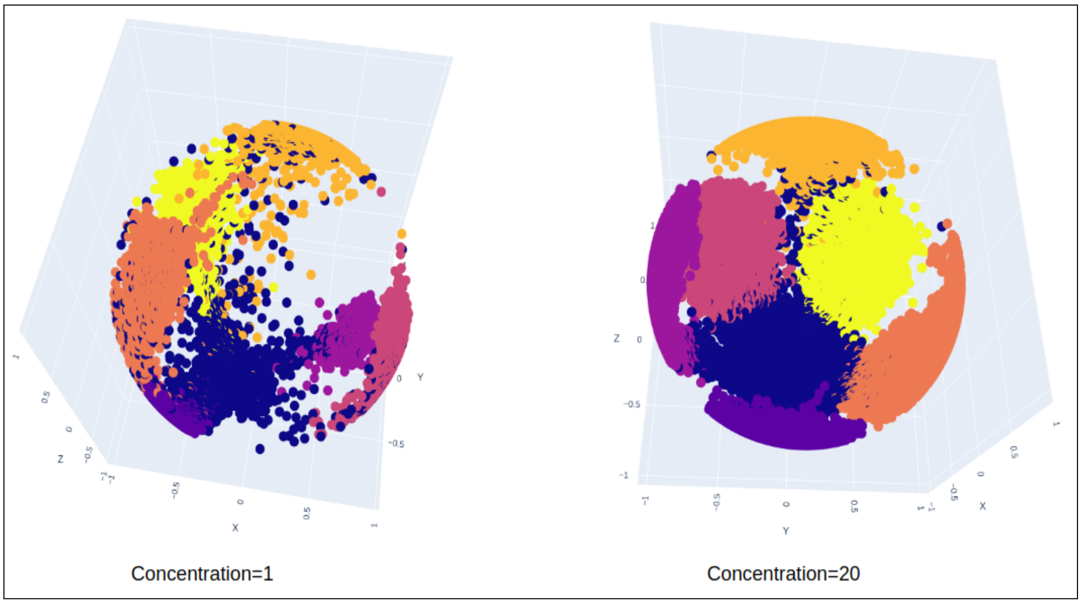


Figure 12: An illustration of the effect of concentration $\kappa$ on the learnt embedding space distribution. The learned embedding of 7 randomly selected pedestrian identities, where each color is associated with one single identity. For the purpose of visualization we set the embedding dimensions to 3. a) low $\kappa = 1$ and b) high $\kappa = 20$. High concentration moves data points that belongs to the same identity towards, but identities are grouped close together, while with low concentration data points that belongs to the same identity are less grouped while the identities are projected far from each other.

# CHAPTER 6

# OUT OD DISTRIBUTION DETECTION METHODOLOGY

In a previous chapter, we introduced a von Mises-Fisher (vMF) based approach to train a neural network for pedestrian re-identification. This approach represents each identity as a compact vMF distribution on the unit sphere, which is distinct from the distributions of other identities. However, this approach has a limitation in that it only recognizes identities present in the training data. In real-world scenarios, new pedestrians or objects may appear that are not in the training data, and the model should be able to detect them as Out-of-Distribution (OOD) samples.

Recently, a framework called Virtual Outlier Synthesis (VOS) was proposed by [16], which detects OOD data points in an online manner by synthesizing virtual outliers in the embedding space. This framework is based on the assumption that samples from the same object should map to a compact space, which aligns with the vMF-based approach.

In this chapter, we revisit the vMF approach and integrated it with VOS to detect OOD data points. Our approach merges the two frameworks and is presented in detail. We conducted several experiments to evaluate the performance of our proposed framework. The results showed that our framework was able to detect new pedestrians not present in the training data during inference. Additionally, it slightly improved the re-identification performance.

In figure 13, a representation of what Convolutional Neural Networks (CNNs) have learned in the embedding space is depicted. The figure consists of two subfigures, (a) and (b), each showing a different aspect of the learned embeddings.

(a) illustrates how deep learning methods can distinguish between 3 classes. The figure shows that the learned decision boundary is not tight around the compact embedding of each class region. This means that any Out of Distribution (OOD) object embedding that is far from the decision boundary will likely be predicted as one of the 3 classes with a high score assigned. In other words, the model may misclassify OOD objects as in-distribution objects.

(b) shows the OOD region in the embedding space, which the model should detect. This region corresponds to embeddings that are not associated with any of the 3 classes and, therefore, represent new objects or instances that the model has not seen before. It is crucial for deep learning models to be able to detect such OOD instances, especially in real-world scenarios where new objects and instances are likely to appear. The ability to detect OOD instances can help the model to generalize better and avoid making incorrect predictions.

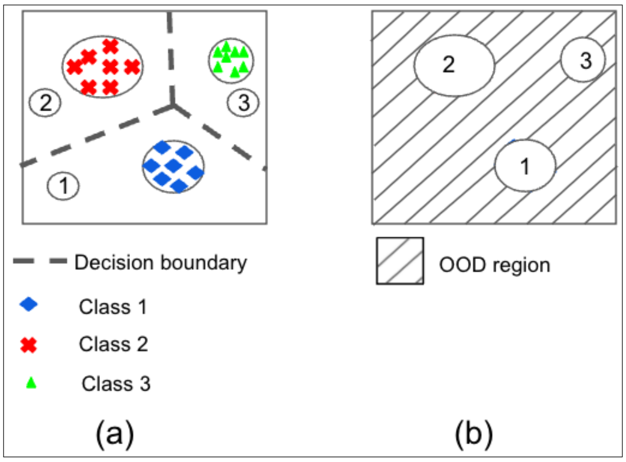


Figure 13: This figure show a representation of what CNNs learns in the embedding space. (a) illustrates how deep learning methods learns to distinguish between 3 classes. The learned decision boundary is not tight around the compact embedding of each class region. Any OOD object embedding far from the decision boundary, will be predicted as one of the 3 classes with a high score assigned. (b) shows the OOD region in the embedding space which the model should detected.

### 6.1 Out of Distribution Pedestrian based on Von-Mises Fisher Distribution

In real-world scenarios, particularly in security environments, the presence of new, unseen pedestrians is highly likely. As a result, the model must be equipped to detect these pedestrians as Out-of-Distribution (OOD) entities.

The scoring functions used to distinguish between ID and OOD entities in the previous method mainly rely on the compact embedding of the objects from the ID set. These functions encourage the Convolutional Neural Networks (CNNs) to embed each class in a tightly clustered region.

Considering the benefits of the compact embedding of ID entities and the importance of detecting OOD entities, the combination of vMF-based models and OOD scoring functions is a promising direction for further exploration. In light of this, we propose a framework that integrates both approaches.

### 6.1.1 Learning

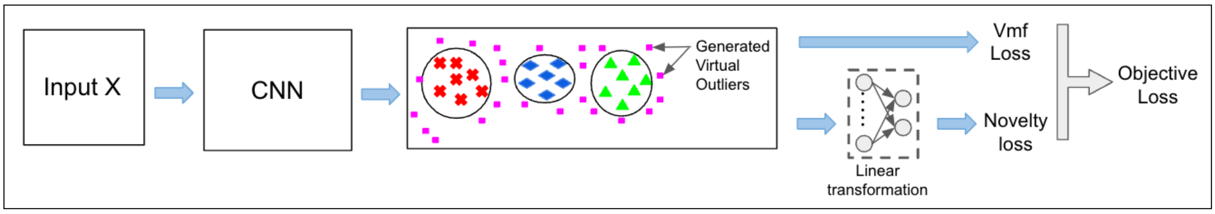


Figure 14: Pedestrian re-identification and novelty detection framework. Inputs is mapped using the CNN backbone. Then, virtual outliers are generated using VOS framework. vMF loss is computed using only the features of the inputs. Novelty loss is computed over the generated virtual outlier features and the the inputs features.

We propose an end-to-end learning framework for pedestrian re-identification and novelty detection. The training procedure of this framework is as simple as the traditional training with soft-max loss. It consists on three main components. A representational visual features based on CNN.In order to detect OOD pedestrians, we adopt the VOS method [TO ADD]. The VOS uses the class-conditional multi-variate distribution to generate virtual outliers. We think sampling from the embedding space is can help not only detect novelty but also can help build a robust model. By sampling hard virtual outlier, we assume that the score function will help learn more compact embedding for each ID at the same detect OOD pedestrians as non-semantic shifts.

Once the hard virtual outliers are generated, two parallel heads are computed. The first head computes the vMF loss using the embeddings of the ID. The second, head computes the novelty loss over the binary output of the linear transformation. This loss aims to distinguish between virtual outlier and ID embeddings. The objective loss is then computed by a weighted sum of the two losses.

In the VOS framework they uncertainty loss (novelty loss) is defined with the binary sigmoid loss.

$$\mathcal{L}_{Novelty} = -\log \frac{1}{1 + exp^{-\Theta_u . \mathbb{E}(v,\Theta)}} + -\log \frac{exp^{-\Theta_u . \mathbb{E}(x,\Theta)}}{1 + exp^{-\Theta_u . \mathbb{E}(x,\Theta)}}. \tag{15}$$

Where, $\Theta_u$ represents the weights of the classification head for novelty detection. $\mathbb{E}(.)$ is the energy score function. $\Theta$ is the weights of the CNNs base encoder.

To train this framework in an end-to-end manner, we combine the two losses to form one objective training loss.

$$L_{loss} \quad = \quad L_{vmf} + \gamma * L_{Novelty}. \tag{16}$$

This is a weighted loss, where $\gamma$ is the weighted of the uncertainty loss.

```
Algorithm 2 Virtial Outlier Synthesis learning algorithm
Input: ID data $D_{in}$, queue size $|Q|$ for Gaussian density estimation, $\gamma$ weight for uncertainty regularization, and $\epsilon$.
Output: pedestrian re-identification parameterized by $\Theta$, and novelty detector $\mathcal{S}$ parameterized by $\Theta_u$
While train do:
  1. Initialize CNN parameters $\Theta$.
  2. Repeat:
     (a) Estimate mean directions using (8) and all the training data.
     (b) For several iteration:
         i. update the ID queue $Q_k$ with the embeds of training inputs.
         ii. Estimate the multi-variate distribution using the $Q_k$.
         iii. sample virtual outlier
         iv. compute the objective loss using Equation 16.
     (c) Train CNN for several iterations and update $\Theta$.
  3. Until convergence.
```

### 6.1.2 Inference

The pedestrian re-identification and out-of-distribution detection are the two main objectives of the proposed framework. The inference of the proposed framework can be divided into two main parts: the out-of-distribution (OOD) scoring function $\mathcal{S}$ and the von Mises-Fisher (vMF) framework.

The first part of the inference involves the use of the OOD scoring function $\mathcal{S}$ to decide whether the input pedestrian is OOD. The decision can be made by selecting a threshold $T$ based on the experimental settings. The threshold can be used to determine if the value of $\mathcal{S}(x)$ is greater than $T$ or not. If the value of $\mathcal{S}(x)$ is greater than $T$, the input pedestrian is considered to be in-distribution (ID), otherwise, it is considered to be OOD.

$$\begin{cases} ID, & \text{if } \mathcal{S}(x) > T \\ OOD, & \text{otherwise} \end{cases}. \tag{17}$$

In the second part of the inference, in case the input pedestrian is ID, the proposed framework uses the vMF framework to re-identify the pedestrian. The re-identification can be achieved by measuring the similarity between the embedding of the input in the unit sphere and the learned mean direction ${\mu_i}_{i=1}^N$. The input pedestrian can be assigned the identity with the highest cosine similarity to its embedding.

### 6.2 Experiments

In this section, we present a comprehensive summary of the experiments conducted to assess the performance of our proposed framework. We followed a standard procedure in all experiments, which involved training the proposed framework on the training set of the in-distribution (ID) dataset, denoted as $D_{in}$. The evaluation was performed on the union of two sets: the validation set of the ID, denoted as $D_in^val$, and the out-of-distribution (OOD) dataset, denoted as $D_{out}$. It is crucial to note that the ID and OOD datasets should not have any overlapping person identities.

The setting of the ID and OOD datasets can be performed in two distinct ways. Firstly, two separate datasets with no common person identities can be selected and set as the ID and OOD datasets, respectively. Secondly, the same dataset can be divided into ID and OOD by splitting each set into ID and OOD with a predetermined ratio, for the training, validation, and testing sets.

It is important to emphasize that during both the training and validation phases, the framework does not have access to the OOD dataset. The performance of the model was monitored based on the validation set of the ID and the online generated virtual outliers. This involved selecting the appropriate model checkpoint and optimizing the hyper-parameters of the framework. During the testing phase, we used both the test set of $D_{in}$ and $D_{out}$ to evaluate the model's long-term re-identification and OOD detection capabilities.

#### 6.2.1 Two Different Datasets setting

##### 6.2.1.1 ID Dataset

The P-DESTRE dataset was utilized as the identity (ID) data in the pedestrian re-identification experiments. A thorough description of the P-DESTRE dataset can be found in Section III. To perform the experiments, the data was randomly divided into 5 folds, with each fold consisting of a 50% learning set, a 10% gallery set, and a 40% query set. This division of data into folds helps to ensure a fair evaluation of the models and provides a comprehensive understanding of the performance of the models across different data splits. Additionally, this also helps to mitigate the risk of overfitting and provides a more robust evaluation of the models, given that the models are tested on unseen data. By using a randomly divided dataset, the results of the experiments are more representative of the models' general performance and can be better compared to other models and their results. The detailed information about this split is provided in [TODO: look at VMF paper].

#### 6.2.1.2 OOD Dataset

As OOD, we used **CUHK03-NP**, which is a widely used dataset in the field of pedestrian re-identification. It contains 14,097 images of 1,467 identities [28]. All the splits were used in the evaluation as an $D_{out}$. The CUHK03-NP dataset is a commonly used benchmark dataset in the field of computer vision and machine learning, particularly for evaluating person re-identification algorithms. The dataset was collected from two camera views at the Chinese University of Hong Kong and consists of 1467 identities captured in a multi-shot manner. Each identity is captured with both color and depth images, providing a rich source of information for developing and testing algorithms. The dataset has become a popular choice for researchers due to its large scale and the presence of challenging conditions, such as occlusions and viewpoint changes, which pose difficulties for re-identification algorithms. The CUHK03-NP dataset has been used in many recent studies, demonstrating its utility as a benchmark for evaluating the performance of re-identification algorithms under various conditions.

The CUHK03-NP dataset is a popular dataset used for pedestrian re-identification. It was published in 2013 and contains 1,360 identities with a total of 14,096 images. The images in this dataset have a resolution of 300 x 300 pixels and are taken from multiple cameras. The annotations in this dataset were manually labelled, making it a high-quality dataset for use in research and development of pedestrian re-identification algorithms.

| Property | Value |
|---|---|
| Dataset Name | CUHK03-NP |
| Number of Images | 14,097 |
| Number of Identities | 1,467 |
| Resolution | 640x480 |
| Annotations | Pedestrian bounding boxes |
| Source | CUHK Person Re-Identification Dataset |

Table 5: Properties of the CUHK03-NP Dataset

#### 6.2.1.3 Training Details

From each image, the bounding boxes are cropped and scaled to patches of dimension $(48, 64, 3)$. For the prepossessing, the input patches are normalized using the mean and standard deviation learned from imagenet dataset. The feature extractor architecture is made up of two parts: a base model and a header. As a base model, we used Wide ReseNet-50 (WRN) [49], with a header consists of two Fully connected Pooling Layers (FPL) of sizes $[4096, 128]$. The feature extractor was trained using 50 epochs with a batch size of 64. 128 is the embedding space dimension. The used optimizer is Adam with a learning rate starts with 0.2 and decreases by a factor of 0.5 every 25 of the training epochs. We set the concentration parameter $\kappa$ to 15 for the learning algorithm hyper-parameter. This number produced the best outcomes experimentally. We update the mean directions after every epoch.

For hyper-parameters related to VOS, we used 500 samples per identity to estimate the the class-conditional Gaussians. WE set $\epsilon = 0.0001$. We sampled 1000 virtual outliers from the embedding space. We also set $\gamma = 0.1$. The linear transformation consists of two layers $[269, 1]$. The first layer has a number of nodes equal to the number of identities so that the energy based can be computed per identity as designed in VOS.

#### 6.2.1.4 Two Datasets Settings

In evaluating the framework's capability in detecting Out-of-Distribution (OOD) samples, we employed the Area Under the Curve (AUC) based on Precision/Recall curve metric. This metric was chosen for a number of compelling reasons. Firstly, it is a more accurate metric when dealing with imbalanced data, which is the case in the ratio of Identities (ID) vs OOD samples in the open world scenario. Secondly, it is preferred when the positive class, in this case ID, is of utmost importance. The results of the framework evaluation based on AUC-PR are summarized in Table I. As can be observed, the model demonstrates a commendable performance in terms of detecting OOD samples.

The results in this table summarize the performance of a pedestrian tracking and re-identification framework based on the Area Under the Curve based on Precision/Recall curve (AUC-PR) metric. The framework was tested using two datasets, the ID dataset P-DESTRE and the OOD dataset CUHK03-NP. The results show that the framework had a performance of 63.10% ± 1.64% AUC-PR using the Wide Residual Network (WRN) as its backbone. This suggests that the framework performed well in detecting ODD samples, however, the results should be interpreted with caution as the standard deviation of the results is not provided, which can give an indication of the variance in the performance of the model. Additionally, it would be beneficial to compare the performance with other existing methods to put the results into perspective.

| ID Dataset | OOD Dataset | Backbone | AUC-PR |
|---|---|---|---|
| **P-DESTRE** | **CUHK03-NP** | WRN | 63.10 % ± 1.64 % |

Table 6: The Area Under the Curve (AUC) based Precision/Recall curve (PR) results obtained by applying our framework on the two different datasets setting. The ID is **P-DESTRE**, and the OOD is **CUHK03-NP**.

### 6.2.2 One Dataset setting

Another way to evaluate the framework is on the same pedestrian dataset but divided into ID and OOD by identities. Detecting OOD data points in a different dataset setting is expected to be easier comparing to the one dataset setting regarding the image acquisition setups. The differences between the two image acquisitions such as the type of cameras, the angel, and the lightening can play a role in obtaining a separable embedding between ID and OOD. Although, it is important to detect OOD from a different setup, it also important to test the model on ID and OOD from the same image acquisition setup. We believe that the one dataset setup is more likely to happen in the open world environment. In addition, to detect the non-semantic shifts (new pedestrians) in the same setup is more challenging and the model has to rely on features such as face and body features rather than clothing appearances.

Since the **P-DESTRE** dataset is randomly divided into $5-$folds, we divided each fold to two dataset, ID and OOD based on the identities. The division ration is 70/30% for ID and OOD respectively. It worth mentioning that this division is preformed for each fold.

For the training details, everything almost the same as in the two different dataset setting except we lowered the value $\epsilon$ to 0.001. This can be explained by the fact that the distinguishing between ID and OOD in this setting is more challenging and it required harder virtual outlier to learn a better score function $\mathcal{S}$.

We evaluate the model the same way we evaluated in the two different dataset setting. Table $II$ summarises the results, we clearly notice that the performance drops, which confirms our assumptions about the challenges of the one dataset setting.

The results presented in the table show the performance of a framework using the WRN backbone on the ID dataset of the P-DESTRE dataset. The performance is evaluated using the Area Under the Curve based on Precision-Recall (AUC-PR) metric and the results are reported as the mean ± standard deviation over multiple runs. The results show that the framework has an AUC-PR of 55.19% ± 3.02%. This indicates that the framework has moderate performance in identifying instances of the positive class (ID) in the P-DESTRE dataset, with some variance in performance between runs.

|  | Dataset | Backbone | AUC-PR |
|---|---|---|---|
| ID dataset | **P-DESTRE** | WRN | $55.19\,\% \pm 3.02\,\%$ |

Table 7: The Area Under the Curve (AUC) based Precision/Recall curve (PR) results obtained by applying our framework on the two different datasets setting. The ID is **P-DESTRE**, and the OOD is **CUHK03-NP**.

### 6.2.3 Long-Term Pedestrian Re-Identification

The development of pedestrian re-identification based on von Mises-Fisher (vMF) distribution has been motivated by the challenge of long-term re-identification in real-world scenarios, where clothing appearance can differ between time gaps. In order to tackle this issue, our approach relies on identifying attributes other than clothing appearance, such as facial or body characteristics.

The effectiveness of our re-identification method can be evaluated using two methods. Firstly, we calculate the nearest mean direction of the IDs using equation (9). Secondly, we assess the top-N recall performance using commonly used metrics in the field. The results, summarized in Table *III*, demonstrate a significant improvement over other state-of-the-art methods in different metrics. This confirms that the vMF-based feature extractor is able to learn robust features that help in recognizing a person, rather than their clothing appearance. Furthermore, our results show that the integration of the vMF method with the VOS framework leads to even better re-identification performance, as it creates a synergy that pushes the embedding of each identity to be more compact and distinct from other identities and out-of-distribution samples. The comparison between the method ArcFace with COSAM and the proposed vMF identifier with and without the VOS framework was performed in terms of Mean Average Precision (mAP), Rank-1 accuracy, Rank-20 accuracy and Mean Direction. The results shown in the table indicate that the vMF identifier, both with and without VOS, significantly outperforms the ArcFace with COSAM method. With a mAP of 40.85% ± 3.42%, the vMF identifier achieved a higher performance compared to the ArcFace with COSAM which had a mAP of 34.90% ± 6.43%. Similarly, the Rank-1 accuracy and Rank-20 accuracy were higher for the vMF identifier with values of 63.81% ± 4.50% and 88.61% ± 8.50%, respectively, compared to the 49.88% ± 8.01% and 70.10% ± 11.25% achieved by the ArcFace with COSAM. Furthermore, the vMF identifier also outperformed the ArcFace with COSAM in terms of the Mean Direction with a value of 64.45% ± 3.90%. These results demonstrate the effectiveness of the proposed vMF identifier in pedestrian re-identification.

| Method | mAP | Rank-1 | Rank-20 | Mean direction |
|---|---|---|---|---|
| ArcFace[14] + COSAM[41] | 34.90 % ± 6.43 % | 49.88 % ± 8.01 % | 70.10 % ± 11.25 % | — |
| vMF identifier without VOS | 37.85 % ± 3.42 % | 53.81 % ± 4.50 % | 74.61 % ± 8.50 % | 64.45 % ± 3.90 % |
| vMF identifier with VOS | 39.15 % ± 2.41 % | 56.18 % ± 3.20 % | 78.59 % ± 7.30 % | 66.5 % ± 2.9 % |

Table 8: COMPARISON BETWEEN THE RE-IDENTIFICATION PERFORMANCE ATTAINED BY the STATE-OF-THE-ART METHODS AND OURS BASED ON vMF IN THE P-DESTRE

#### 6.2.4 Analysis

Figure 15 presents a binary confusion matrix that delineates instances of ID and OOD, whereby the model prediction is compared to the ground truth. In delving deeper into examples of wrong predictions, we observed that these instances were frequently mapped in close proximity to the decision boundary where low energy function is learned.

In the case of the example where OOD was predicted as ID, we noted that when an individual's image was captured from the back, it was often predicted as ID. Conversely, when the same individual's image was captured from the front, the model predicted it as ID with greater accuracy.

In the instance where ID was predicted as OOD, our analysis suggested that this was a limitation of the trained model. Further improvements in the model are needed to better distinguish between similar-looking identities.

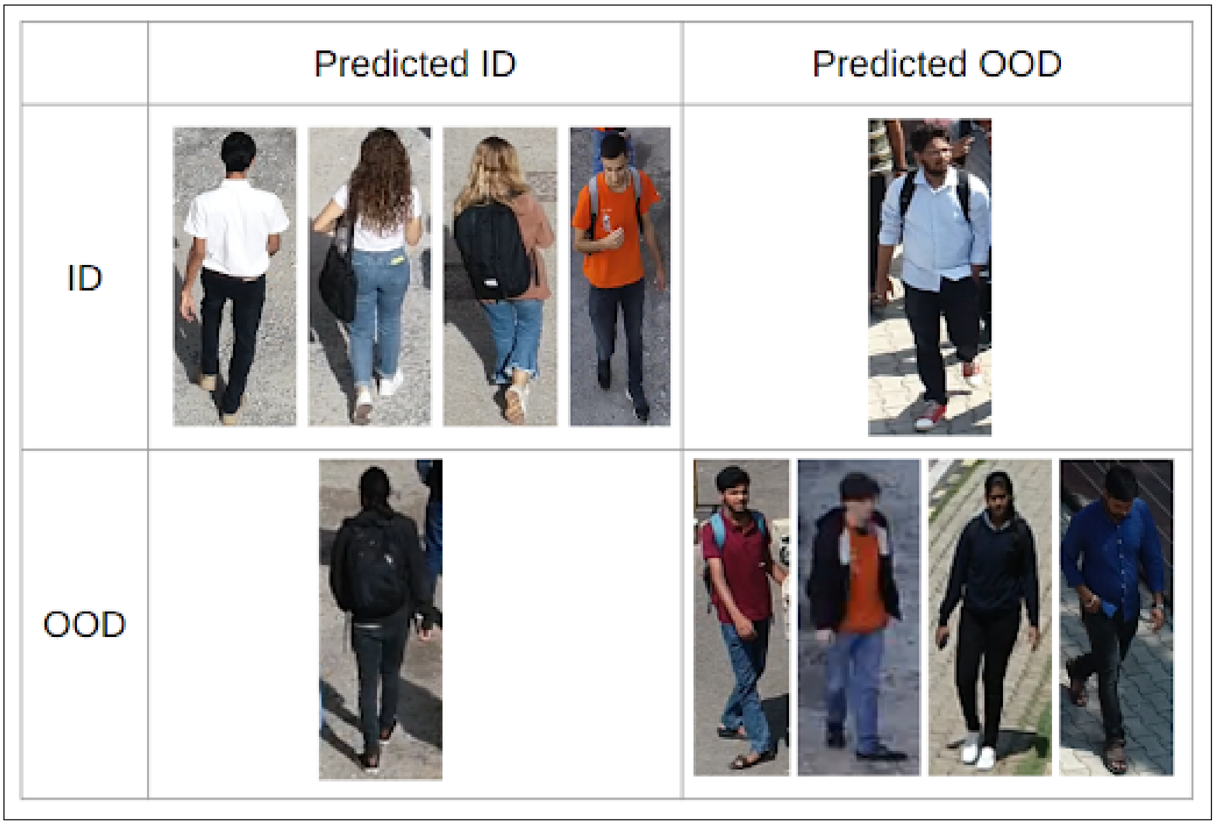


Figure 15: We presented a binary confusion matrix illustrating instances of ID and OOD, where the ground truth was compared against the model prediction.

### 6.3 Train and Inference Time

The proposed approach is aimed at integrating into larger frameworks that operate in real-time, such as security systems. One crucial aspect of the evaluation of the proposed framework is the amount of time required for both training and inference. As anticipated, the addition of the VOS framework resulted in increased training time, due to the memory requirements for storing the coreset and the increased computational requirements. In terms of inference time, a slight increase was observed, as we only added a linear transformation to predict OOD samples. The trade-off between the improved performance and the increased time requirements will need to be carefully considered when implementing this approach in practical applications.

The comparison of the training and inference time of the two methods, vMF identifier without VOS and vMF identifier with VOS, indicates that the inclusion of the VOS framework has a significant impact on the training time, increasing it by 1.4 hours. However, the inference time only increased slightly, by a difference of 0.0002 seconds. This suggests that the overhead of using the VOS framework in terms of time complexity is mainly during the training phase and not during the inference phase. It is important to note that this evaluation is based on a single input (patch), and the time complexity may change when considering larger datasets or more complex models.

| Method | Train | Inference on a single input (patch) |
|---|---|---|
| vMF identifier without VOS | 2 hours | 0.001 s |
| vMF identifier with VOS | 3.4 hours | 0.0013 s |

Table 9: comparison between training and inference time needed between vMF with and without VOS on P-DESTRE dataset.

# CHAPTER 7
# LIMITATION AND FUTURE WORK

## 7.1 Limitations

Pedestrian tracking and re-identification using deep learning is a rapidly growing field that has garnered significant attention in recent years. Despite the advances in this area, there are still several limitations that need to be addressed for the development of more efficient and effective algorithms.

One of the major limitations is the limited size and diversity of datasets. Most existing datasets are small and do not represent a wide range of environmental conditions, pedestrian characteristics, and camera viewpoints. This results in limited generalization capabilities of the algorithms, making it difficult to apply them to real-world scenarios.

Another limitation is the sensitivity of deep learning algorithms to occlusions. Occlusions occur when a part of the pedestrian is obscured from view, which makes it challenging for the algorithms to track and re-identify the individual. Additionally, deep learning algorithms are often computationally intensive, making it challenging to implement them in real-time scenarios.

Moreover, deep learning algorithms are highly dependent on the quality of the training data. If the training data is not representative of the test data, the performance of the algorithms may deteriorate significantly. This highlights the importance of collecting large and diverse datasets that can be used to train the algorithms.

Training Convolutional Neural Network (CNN) models for pedestrian tracking and re-identification is a complex and time-consuming task. The process involves feeding large amounts of data into the model, which takes significant computational power and time. The process requires the use of specialized hardware, such as GPUs, and can take several hours to several days to complete, depending on the size of the data set and the complexity of the model. Furthermore, the training process must be repeated multiple times to fine-tune the model, further adding to the cost in terms of time. In addition, the process also requires large amounts of memory to store the model and its parameters, which can increase the costs associated with training. Overall, the time and computational demands of training CNN models for pedestrian tracking and re-identification make it an expensive and challenging task for researchers and practitioners alike.

In conclusion, while deep learning algorithms have shown promising results in pedestrian tracking and re-identification, there are still several limitations that need to be addressed. The development of larger and more diverse datasets, as well as improvements in algorithms that can effectively handle occlusions and real-time applications, will be critical for the advancement of this field.

### 7.2 Future Work

The field of pedestrian tracking and re-identification using deep learning is still in its early stages and there is much room for improvement and growth. Despite the recent advances in the field, there are still several limitations and challenges that need to be addressed in order to fully realize the potential of deep learning for pedestrian tracking and re-identification. Some of the future work that needs to be done in the field include:

Improved dataset: One of the biggest challenges facing the field is the lack of large-scale datasets with high-quality annotations for pedestrian tracking and re-identification. This is particularly important for deep learning-based approaches as they rely on large amounts of data to learn robust representations. The development of large-scale datasets with high-quality annotations will be essential to drive future advances in the field.

Real-time processing: Another important challenge facing the field is the need for real-time processing. Most deep learning-based approaches are computationally intensive and require large amounts of processing power. In order to be practical for real-world applications, future work needs to focus on developing more efficient algorithms that can run in real-time.

Robustness to changes in the environment: Pedestrian tracking and re-identification algorithms need to be robust to changes in the environment such as illumination changes, occlusions, and changes in viewpoint. Future work needs to focus on developing algorithms that are more robust to these changes in order to increase their reliability in real-world applications.

Transfer learning: Transfer learning is an important approach that can be used to improve the performance of pedestrian tracking and re-identification algorithms. Future work needs to focus on developing transfer learning algorithms that can be used to transfer knowledge from one dataset to another, thereby increasing their performance and reducing the amount of training data required.

Multi-modal data: Most current algorithms for pedestrian tracking and re-identification rely on visual data. However, there is a growing body of evidence that suggests that incorporating other modalities such as audio, depth, and motion can lead to significant improvements in performance. Future work needs to focus on developing algorithms that can leverage multiple modalities to improve the performance of pedestrian tracking and re-identification algorithms.

In conclusion, there is much future work to be done in the field of pedestrian tracking and re-identification using deep learning. By addressing the current challenges and limitations, the field has the potential to make significant contributions to the development of real-world applications in areas such as public safety, surveillance, and transportation.

# CHAPTER 8
# CONCLUSION

In conclusion, this research proposes an innovative online pedestrian tracking and re-identification approach based on aerial devices, such as drones. The approach leverages the capability of machine learning algorithms to learn the vMF distribution for each individual pedestrian and tracks their movements over time. The neural network encoder, trained using the P-DESTRE dataset, was shown to have exceptional performance in terms of tracking and re-identifying pedestrians.

The proposed approach demonstrates the potential for practical applications in various domains such as public area monitoring and surveillance. Its capability to accurately track and recognize individuals has the potential to greatly enhance the safety and security of society. The results of this study serve as a starting point for further research in the field of pedestrian tracking and re-identification, particularly in the context of aerial devices. The proposed approach's scalability and generalizability to different scenarios can be further tested and validated in future studies. Overall, the proposed approach presents a promising solution for enhancing the reliability and accuracy of pedestrian tracking and re-identification in practical contexts.

The proposed framework was thoroughly evaluated on pedestrian dataset obtained from aerial devices. The results of the experiments demonstrated that the framework demonstrated significant improvements in long-term re-identification compared to previous methods, as well as to a similar approach that lacked the VOS component. The study aimed to detect non-semantic shifts as Out-of-Distribution samples. The results of the experiments further indicated that it is a challenging task to detect non-semantic shifts when the Out-of-Distribution samples come from the same acquisition setup. These findings provide valuable insights into the potential limitations and areas for improvement of the proposed framework in pedestrian tracking and re-identification.

# CHAPTER 9

# ETHICS AND STATEMENT

As researchers in the field of machine learning and computer vision, it is our ethical responsibility to ensure the protection of individual privacy and rights while conducting our work. In applying machine learning algorithms to detect, track, and re-identify pedestrians from UAVs, we acknowledge the potential consequences of our research and we were committed in conducting our work in an ethical and responsible manner.

We ensured that all data collected and used in our research is obtained legally and with the proper consent from individuals. Furthermore, we took necessary steps to protect the privacy and security of the data.

Additionally, we considered the potential impacts of our research on the public, including any potential biases that may exist in our algorithms and the potential for misuse of our technology. We strived to be transparent in our research methodology and findings.

Our research work aims to improve the reliability, security, and safety of machine learning-based pedestrian tracking and re-identification models. Our study has the potential to bring direct benefits to society, particularly in the area of public safety and surveillance. Our research activities are fully compliant with relevant laws and regulations, and we believe that the findings from this study will provide valuable insights into the practical aspects of pedestrian monitoring and re-identification. We expect that this study will help to raise awareness about the importance of ensuring safety in public areas, and will not result in any unintended negative consequences.

# REFERENCES

[1] Tamar Avraham et al. “Learning implicit transfer for person re-identification”. In: *European Conference on Computer Vision*. Springer. 2012, pp. 381–390.

[2] Jerome Berclaz et al. “Multiple object tracking using k-shortest paths optimization”. In: *IEEE transactions on pattern analysis and machine intelligence* 33.9 (2011), pp. 1806–1819.

[3] Philipp Bergmann, Tim Meinhardt, and Laura Leal-Taixe. “Tracking without bells and whistles”. In: *Proceedings of the IEEE/CVF International Conference on Computer Vision*. 2019, pp. 941–951.

[4] Keni Bernardin and Rainer Stiefelhagen. “Evaluating multiple object tracking performance: the clear mot metrics”. In: *EURASIP Journal on Image and Video Processing* 2008 (2008), pp. 1–10.

[5] Erik Bochinski, Volker Eiselein, and Thomas Sikora. “High-speed tracking-by-detection without using image information”. In: *2017 14th IEEE International Conference on Advanced Video and Signal Based Surveillance (AVSS)*. IEEE. 2017, pp. 1–6.

[6] Erik Bochinski, Tobias Senst, and Thomas Sikora. “Extending IOU based multi-object tracking by visual information”. In: *2018 15th IEEE International Conference on Advanced Video and Signal Based Surveillance (AVSS)*. IEEE. 2018, pp. 1–6.

[7] Alexey Bochkovskiy, Chien-Yao Wang, and Hong-Yuan Mark Liao. “Yolov4: Optimal speed and accuracy of object detection”. In: *arXiv preprint arXiv:2004.10934* (2020).

[8] Margherita Bonetto et al. “Privacy in mini-drone based video surveillance”. In: *2015 11th IEEE international conference and workshops on automatic face and gesture recognition (FG)*. Vol. 4. IEEE. 2015, pp. 1–6.

[9] Abdelhamid Bouzid. “Automatic target recognition with deep metric learning.” In: (2020).

[10] Guillem Brasó and Laura Leal-Taixé. “Learning a neural solver for multiple object tracking”. In: *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 2020, pp. 6247–6257.

[11] Fanta Camara et al. “Pedestrian models for autonomous driving part I: low-level models, from sensing to tracking”. In: *IEEE Transactions on Intelligent Transportation Systems* (2020).

[12] Ting Chen et al. “A simple framework for contrastive learning of visual representations”. In: *International conference on machine learning*. PMLR. 2020, pp. 1597–1607.

[13] KJ Dai and YL R-FCN. “Object detection via region-based fully convolutional networks. arxiv preprint”. In: *arXiv preprint arXiv: 1605.06409* (2016).

[14] Jiankang Deng et al. “Arcface: Additive angular margin loss for deep face recognition”. In: *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition.* 2019, pp. 4690–4699.

[15] Mert Dikmen et al. “Pedestrian recognition with a learned metric”. In: *Asian conference on Computer vision.* Springer. 2010, pp. 501–512.

[16] Xuefeng Du et al. “VOS: Learning What You Don’t Know by Virtual Outlier Synthesis”. In: *arXiv preprint arXiv:2202.01197* (2022).

[17] Priya Goyal et al. “Scaling and benchmarking self-supervised visual representation learning”. In: *Proceedings of the IEEE/CVF International Conference on Computer Vision.* 2019, pp. 6391–6400.

[18] Shoudong Han et al. “Mat: Motion-aware multi-object tracking”. In: *arXiv preprint arXiv:2009.04794* (2020).

[19] Md Hasnat et al. “von mises-fisher mixture model-based deep learning: Application to face verification”. In: *arXiv preprint arXiv:1706.04264* (2017).

[20] Martin Hirzer et al. “Person re-identification by descriptive and discriminative classification”. In: *Scandinavian conference on Image analysis.* Springer. 2011, pp. 91–102.

[21] Andrea Hornakova et al. “Lifted disjoint paths with application in multiple object tracking”. In: *International Conference on Machine Learning.* PMLR. 2020, pp. 4364–4375.

[22] Yen-Chang Hsu et al. “Generalized odin: Detecting out-of-distribution image without learning from out-of-distribution data”. In: *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition.* 2020, pp. 10951–10960.

[23] Yi-Fan Jiang et al. “Online pedestrian tracking with multi-stage re-identification”. In: *2017 14th IEEE International Conference on Advanced Video and Signal Based Surveillance (AVSS).* IEEE. 2017, pp. 1–6.

[24] Vijay John, Gwenn Englebienne, and Ben JA Kröse. “Solving Person Re-identification in Non-overlapping Camera using Efficient Gibbs Sampling.” In: *BMVC.* 2013.

[25] Martin Koestinger et al. “Large scale metric learning from equivalence constraints”. In: *2012 IEEE conference on computer vision and pattern recognition.* IEEE. 2012, pp. 2288–2295.

[26] SV Aruna Kumar et al. “The p-destre: A fully annotated dataset for pedestrian detection, tracking, and short/long-term re-identification from aerial devices”. In: *IEEE Transactions on Information Forensics and Security* 16 (2020), pp. 1696–1708.

[27] Ryan Layne, Timothy M Hospedales, and Shaogang Gong. “Investigating open-world person re-identification using a drone”. In: *European conference on computer vision.* Springer. 2014, pp. 225–240.

[28] Wei Li et al. “Deepreid: Deep filter pairing neural network for person re-identification”. In: *Proceedings of the IEEE conference on computer vision and pattern recognition.* 2014, pp. 152–159.

[29] Tsung-Yi Lin et al. "Focal loss for dense object detection". In: *Proceedings of the IEEE international conference on computer vision*. 2017, pp. 2980–2988.

[30] Wei Liu et al. "Ssd: Single shot multibox detector". In: *European conference on computer vision*. Springer. 2016, pp. 21–37.

[31] Ishan Misra and Laurens van der Maaten. "Self-supervised learning of pretext-invariant representations". In: *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 2020, pp. 6707–6717.

[32] Mehdi Noroozi et al. "Boosting self-supervised learning via knowledge transfer". In: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*. 2018, pp. 9359–9367.

[33] Ioannis Papakis, Abhijit Sarkar, and Anuj Karpatne. "Gcnnmatch: Graph convolutional neural networks for multi-object tracking via sinkhorn normalization". In: *arXiv preprint arXiv:2010.00067* (2020).

[34] Jinlong Peng et al. "Chained-tracker: Chaining paired attentive regression results for end-to-end joint multiple-object detection and tracking". In: *European Conference on Computer Vision*. Springer. 2020, pp. 145–161.

[35] Sergey Prokudin, Peter Gehler, and Sebastian Nowozin. "Deep directional statistics: Pose estimation with uncertainty quantification". In: *Proceedings of the European conference on computer vision (ECCV)*. 2018, pp. 534–551.

[36] Oren Rippel et al. "Metric learning with adaptive density discrimination". In: *arXiv preprint arXiv:1511.05939* (2015).

[37] Tim Salimans et al. "Improved techniques for training gans". In: *Advances in neural information processing systems* 29 (2016).

[38] Florian Schroff, Dmitry Kalenichenko, and James Philbin. "Facenet: A unified embedding for face recognition and clustering". In: *Proceedings of the IEEE conference on computer vision and pattern recognition*. 2015, pp. 815–823.

[39] Amarjot Singh, Devendra Patil, and SN Omkar. "Eye in the sky: Real-time Drone Surveillance System (DSS) for violent individuals identification using ScatterNet Hybrid Deep Learning network". In: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops*. 2018, pp. 1629–1637.

[40] Julian Straub et al. "A Dirichlet process mixture model for spherical data". In: *Artificial Intelligence and Statistics*. PMLR. 2015, pp. 930–938.

[41] Arulkumar Subramaniam, Athira Nambiar, and Anurag Mittal. "Co-segmentation inspired attention networks for video-based person re-identification". In: *Proceedings of the IEEE/CVF International Conference on Computer Vision*. 2019, pp. 562–572.

[42] Jae Kyu Suhr and Ho Gi Jung. "Rearview camera-based backover warning system exploiting a combination of pose-specific pedestrian recognitions". In: *IEEE transactions on intelligent transportation systems* 19.4 (2017), pp. 1122–1129.

[43] ShiJie Sun et al. "Simultaneous detection and tracking with motion modelling for multiple object tracking". In: *European Conference on Computer Vision*. Springer. 2020, pp. 626–643.

[44] Brian H Wang et al. "Deep Person Re-identification for Probabilistic Data Association in Multiple Pedestrian Tracking". In: *arXiv preprint arXiv:1810.08565* (2018).

[45] Zhongdao Wang et al. "Towards real-time multi-object tracking". In: *Computer Vision–ECCV 2020: 16th European Conference, Glasgow, UK, August 23–28, 2020, Proceedings, Part XI 16*. Springer. 2020, pp. 107–122.

[46] Longyin Wen et al. "Learning non-uniform hypergraph for multi-object tracking". In: *Proceedings of the AAAI conference on artificial intelligence*. Vol. 33. 01. 2019, pp. 8981–8988.

[47] Yihong Xu et al. "Deepmot: a differentiable framework for training multiple object trackers". In: *arXiv preprint arXiv:1906.06618* (2019).

[48] Dong Yi et al. "Deep metric learning for person re-identification". In: *2014 22nd International Conference on Pattern Recognition*. IEEE. 2014, pp. 34–39.

[49] Sergey Zagoruyko and Nikos Komodakis. "Wide residual networks". In: *arXiv preprint arXiv:1605.07146* (2016).

[50] Amir Roshan Zamir, Afshin Dehghan, and Mubarak Shah. "Gmcp-tracker: Global multi-object tracking using generalized minimum clique graphs". In: *European conference on computer vision*. Springer. 2012, pp. 343–356.

[51] Andrei Zanfir and Cristian Sminchisescu. "Deep learning of graph matching". In: *Proceedings of the IEEE conference on computer vision and pattern recognition*. 2018, pp. 2684–2693.

[52] Xiaohua Zhai et al. "S4l: Self-supervised semi-supervised learning". In: *Proceedings of the IEEE/CVF International Conference on Computer Vision*. 2019, pp. 1476–1485.

[53] Xuefei Zhe, Shifeng Chen, and Hong Yan. "Directional statistics-based deep metric learning for image classification and retrieval". In: *Pattern Recognition* 93 (2019), pp. 113–123.

[54] Wei-Shi Zheng, Shaogang Gong, and Tao Xiang. "Person re-identification by probabilistic relative distance comparison". In: *CVPR 2011*. IEEE. 2011, pp. 649–656.

# CURRICULUM VITAE

**NAME:** Abdelhamid Bouzid

**ADDRESS:** Computer Engineering & Computer Science Department
Speed School of Engineering
University of Louisville
Louisville, KY 40292

**EDUCATION:**

Ph.D., Computer Science
May 2023
**University of Louisville**, *Louisville, Kentucky*

M.S., Computer Science
August 2020
**University of Louisville**, *Louisville, Kentucky*

B.Eng., Polytechnic Engineering
June 2018
**Tunisia Polytechnic School**, *Tunis, Tunisia*

A.P.Chem., Associate Degree in Physics and Chemistry
June 2015
**The Monastir Preparatory Engineering Institute**, *Tunis, Monastir*

**JOURNAL PUBLICATIONS:**

- Bouzid, A.; Sierra-Sosa, D.; Elmaghraby, A. An Integrated vMF Embedding and VOS Framework for Robust Pedestrian Re-Identification and Out-of-Distribution Detection 2023.
- Bouzid, A.; Sierra-Sosa, D.; Elmaghraby, A. Directional Statistics-Based Deep Metric Learning for Pedestrian Tracking and Re-Identification. Drones 2022, 6, 328.
- Moalla, M., Frigui, H., Karem, A. and Bouzid, A., 2020. Application of Convolutional and Recurrent Neural Networks for Buried Threat Detection Using Ground Penetrating Radar Data.

IEEE Transactions on Geoscience and Remote Sensing.

## PROJECTS AND INTERNSHIPS:

1. Machine learning engineer at LANDING AI, March 2023 - Today
2. Machine learning engineer intern at LANDING AI, November 2021 - March 2023
3. Graduate Research Assistant in University of Louisville, January 2019 - November 2021
4. Graduation intern in University of Louisville , Mars 2018 - December 2018
5. Engineer intern in COSIM Laboratory , July 2017 - August 2017

## HONORS AND AWARDS:

1. CECS Master of science Award, April 2020
2. Top 1% student in the National Exam for Admittance to Engineering Schools Tunisia: September 2015