FACULTY OF INFORMATION TECHNOLOGY CTU IN PRAGUE

Assignment of master's thesis

Title:	People detection and re-identification from a stationary camera located indoors
Student:	Bc. Adam Jirovský
Supervisor:	Ing. Lukáš Brchl
Study program:	Informatics
Branch / specialization:	Knowledge Engineering
Department:	Department of Applied Mathematics
Validity:	until the end of summer semester 2021/2022

Instructions

The aim of this work is to design and implement algorithms to enable the detection of people in video recording and the creation of their robust and unique identity by extracting image characteristics. When people leave the scene, their identities need to be stored appropriately and effectively. For a new person in a scene, the stored identities are compared with a person's image characteristics, and if they match, the existing identity is associated with that new person. This can be useful, for example, for analysis of the people's movement in an indoor environment.

- Research existing solutions.
- Design and implement detection and re-identification algorithms using computer vision methods.
- Research and implement appropriate output metrics (might be related to time, position, and biometrics).
- Consider creating your own dataset to evaluate algorithms or choose from existing ones.
- Evaluate the results achieved and suggest future improvements.

Electronically approved by Ing. Karel Klouda, Ph.D. on 1 October 2020 in Prague.

FACULTY
OF INFORMATION
TECHNOLOGY
CTU IN PRAGUE

Master's thesis

People detection and re-identification from a stationary camera located indoors

Bc. Adam Jirovský

Department of Applied Mathematics
Supervisor: Ing. Lukáš Brchl

May 6, 2021

Acknowledgements

I would like to thank my supervisor Ing. Lukáš Brchl for his guidance and helpful advices. I would also like to thank my family and friends for supporting me both during writing of this thesis and during entirety of my studying.

Declaration

I hereby declare that the presented thesis is my own work and that I have cited all sources of information in accordance with the Guideline for adhering to ethical principles when elaborating an academic final thesis.

I acknowledge that my thesis is subject to the rights and obligations stipulated by the Act No. 121/2000 Coll., the Copyright Act, as amended, in particular that the Czech Technical University in Prague has the right to conclude a license agreement on the utilization of this thesis as a school work under the provisions of Article 60 (1) of the Act.

In Prague on May 6, 2021 . . . . . .

Czech Technical University in Prague
Faculty of Information Technology

Citation of this thesis

Jirovský, Adam. *People detection and re-identification from a stationary camera located indoors.* Master's thesis. Czech Technical University in Prague, Faculty of Information Technology, 2021.

Abstrakt

Cílem této práce je tvorba systému schopného detekovat a sledovat pohyb osob pomocí informací ze stacionární kamery. Systém také dokáže z detekcí extrahovat biometrické informace jako věk a pohlaví. Využití tohoto systému se nabízí zejména v komerčním prostředí, kde obchod může použít tyto informace k predikování chování zákazníků a/nebo plánování marketingových strategií.

Klíčová slova počítačové vidění, re-identifikace, detekce osob, sledování více objektů

Abstract

The goal of this thesis is the creation of a system, which is able to detect and track persons using information from a stationary camera. This system is also able to extract biometric information of age and gender from the detections. This can be useful for example in a commercial setting, where a retail store can use this information to predict customer behavior and/or plan marketing strategies.

Keywords computer vision, re-identification, person detection, multiple object tracking

Contents

List of Figures

List of Tables

Introduction

The topic of this thesis is to develop a system that, when given a video sequence, is able to track people in it by creating unique identities extracted from their image characteristics.

The proposed system is also able to extract biometric data from the persons, such as age or gender, which then can be used to create statistics from the environment of the video sequence.

This thesis is mainly focused on Multiple object tracking (MOT), which is the task of accurately following multiple objects in a sequence, with emphasis on minimal error of object identification. Generally, MOT consists of two main objectives: object detection and re-identification (re-ID) of said objects in-between frames, thus creating one identity per object. While the scope of this thesis is solely person tracking, MOT can be applied to any object in general.

Objectives

This thesis aims to design an algorithm that enables the detection of people in video recording and is able to assign an identity to the person based on the image information extracted from the detections.

The algorithm is also able to store identities of detected persons effectively and in such a way that person that left the scene and later returned can be re-identified.

Prerequisite of this is research of the current state-of-the-art in all the necessary fields.

After that, based on the research and experimentation, an algorithm is designed and implemented. Following the implementation, the algorithm's performance is evaluated on the target environment.

Motivation

As a result of the growing number of surveillance systems and thanks to the progress of AI and ML, the popularity of MOT and person re-ID is significantly increased.

For example, retail stores can use the statistics obtained by MOT to evaluate marketing strategies and/or predict customer behavior.

The task of person re-ID, as a task where from queried detection we try to find a match from known identities, can be used in a number of cases, for example, in search for a missing person.

Challenges

While advancement in MOT and re-ID alone is noteworthy, in the practical world they encounter their own set of challenges.

Possible sources of confusion that need to be tackled are overlapping or occluded detections, where in some of the frames, the detected person's image characteristics are disturbed by either information and/or the wrong person's characteristics.

Another source of confusion includes possible lightning changes or a person changing part of its wardrobe, for example, putting on/taking off a backpack in between the detections.

This thesis aims to create a robust MOT system, that tackles these challenges and can be used in a practical real-life setting.

Structure of the thesis

Work is separated into the following chapters:

- **Multiple object tracking** – This chapter provides an overview of the field of MOT.
- **Object detection** – This chapter provides an overview of the field of object detection, including state-of-the-art algorithms
- **Person re-identification** – This chapter provides an introduction to the field of person re-identification, including an overview of state-of-the-art methods.
- **Analysis** – Chapter Analysis provides an analysis of our target environment and requirements on our designed algorithm.
- **Method** – Chapter Method, describes the implementation of the algorithm based on the research conducted in previous chapters.

- **Experiments** – Chapter Experiments describes experiments conducted as a part of this thesis, including performance evaluation of the algorithm.

Chapter 1

Multiple object tracking

Multiple object tracking aims at predicting trajectories of multiple targets in video sequences. MOT is applied in a number of fields including, but not limited to autonomous driving or smart video analysis [1].

The main strategy for this problem is an approach called *tracking-by-detection* [15], which breaks MOT to the detection step, during which the objects are located and the association step, during which the detections are assigned to existing trajectories.

To successfully complete these steps MOT system uses two components: object detector and re-ID model (described in chapters 2 and 3) to obtain discriminative identities of the object from cropped detection.

With the obtained identities the detected objects are then associated with trajectories. This is usually done by creating a cost matrix based on information about differences between trajectories and detections. The matches are then obtained with emphasis on minimizing total cost.

One of the most popular strategies to obtain matches with the minimal total cost is the Hungarian method [16] (also known as Kuhn-Munkres algorithm) as presented in alg. 1.0.1.

An example approach using this method is Simple Online Realtime Tracking (SORT) [18]. SORT uses Kalman Filter [19] to predict the track's location in the next frame and then constructs cost matrix using IoU (Intersection of Union) of the prediction and the newly detected objects.

While this approach is computationally very inexpensive, it is prone to identity switches and losing tracks when detections are occluded and/or overlapped due to its distance-based approach.

In recent years the focus has shifted towards approaches using neural networks to create embeddings from the visual information of the image. Based on the architecture of its network we classify those systems into **Separate detection and embedding** (SDE) and **Joint detection and embedding** (JDE).

Algorithm 1.0.1: Hungarian method [17]

Input : m x n cost matrix C
Output: list of tuples of between rows and columns of C

1 For each row, find the lowest element and subtract it from each element in that row.
2 For each column, find the lowest element and subtract it from each element in that column.
3 Cover all zeros in the resulting matrix using a minimum number of horizontal and vertical lines. If $min\{m, n\}$ lines are required, an optimal assignment exists among the zeros. The algorithm stops.
4 Find the smallest element (call it k) that is not covered by a line in Step 3. Subtract k from all uncovered elements, and add k to all elements that are covered twice.
5 **return** (row, column) indices of every 0 in matrix

1.1 Separate Detection and Embedding

SDE model considers detector and embedding model as two separate networks as illustrated in Figure 1.1. Targets localized using the detector are then cropped and fed to the re-ID model.

Overall inference time of this approach for an image is roughly the sum of the inference time of both components. A disadvantage of this approach is the fact, that inference time is heavily dependent on the number of detections.

To create an SDE model any valid object detector and re-ID model can be combined, hence both models can be chosen separately, to achieve the best possible results in both tasks. An example of this approach is DeepSORT [20]. DeepSORT extends upon SORT by using a neural network to obtain appearance information from the detected object.

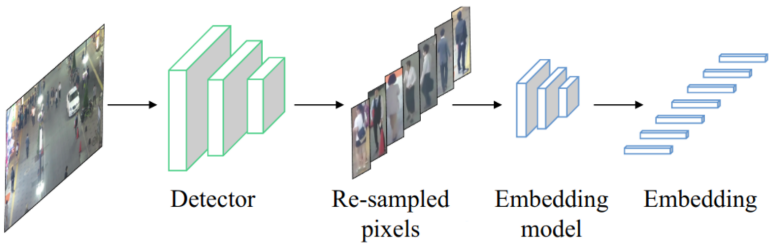

Figure 1.1: SDE architecture [1].

1.2 Joint Detection and Embedding

JDE model combines the object detection and re-ID into a single network, thus simultaneously outputting detection results and their appearance embeddings. An illustration of this architecture can be seen in Figure 1.2

This method usually leads to a significant performance boost, due to avoiding re-computation of low-level features.

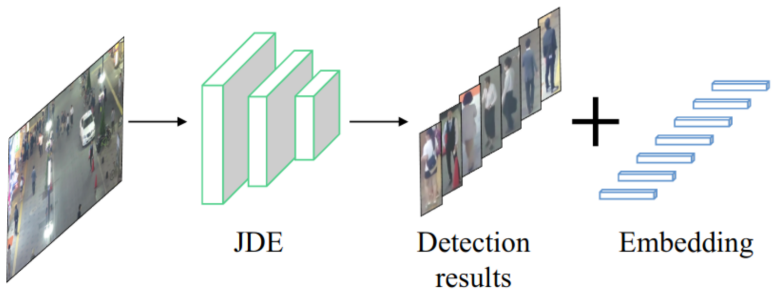

Figure 1.2: JDE architecture [1].

Notable example of JDE is a model called *FairMOT* [2]. As illustrated in Figure 1.3, its network consists of encoder-decoder, for extraction of high-resolution feature maps, detection branch, and re-ID branch. The detection branch of *FairMOT* is built on object detector CenterNet, because of its anchor-free approach, to avoid issues with anchor-based detectors, such as multiple anchors corresponding to one identity [2].

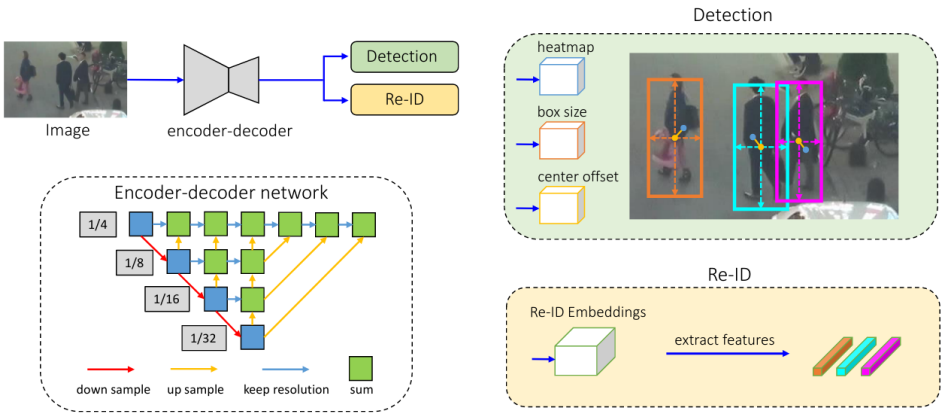

Figure 1.3: *FairMOT*network architecture [2].

For robust storing, information about multiple embedding vectors should be saved for every identity. One way to achieve this, is saving vectors of last

N occurences and then use clustering distance metrics, which can be chosen from following examples:

Minimum – $min\{d(a,b) : a \in A, b \in B\}$ – Consider the distance between track A and detection B as a distance of the two closest vectors.

Maximum – $max\{d(a,b) : a \in A, b \in B\}$ – Consider the distance between track A and detection B as a distance of the two farthest vectors.

Mean – $mean\{d(a,b) : a \in A, b \in B\}$ – Consider the distance between track A and detection B as a mean of all distances.

Centroid – $\|c_a - c_b\|_2$ – Consider the distance between centroid of track A and centroid detection B.

However, in the field of tracking, a popular method of storing embedding vector representing track is updating the embedding every detection as defined in equation 1.1, where e is the embedding vector of the detection and λ is smoothing parameter, which decides the importance of the past vectors and follows $0 \leq \lambda \leq 1$. This method is used for example in *FairMOT* [2] and tends to perform better and with more efficient memory representation than clustering methods.

$$y_t = \lambda * e + (1 - \lambda) * y_{t-1} \tag{1.1}$$

1.3 Evaluation metrics

The quality of the MOT system is measured by several metrics with the most popular being number of identity switches (IDSW), multiple object tracking accuracy (MOTA) and multiple object tracking precision (MOTP) [15].

These metrics use the following variables:

True positive (TP) – Any prediction that is correctly matched to ground truth is true positive.

False positive (FP) – Any extra predictions that are not matched to ground truth tracks are false positives.

False Negative (FN) – Ground truth tracks that are not matched are false negatives.

1.3.1 Number of identity switches

Number of identity switches (IDSW) counts the number of situations when a target is incorrectly assigned a different identity then in the previous frame or when a target's identity becomes lost and is then reinitialized with a different identity.

1.3.2 Multiple object tracking accuracy

Multiple object tracking accuracy (MOTA) measures three types of possible errors, which are the number of false positives, number of false negatives and number of identity switches.

As defined in equation 1.2, MOTA is calculated by calculating a sum of the number of occurences errors over all the frames, divided by the number of ground truth tracks. This sum is then subtracted from one.

$$MOTA = 1 - \frac{\sum_t (FN_t + FP_t + IDSW_t)}{\sum_t GT_t} \tag{1.2}$$

1.3.3 Multiple object tracking precision

MOTP measures the average dissimilarity between true positives detections and their ground truth tracks. It shows the tracker's ability to precisely locate the target positions.

Definition can be seen in equation 1.3 with c_t being number of matches in frame t and $d_{t,i}$ being the distance between bounding boxes of target i and its ground truth.

$$MOTP = \frac{\sum_{t,i} d_{t,i}}{\sum_t c_t} \tag{1.3}$$

Chapter 2

Object detection

Object detection is a field of computer vision which aims to detect certain objects, for example, humans, buildings or cars, from video or image source. With re-ID model it is one of the building blocks for MOT system.

Similar to MOT systems, while object detectors can be used for basically any object, due to the scope of this thesis, we consider mainly person detection.

Person detection is a subset and one of the most important fields of the object detection problem and has been widely implemented in a number of aspects such as surveillance, autonomous driving and many more.

2.1 Detector architecture

Object detector architecture can be divided into two categories of one-stage detectors and two-stage detectors [3]. As seen in Figure 2.1, two-stage detectors first propose regions of interest (RoI), which are in the second stage classified into object classes. One-stage detectors propose predicted boxes with their classes directly from input, without region proposal step.

In general two-stage detectors tend to have higher localization and object recognition accuracy, but also tend to be more computationally expensive, thus making them less appropriate for real-time detection than their one-stage counterparts.

Both of these architectures share a common backbone network that conducts basic feature extraction with the input of the network being an image and the output being a corresponding feature map. Most of these backbone networks for detection are networks for classification without the last fully-connected layers.

The backbone network is usually chosen based on requirements on accuracy and efficiency. Deeper and densely connected networks like *ResNet* [21] should achieve more competitive accuracy than shallower networks like *VGG* [22].

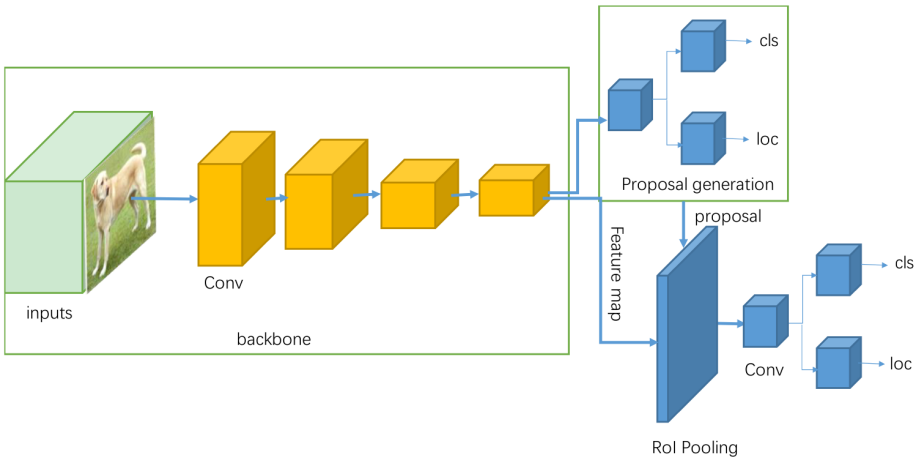

Figure 2.1: Two-stage detector architecture [3].

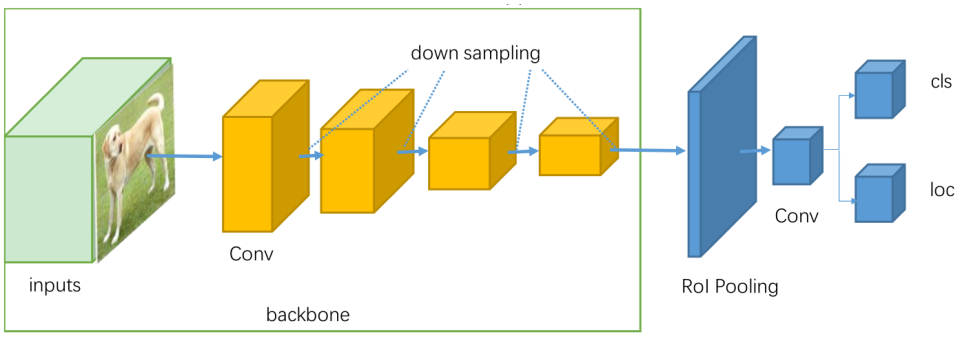

Figure 2.2: One-stage detector architecture [3].

2.2 State-of-the-art models

This section provides a few of the state-of-the-art models with a short description of each of them.

2.2.1 Faster R-CNN

Faster R-CNN [4] is an example of two-stage detector. It uses a convolutional network called Region Proposal Network (RPN) to efficiently predict region proposals. RPN achieves this by sliding a small network over the feature map output of a convolutional layer, which is then fed simultaneously into box-regression and box-classification layer.

For each position of the sliding window, the model uses so-called anchors, which are bounding boxes used to capture the scale and aspect ratio of the object. Illustration of region proposal network can be seen in Figure 2.3.

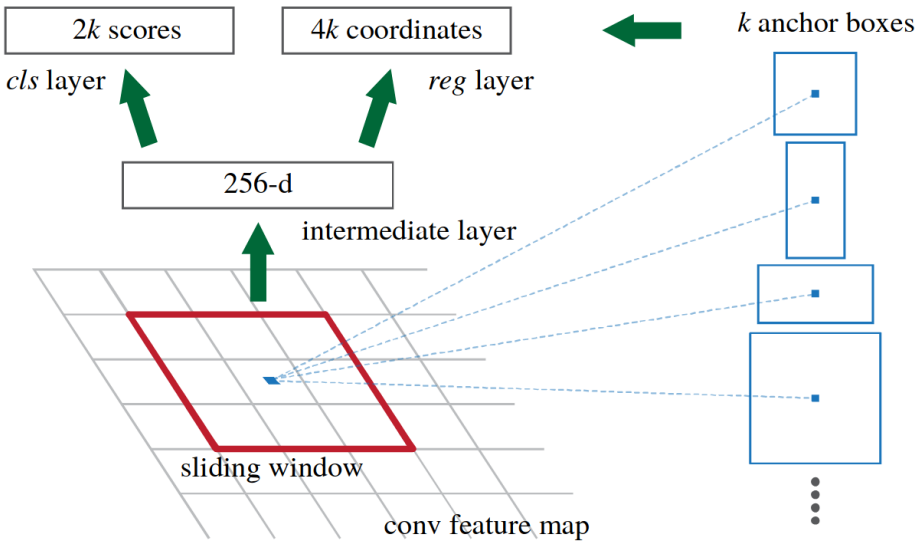

Figure 2.3: Architecture of RPN. At every pass of sliding window up to predefined k detections is fed to classification *cls* and *req* regression layer [4].

2.2.2 YOLO-v4

Yolo-v4 [23] is a detector from the family of one-stage detectors named *You only look once (YOLO)*, which similarly to *Faster R-CNN* use an anchor-based approach. It improves upon its previous version *Yolo-v3* by several features.

These features are separated into two categories called Bag of Specials, which provide accuracy of the model for a small increase of the inference cost, and Bag of freebies, which are methods to enhance the training phase, thus providing accuracy improvements without any increase to inference cost.

Examples of Bag of Specials features are special activation function called mish activation and cross-stage partial connections, which separates input feature map into two parts, where one of the parts skips one of the stages of inference as illustrated in Figure 2.4.

Examples of Bag of freebies features are cosine annealing scheduler for learning rate and DropBlock regularization, which similarly to dropout hides part of the input during the training, with the difference of dropping continuous regions as opposed to dropout's completely random information removing.

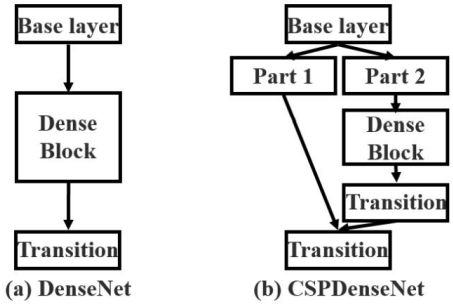

Figure 2.4: Illustration network without CSP connections at a) and with CSP b) [5].

2.2.3 CenterNet

CenterNet [6] is center point based detector. Instead of bounding boxes generated by sliding window, it represents detections by a single point at the center of the object, with other features regressed from image features at the location of the detected center as illustrated in Figure 2.5.

There is several possible backbones for the *CenterNet* detector, with backbone *DLA-34* achieving best results in accuracy and frames per second (FPS) trade-off.

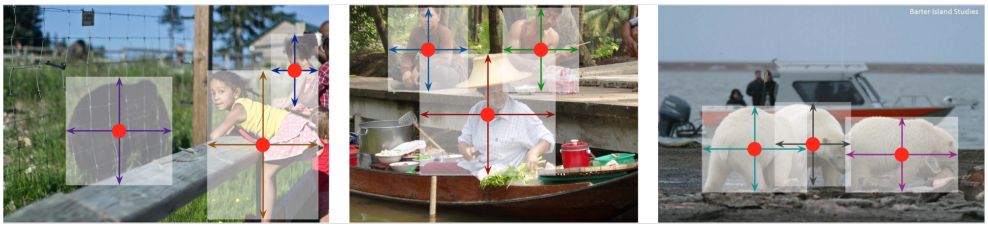

Figure 2.5: Illustration of center-based detectors [6].

When an input image is fed to the network it generates a heatmap with peaks corresponding to the centers of the objects. The width and height of the object are then predicted from the image features in an area of the heatmap peak.

This approach provides some benefits as no need for non maximum suppression due to the fact that there cannot be multiple detections overlapping, as is the case with anchor-based detectors.

2.3 Datasets

The task of detection has several famous datasets with the most popular being MS COCO (Microsoft Common Objects In Context) [24] dataset for general object detection. Another important dataset specialized for person detection is CrowdHuman [14].

To evaluate the performance of an object detector one of the metrics used is called average precision (AP). Average precision uses Intersect over Union (IoU), which is the area of intersect over the area of union of two bounding boxes, precision and recall.

Precision and recall can be seen defined in equations 2.1 and 2.2 respectively, with True Positives (TP) being a number of detections where IoU between the detected bounding box and a ground truth annotated bounding box is larger than a predefined threshold τ.

False Positives (FP) is a number of detections not fulfilling this condition and False Negatives (FN) being a number of objects overlooked by the detector.

$$Precision(\tau) = \frac{TP}{TP + FP} \tag{2.1}$$

$$Recall(\tau) = \frac{TP}{TP + FN} \tag{2.2}$$

Average precision is then obtained by averaging the values of precision at specified recall values of $\{0, 0.1, \ldots, 1\}$ on the precision/recall curve.

2.3.1 MS COCO

MS COCO is a dataset containing 164K images with annotations for 80 object categories including persons. For person detection purposes, usually, a subset of only person annotations called COCOPersons is used. An example of an image from this dataset can be seen in Figure 2.6.

For illustration of performance of the previously mentioned models, table showing their results on MS COCO test-dev benchmark is included in table 2.1.

	MS COCO AP
Faster R-CNN	43.9
Yolo-v4	**56.0**
CenterNet-DLA-34	41.6

Table 2.1: Results achieved by models on benchmark datasets.

Figure 2.6: MS COCO image example with annotated objects [7].

2.3.2 CrowdHuman

Opposed to COCO, CrowdHuman focuses only on person detection and thus contains annotations of the head bounding box, human visible bounding box and full-body bounding box as illustrated in Figure 2.7.

Another difference between the CrowdHuman dataset and COCO dataset is the focus of CrowdHuman on crowd scenarios, thus having much more persons per image as seen in comparison table 2.2.

	COCOPersons	**CrowdHuman**
Images	64,115	15,000
Persons	257,252	339,565
Ignore regions	5,206	99,227
Person/image	4.01	22.64
Unique persons	257,252	339,565

Table 2.2: Statistics of benchmark datasets [14].

Figure 2.7: CrowdHuman image example [8]. The bounding box for head, visible human and full human (dashed) can be seen. Occluded detections are crossed out.

Chapter 3

Person re-identification

Similarly to object detection, the task of person re-ID is an important building block for an MOT system. The task itself is defined as a subset of person retrieval problem, where given a query of a person (specified by either image, video sequence or text), the goal is to match it to a previous detection of the same person when there is one.

There is a number of challenges based on the configuration of the target domain which can lead to mistakes in determining the identity of a person, thus making the task of re-ID much harder to solve. These challenges include different viewpoints of the person between detection either by the person simply turning around or by detection from different cameras from different environments, changes in lighting, overlapping detections and/or occluded detections and many more.

3.1 Re-ID system architecture

As shown in Figure 3.1, default re-ID architecture consists of a gallery of saved persons which for every query of detected pedestrians retrieves the most relevant result.

3.1.1 Development

Default development of re-ID system consists of five steps [10]:

1. **Raw data collection** – Obtaining raw video data from cameras, often from more sources at once with different environments.

2. **Bounding box generation** – Extraction of bounding boxes containing person from the raw data. Often done by person detection or tracking algorithms, instead of manual cropping.

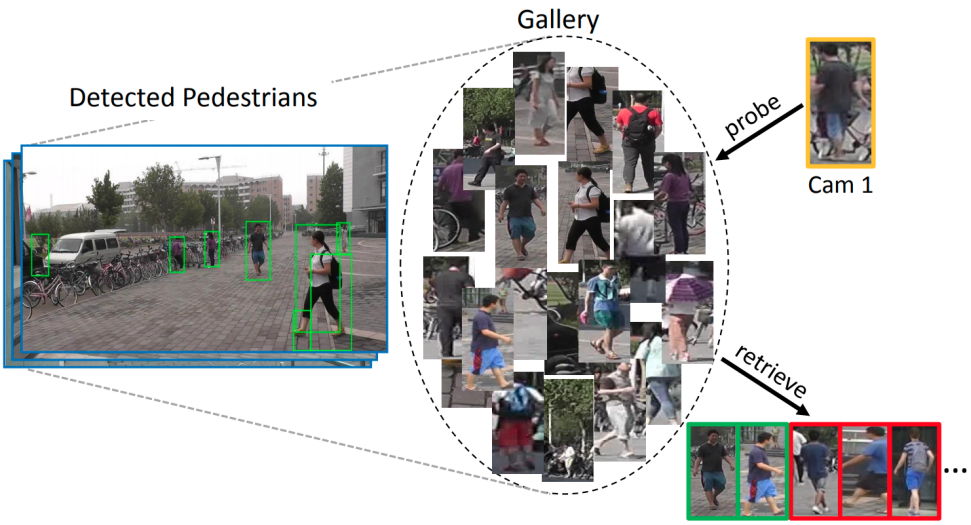

Figure 3.1: Default re-ID scheme [9].

3. **Training data annotation** – Annotating labels of the detected cropped persons. This can be done by an imperfect re-ID model with subsequent correction of errors.

4. **Model training** – Training a robust re-ID model with the annotated data is the main step of developing a re-ID system. Development of the model can be focused on feature representation learning, distance metric learning, handling various challenges of the task such as occlusion or any combination of these.

5. **Person retrieval** – Testing phase of the trained model is conducted by person retrieval. Given a query and a gallery set, the model is used to extract a representation of the persons, and a ranking list is obtained by sorting the query-to-gallery similarity.

3.1.2 Closed-world versus open-world setting

Based on steps defined in subsection 3.1.1, re-ID methods can be divided into two categories of open-world and closed-world settings. As shown in table 3.1 the difference between open-world and closed-world settings can be seen in following sections [10]:

Single-modality versus heterogeneous data – During the raw data collection in a closed-world setting, all data sources are captured by single-modality cameras, while in open-world there might be a need for the

Closed-world	Open-world
Single-modality data	Heterogeneous data
Bounding boxes generation	Raw images/videos
Sufficient annotated data	Unavailable/limited labels
Correct annotation	Noisy annotation
Query exists in gallery	Open-set

Table 3.1: Closed-world vs. Open-world re-ID [10]

processing of heterogeneous channels of data like infrared images, depth images, text descriptions and many more.

Bounding box generation versus raw images/videos – For the closed-world setting re-ID model uses generated bounding boxes of detected persons for training and testing. However, in an open-world setting some systems require end-to-end person search, where the model is able to process raw images/videos without the need for an extra step of bounding box generation. This type of system is more described in subsection 3.3.2.

Sufficient annotated data versus unavailable/limited labels – In a closed-world setting there is enough annotated training data for supervised model training. In contrast, in an open-world setting annotating every piece of data can be too time-consuming and there might be a limited number of labels or even no labels at all, leading to unsupervised or semi-supervised re-ID.

Correct annotation versus noisy annotation – Closed-world setting person re-ID expects correct annotations with clean labels. But this noise tends to be unavoidable in an open-world setting due to imperfect detections and/or annotation errors.

Query exists versus open-set – In person retrieval step, closed-world setting assumes that the queried individual occurs in the gallery set. However, in an open-world setting, it is highly likely that the person does not appear in the gallery set.

3.2 Closed-world re-ID

Standard close-world re-ID system consist of three main parts: *feature representation learning*, *loss function design* and *ranking optimization*. Feature representation learning focuses on the extraction of features, while loss function design focuses on training objectives and ranking optimization concentrates on optimization of the retrieved ranking list.

3.2.1 Feature representation learning

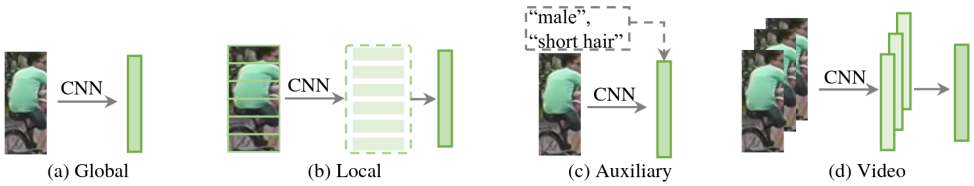

Figure 3.2: Feature representation learning strategies [10].

Closed-world feature representation learning consists of four categories which are illustrated in Figure 3.2:

Global feature – Strategy of global feature representation extracts one vector representing global feature for each person without additional cues.

Local features – Using local features, every person is characterized by a combination of aggregated part-level local features of regions, thus making it robust against misalignment. These regions are either automatically generated by parsing of the human by body parts or by horizontal division. Using body part detection, the popular solutions tend to be to combine representation of the full body with local features of parts or using the pose-driven matching to combat background clutter. Solutions for regions obtained by horizontal division are for example part-level classifiers such as Part-based Convolutional baseline (PCB) [11].

Auxillary features – Auxillary features improve the feature representation learning by auxiliary information. For example in [25] *semantic attributes* such as gender, are used to improve robustness of the feature representation. Another family of auxiliary features is using *domain information* where, for example, camera view information or location is used to improve the feature representation.

Video features – If each person can be represented by a video sequence with multiple frames, video features such as temporal information can be used. This is part of video-based re-ID and while these sequences tend to be rich in appearance and temporal information there is a lot of challenges to overcome. The primary challenge is to capture the temporal information accurately. Another issue that needs to be addressed is the fact that for video-based feature learning every detection needs to have a certain number of frames to be accurately represented.

3.2.2 Loss function design

In person re-ID the most popular and the most studied loss functions are:

Identity loss – Identity loss treats the training process as an classification problem, considering every identity as one of the classes as illustrated in Figure 3.3. In the evaluation, output of embedding layer is used as the feature extractor. Identity loss is computed by the cross-entropy equation specified in equation 3.1 [10] where x_i is the input image, y_i is the target label and $p(y_i|x_i)$ is the probability of x_i being y_i.

$$\mathcal{L}_{id} = -\frac{1}{N}\sum_{i=1}^{n}\log p(y_i|x_i) \tag{3.1}$$

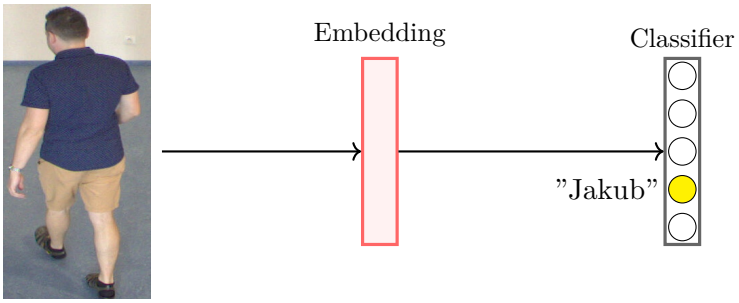

Figure 3.3: Illustration of identity loss.

Verification loss – Verification loss decides whether or not two persons are the same using a penalty system called contrastive loss as illustrated in Figure 3.4. The formula of verification loss is described in equation 3.2 [10], where $d(i,j)$ is euclidean distance between the embeddings of input samples x_i and x_j, δ_{ij} is a label identificator where $\delta_{ij} = 1$ when x_i and x_j are the same identity, $\delta_{ij} = 0$ otherwise with ρ as a margin parameter.

$$\mathcal{L}_{con} = (1-\delta_{ij})\{max(0, \rho - d_{ij})\}^2 + \delta_{ij}d_{ij}^2 \tag{3.2}$$

Triplet loss – Triplet loss looks at the training as a retrieval ranking problem using the assumption that the distance between a pair of images with the same identity is smaller than between a pair of images with different identities. Hence the distance is during training minimized between the positive pair and maximized between the negative pair as illustrated Figure 3.5. The triplet loss formula can be seen in equation 3.3 [10]

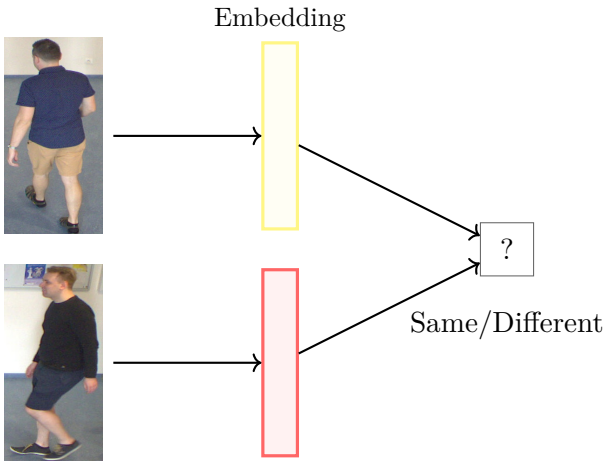

Figure 3.4: Illustration of verification loss.

with d_{ij} and d_{ik} being euclidean distance between the positive pair and negative pair respectively and ρ being pre-defined margin.

$$\mathcal{L}_{tri}(i,j,k) = max(\rho + d_{ij} - d_{ik}, 0) \tag{3.3}$$

3.2.3 Ranking optimization

Ranking optimization improves the retrieval performance in the testing stage by optimizing the ranking order of a given ranking list. One of those methods is rank fusion where multiple ranking lists are obtained using different methods, which are then merged into one final list.

3.3 Open-world re-ID

Open-world person re-ID consists of several interesting approaches to re-ID including heterogenous re-ID, end-to-end re-ID and open-set person re-ID.

3.3.1 Heterogenous re-ID

In heterogenous re-ID there are four prominent fields:

Depth-based re-ID – Depth-based re-ID uses depth images, that capture the body shape and skeleton formation, hence providing the possibility of re-ID in environments where there are either illumination or clothes changes.

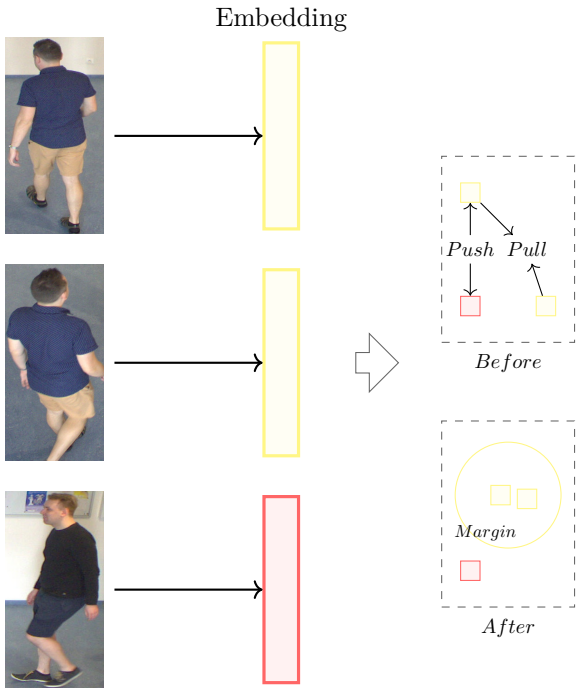

Figure 3.5: Illustration of triplet loss.

Text-to-image re-ID – Text-to-image re-ID provides matching of text description and RGB images, so it provides a possibility for re-ID when only text description can be provided.

Visible-infrared re-ID – Visible-infrared re-ID provides a solution for matching daytime visible images and night-time infrared images/

Cross-resolution re-ID – Cross-resolution re-ID matches between low-resolution and high-resolution images, thus combatting the resolution variation challenge of the re-ID task.

3.3.2 End-to-end re-ID

End-to-end re-ID avoids the step of bounding box generation and instead provides identities from raw images/videos. This addresses the issue of quality of re-ID features being highly dependent on the quality of object detections [2] and can lead to a boost in performance of the system, but proves to be challenging due to the different focuses of object detection and re-ID components.

3.3.3 Open-set person re-ID

Open-set person re-ID is formulated as a person verification problem, deciding if two person images belong to one identity. This usually requires a threshold condition, where the similarity of query and person gallery needs to be bigger than predefined $\tau \in \mathbb{R}^+$.

There are deep learning approaches such as Adversarial PersonNet [26], which learns re-ID feature extractor and GAN module to generate realistic target-like imposters, hence enforcing the feature extractor to be robust to these generated image attacks. However, this method is challenging due to the issue of striking balance between high true target recognition and low false target recognition rate.

3.4 State-of-the-art models

This section provides an overview of a number of state-of-the-art models with short description of each of them and their results on datasets introduced in section 3.5.

3.4.1 PCB

Part-based convolutional baseline (PCB) [11] uses a modified version of an image classification network, *e.g.*, *ResNet* [21] without the fully-connected layers for its re-ID purposes.

To create *PCB* network, the backbone is reshaped by removing the global average pooling layer and its following parts thus creating a 3D tensor, which is then down-sampled into p column vectors. The dimension of these vectors is then reduced using 1x1 kernel convolutional networks, which are connected to fully-connected layers to predict identities as is illustrated in 3.6.

To train the network cross-entropy loss function (eqn. 3.1) is used.

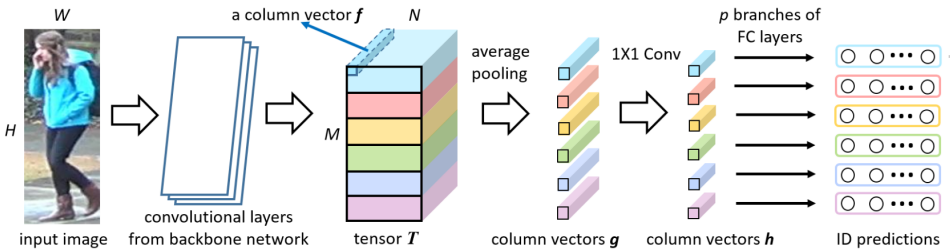

Figure 3.6: PCB architecture [11].

3.4.2 OSNet

OSNet [12] aims to overcome two challenges of re-ID. One of them are changes in camera views, *e.g.*, a single person detected once from the front and once from the back while carrying a backpack, thus creating a significant change in the backpack area and decreasing the probability of being detected as one identity.

The second challenge is the impostor issue, where two people wearing similar clothes can be interchanged when viewed from a distance. To overcome these issues *OSNet* aims to learn features of *omni-scale*, defined as a combination of multiple scales, hence making local regions, such as shoes or glasses, and body regions equally important.

To achieve these omni-scale features, *OSNet* consists of building blocks using multiple convolutional streams with different field sizes which are then aggregated using an adaptive aggregation gate. An illustration of this building block can be seen in Figure 3.7. This can be also seen as a feature representation combination of global features and local features as mentioned in subsection 3.2.1.

To train the model, the best results were achieved by combining the cross-entropy loss function (eqn. 3.1) with the triplet loss function (eqn. 3.3).

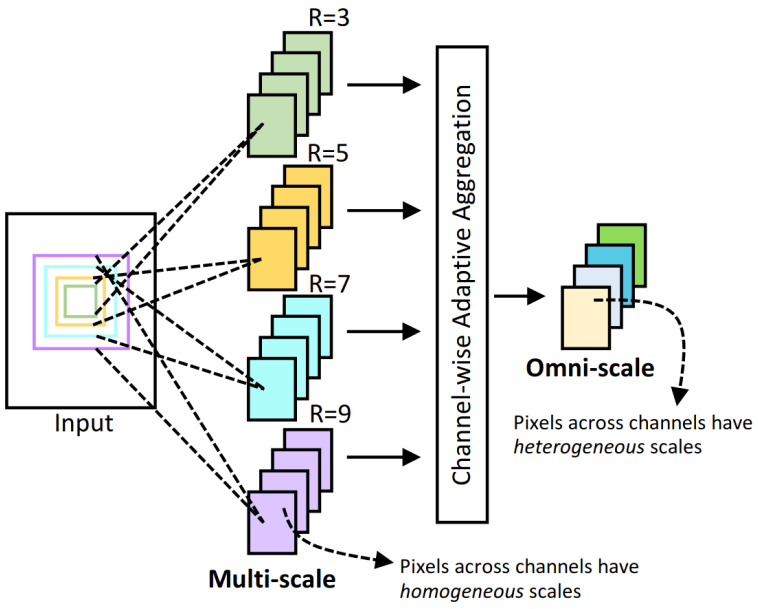

Figure 3.7: *OSNet* building block [12].

3.4.2.1 OSNet-AIN

OSNet-AIN [27] is a network based on *OSNet*, which additionally tries to tackle the issue of domain generalisation. Due to differences in, for example, lighting and background between domains, a re-ID model trained on a different dataset will often have performance issues on an unseen target dataset due to overfitting on the source domain.

Domain-generalisable model is therefore very useful as the model can work in a previously unseen scenario, without the need for data collection and annotation.

To achieve this, *OSNet-AIN* uses instance normalization (IN), which eliminates instance-specific contrast and style induced by the domain settings. Thanks to instance IN only introducing a small number of parameters into the model, OSNet-AIN retains its lightweight network benefits.

A noteworthy benefit of *OSNet* is the fact that it achieves competitive results, with a very lightweight network structure, thus making it less probable to overfit and making it much less computationally expensive.

3.4.3 ABD-Net

Attentive but Diverse Network (ABDNet) [13] aims to create embeddings that are both attentive, focusing on a person related features while eliminating background, and diverse, which aims to lower correlation between features, making the feature space more comprehensive.

Attention part is achieved by using special modules to capture the relationship between different convolutional channels and spatial patterns from person images, while the diversity part is being achieved by enforcing orthogonality by regularization using the Gram matrix. Architecture of *ABDNet* can be seen in Figure 3.8

Loss function used is a combination of cross-entropy (eqn. 3.1), triplet loss function (eqn. 3.3) and orthogonal constraints on feature and weights penalty terms.

3.5 Datasets and evaluation metrics

In re-ID there is a number of used for benchmarking models with the most popular being Market-1501 [28], CUHK03 [29] and MSMT17 [30]. Their statistics can be seen in table 3.2.

To evalute re-ID the most often used metrics are CMC-k and mAP (mean average precision). CMC-k (also known as Rank-k matching accuracy) is the probablity that in the top-k retrieved results is the correct match with $k = 1$ being used the most often. Metric mAP measures the average retrieval performance with multiple ground truths.

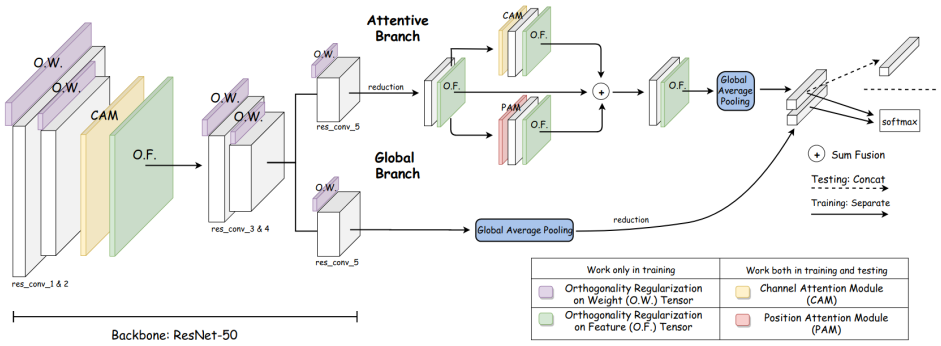

Figure 3.8: ABDNet architecture [13].

	CUHK03	**Market-1501**	**MSMT-17**
Date	2014	2015	2018
Identities	1,467	1,501	4,101
Images	13,164	32,668	126,441
Cameras	2	6	15
Resolution	various	fixed	various

Table 3.2: Statistics of benchmark datasets [10]

For illustration of model performances, comparison of the models mentioned in section 3.4 can be seen in table 3.3. As seen in the table, *ABDNet* is the best performing model on datasets Market-1501 and MSMT-17, while *OSNet* being the best performing model on dataset CUHK03.

	CUHK03		**Market-1501**		**MSMT17**	
	rank-1	mAP	rank-1	mAP	rank-1	mAP
PCB	63.7	57.5	63.8	81.6	68.2	40.4
OSNet	**72.3**	**67.8**	94.8	84.9	78.7	52.9
ABD-Net	N/A	N/A	**95.6**	**88.28**	**89.0**	**78.6**

Table 3.3: Results achieved by models on benchmark datasets [11], [12], [13]. "N/A" means the model was not tested on this benchmark.

Chapter 4

Analysis

This chapter describes the problem defined by the scope of this thesis, including assumptions about the target environment and the developed system.

The primary objective of this thesis is a development of a system that enables person tracking by a video obtained from stationary camera systems. These camera systems can consist of either a single camera or multiple cameras at different locations. Therefore, based on requirements described in section 3.1.1, we can classify the system as mostly closed-world re-ID, with the notable exception of a query not necessarily needing to be in the gallery.

4.1 Enviroment

The target environment should be an area close to the entrance to a retail store. This area should be well lit by natural daylight in case of an outdoor environment or artificial light in case of an indoor environment.

While minor lighting changes are expected and should be handled by the system, we assume that the environment will not be subject to major lighting changes, such as day and night cycles.

Another assumption of the environment is, that due to some factors, like the angle of the camera, some parts of the scene might contain irrelevant information. Henceforth, the developed algorithm should have a parametrizable way of filtering out these parts.

4.2 Age and gender estimation

As a secondary objective of this thesis, we extract biometric information in form of age and gender estimation for every tracked person. While techniques such as linear SVM [31] were used to obtain these estimates, in recent years the focus has shifted to convolutional neural networks (CNN), due to the higher

accuracy they provide in comparison. To obtain estimates of these attributes from CNN, facial detections tend to be used, as the best-performing methods.

Therefore the developed system needs to be able to detect faces and pair them with detected persons.

4.3 Dataset

To properly develop and evaluate the algorithm a valid dataset is needed. This dataset should consist of scenes similar to target environments.

Due to the issues and restrictions caused by the COVID-19 pandemic, we could not gather data for the dataset created specifically for this thesis, so we used the previously created dataset used by ImproLab laboratory here at FIT CTU.

This dataset, similarly to the target environment, consists of sequences of people walking in a well-lit environment. Similarly to real-life conditions, a person's trajectories are often overlapping and/or detection of the person is occluded by a wall.

The camera is located above an entrance to another room, thus limiting the usable field of view due to the angle of recording.

Examples of images from the dataset can be seen in Figure 4.1. On these illustrative images, we can see a representation of some of the challenges encountered such as overlapping person detections (top-right), occluded detections (bottom-left) and unusable detections (bottom-right).

For annotation of the dataset, we used *Faster R-CNN* with *Resnet-50* backbone to generate bounding boxes of detected people. We have chosen this detector for labeling because his high inference time makes him inappropriate for our online tracking, henceforth experiments conducted on this dataset will not be biased towards one of the tested detectors. For the generation of identity labels in the dataset, we used the *OSNet* model to generate identities and then corrected the errors caused by the model.

Figure 4.1: Images from used dataset

Chapter 5

Method

This chapter introduces the proposed algorithm used for tracking people. The chapter consists of several sections describing the algorithm including our definition of a track and overview of the design and implementation of the algorithm. Our approach for this algorithm was inspired by the algorithm used in *FairMOT* [2].

The introduced algorithm outputs a set of identities with an estimate of age and gender. The algorithm also outputs temporal information about every identity, such as times of the first and the last detection.

5.1 Track

To achieve requirements on the algorithm, identities are internally represented as a set of information, called tracks. A single track consists of the following parts:

ID – Unique integer used to identify a single person.

State – Track can reach several states based on its detection statistics. By default track is *tentative*. While track is *tentative*, it is not shown in the gallery, thus decreasing the probability of false positives at the cost of increasing the probability of false negatives. Since missing frames from tracks are not significant to our task, we do not consider this increase in false negatives as relevant. From *tentative*, track can become *confirmed* by having a predefined number of detections associated with it, or *false positive* if it is not detected for a longer period of time. If a *confirmed* track is not detected for a longer period of time, its state becomes *lost*.

Person bounding box – Person bounding box stores the coordinates of the last detection of a person outline.

Face bounding box – Face bounding box stores the coordinates of the last detection of a person's face.

Embeddings – Embedding vectors obtained by the feature extractor are stored.

Bounding box prediction – Prediction of a person's bounding box in next frame based on its trajectory using Kalman filter [19].

Estimation of age and gender – Estimation of age and gender are stored for every identity. To minimize the influence of false positives for both age and gender last ten estimates are saved. Age is then considered as a mean of these ten last estimates while gender is specified as the most frequent value of the estimations.

Detection times – Timestamps, represented as frame IDs, of the first and the last detection are stored in every track.

Cropped detection – Detection of a person is cropped from the input image and saved. This gives a human the possibility to fix any errors in identity matching made by the algorithm.

5.2 Algorithm overview

The algorithm uses MOT *tracking-by-detection* paradigm with the detection and association step. Every frame is processed by analysis which consists of **person detection** to generate bounding boxes, **feature extraction** to obtain embeddings from the detected persons for re-ID, **age and gender estimation** and **association** of detected persons to saved tracks.

5.2.1 Detection and feature extraction

In this section, detections of persons and faces are generated from the input images.

While it is possible to use one detector pretrained for person outlines detection and one detector pretrained for face detection, it is much more computationally efficient to use one of the state-of-the-art detectors trained on the CrowdHuman dataset (as mentioned in subsection 2.3.2) and obtain both face and person detections in one inference.

To fulfill our requirement on the algorithm established in section 4.1, we ignore detections where the center of a bounding box is located close to the parametrizable edge of the frame. To make this filter useful in more scenes we make filter use four independent parameters each specifying one side of the scene. Therefore, the algorithm can be set to ignore, for example, 10% of the left side of the screen and 0% of the right side of the screen.

To match face detection to its person counterpart, we propose function *FacePersonMatch* as described in alg. 5.2.1. This function uses points of bounding boxes defined as $ct = \{c_x, tl_y\}$, where c_x is the x coordinate of

center of the bounding box and tl_y is the y coordinate of top-left point of the bounding box. These points are chosen to significantly lower the probability of confusion with overlapping detections, as opposed to more intuitive choices like the top-left corner of the bounding boxes, which can be easily interchanged. Example of this error is illustrated in 5.1.

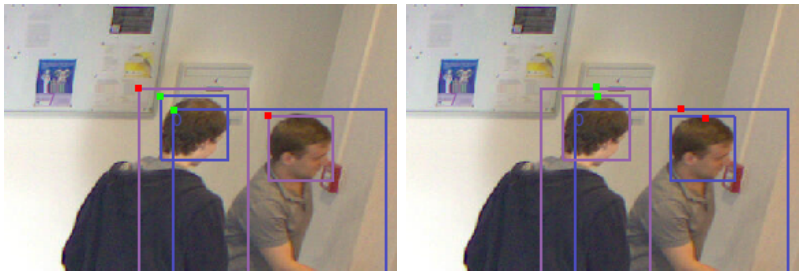

Figure 5.1: Example of matching based on distances of top-left corners (left) and *ct* (right).

Algorithm 5.2.1: FacePersonMatch

Input : Person bounding boxes p, face bouding boxes f
Output: person and face bounding box matches

$ct_p \leftarrow [ct_p \text{ of bbox for bbox in } p]$
$ct_f \leftarrow [ct_f \text{ of bbox for bbox in } f]$
$C \leftarrow \emptyset$
for *i in range* $(0, |ct_p|)$ **do**
    for *j in range* $(0, |ct_f|$ **do**
        $C[i, j] \leftarrow \|ct_p[i] - ct_f[j]\|_2$
$matches \leftarrow HungarianMethod(C)$
for *i,j in matches* **do**
    if $overlap(b[i], f[j]) < 0.75$ **then**
        remove i,j from matches
return *matches*

The function then calculates euclidean distances between *ct* points of every face and person detection and uses the Hungarian method (as specified in alg. 1.0.1) to obtain matches proposals. From these proposals, only those matches, where at least 75% of face detection area overlaps with the matched person detection are returned, to minimize false positives.

Due to the independence of face and person detection, two undesirable situations can occur. Firstly, the detected person might not have paired face detection to it. In this situation, we simply do not conduct age and face

estimation and wait until another frame is detected. The second possible problem is if we have extra face detection without paired person detection. Because face detection is not paramount to our track association we can simply ignore this detection.

After the detection is completed, detections of the persons are cropped from the input image and from these crops embedding vector is extracted using the re-ID model.

5.2.2 Age and gender estimation

In this section, age and gender estimations are generated for every detection. To obtain these estimations, cropped face detections are fed into an age and gender estimator based on network EfficientNetB3 [32].

To achieve better results in the estimation, the bounding box of the face detection is increased by a margin from all sides. Difference between raw detections and padded detections are illustrated in Figure 5.2.

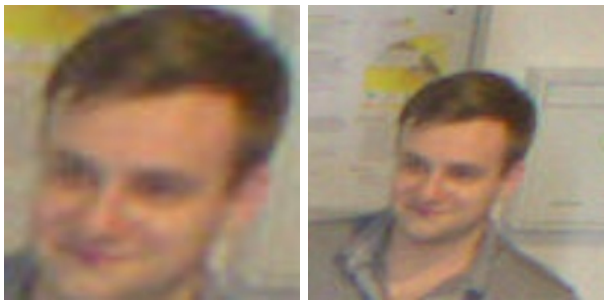

Figure 5.2: Raw and padded face detection.

5.2.3 Association

In the association step, the detected persons are connected to saved tracks. To achieve robust matching with minimal errors, our algorithm uses three layers of matching based on spatial difference, appearance difference and IoU of bounding boxes.

In the first step distance matrix $\mathcal{D}$ is created. Every element of this matrix is a distance between the bounding box of detection and a prediction of a *confirmed* or *lost* track's location provided by Kalman filter. Every distance larger than predefined $\tau_{\mathcal{D}}$ is replaced by infinity to avoid matching of detections and tracks too far away.

After that, matrix $\mathcal{R}$ is created. Elements of the matrix $\mathcal{R}$ are calculated using distances between the embeddings of detections and tracks. $\mathcal{D}$ and $\mathcal{R}$ are

then merged into one cost matrix $\mathcal{C}$ using equation 5.1, where λ is a predefined value deciding the weight of distance matrix on final cost matrix $\mathcal{C}$.

$$\mathcal{C} = \lambda * \mathcal{R} - (1 - \lambda) * \mathcal{D} \tag{5.1}$$

Similarly to before, every value of $\mathcal{C}$ larger than predefined threshold $\tau_{\mathcal{C}}$ is set to infinity to avoid matching corresponding track and detection. In the next step, detections and tracks are separated into matched and unmatched using function *MinCostMatching* (alg. 5.2.2). Example iteration of creation of $\mathcal{C}$ matrix is illustrated in Figure 5.3.

Algorithm 5.2.2: MinCostMatching

Input : Detection bounding boxes d, Track bounding boxes t
Output: matches, unmatched detections, unmatched tracks

$\mathcal{C} \leftarrow 1 - CostMatrix(d, t)$
$matches \leftarrow HungarianMethod(\mathcal{R})$
$unmatched\ tracks, unmatched\ detections \leftarrow \emptyset, \emptyset$
for *i,j in matches* **do**
    if $\mathcal{C}[i, j] > \tau_{\mathcal{C}}$ **then**
        remove *i,j* from *matches*
        add i to *unmatched tracks*
        add j to *unmatched detections*
return *matches*

In the second step, after the first matches are generated, algorithm uses function *IoUMinCostMatching* (alg. 5.2.3) to match remaining unmatched detections and tracks using IoU of bounding boxes. Since matching *lost* tracks based on their position with new detection would not yield reasonable results, in this step, we consider only *confirmed* tracks.

Matrix $\mathcal{I}_c$ is created by calculating every detection's overlap with its last detection of track and the Hungarian method is used to generate possible matches. Similar to before, matches with IoU smaller than predefined threshold τ_{IoU} are filtered.

In the last step, we create matrix $\mathcal{I}_t$, which is created identically to $\mathcal{I}_c$, with the only difference being that we use *tentative* instead of *confirmed* tracks.

After that, the association step is completed. Matched tracks are updated by newly obtained information, unmatched tracks have their unmatched counter increased and unmatched detections are saved into the gallery as a new track with *tentative* state. For updating the appearance vector we use the method introduced in equation 1.1.

An illustration of a step of the algorithm can be seen in Figure 5.5. Difference between our approach for track association and the approach of *FairMOT*

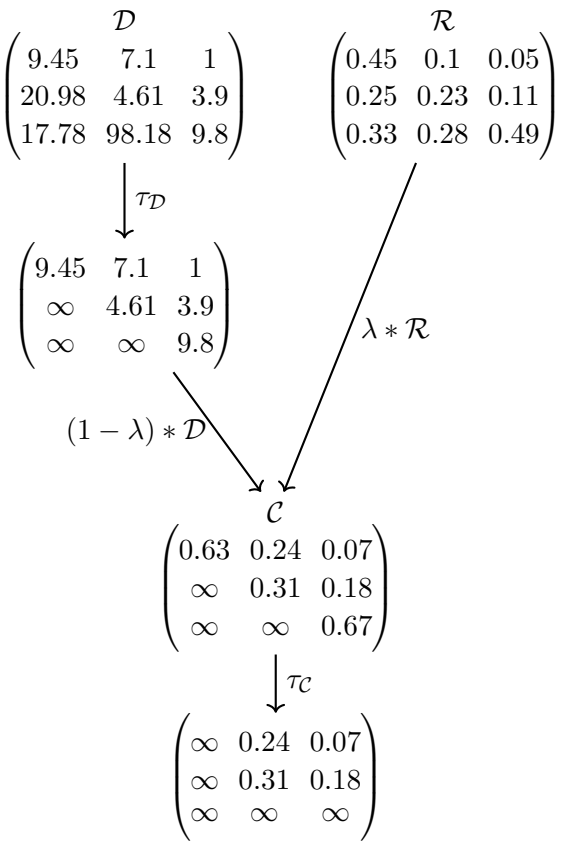

Figure 5.3: Example of $\mathcal{C}$ creation with $\tau_D = 10$, $\tau_C = 0.5$ and $\lambda = 0.98$

[2] is that *FairMOT* in the first step of association creates matrix $\mathcal{C}$ using *tentative*, *confirmed* and *lost* tracks, while our approach in this step considers only *lost* and *confirmed* tracks.

The reasoning behind this is that newly created, still unconfirmed tracks, tend to be volatile before enough appearance samples are collected and therefore we connect them solely by their position.

Another difference in our approach as opposed to *FairMOT* is the way of storing appearance embedding vectors. While *FairMOT* stores every appearance vector, our approach tries to filter out appearance vectors from detections where large enough overlap has been detected. This is done to combat the issue of overlapping detections, where part of the visual information from one detection will be stored in another detection. Since it is possible that the track will be overlapped by another track from the beginning, we apply this filter only after at least one clear detection has been captured.

Algorithm 5.2.3: IoUMinCostMatching

Input : Detection bounding boxes d, Track bounding boxes t
Output: matches, unmatched detections, unmatched tracks

$\mathcal{I} \leftarrow 1 - IoUMatrix(d, t)$
$matches \leftarrow HungarianMethod(\mathcal{R})$
$unmatched\ tracks, unmatched\ detections \leftarrow \emptyset, \emptyset$
for *i,j in matches* **do**
    if $\mathcal{I}[i, j] > \tau_{IoU}$ **then**
        remove *i,j* from *matches*
        add i to *unmatched tracks*
        add j to *unmatched detections*
return *matches*

5.3 Implementation

The programming language chosen for the implementation of this algorithm was Python. This language was chosen due to its support and popularity in the field of AI and ML.

For both object detection and re-ID deep learning framework *PyTorch* [33] was used and for age and gender estimation we used framework *TensorFlow* [34]. For input video sequence loading and processing, library *OpenCV*[35] was used.

5.3.1 MOTeel

As part of the scope of this thesis, GUI application named *MOTeel* using framework *Qt* [36] was developed. This application obtains a video sequence, or image folder as the input, which it afterward starts to process.

As illustrated in Figure 5.4, on the left side of the application, the gallery can be seen. For every confirmed track, information about age and gender is shown. IDs of first and last detection of this track are also included. Also, a cropped detection of the person is added for illustration purposes.

On the right side of the application, displaying canvas is located. On this canvas, an input image is displayed, with every detection generated by the object detectors being highlighted. In the top left corner of the canvas, the current frame ID with FPS can also be seen.

Additionally, if a confirmed track is not detected on a current frame, Kalman filter prediction can still be seen to illustrate the track's possible location.

For the user's convenience, an object detector and re-ID feature extractor can be chosen at the start of the application.

Figure 5.4: MOTeel application.

Currently supported detectors are *Faster R-CNN*, *Yolo-v4*, *CenterNet* and *FairMOT*. In case of *FairMOT*, application does not initialize feature extractor and uses *FairMOT* for both detection and feature extraction, thus decreasing computional expense.

Supported feature extractors are *ABDNet*, *PCB*, *OSNet* and *OSNet-AIN*. For implementation of all feature extractors except *ABDNet*, library *Torchreid*[37] was used.

To increase the speed of the application, the possibility of turning off age and gender estimation is implemented. In this mode, the application stores only temporal information about tracks.

Upon closing, the application saves the content of the gallery in JSON format and outputs it together with illustrative cropped detection.

This JSON contains following information about every *confirmed* or *lost* track:

ID – Unique integer identifying track.

First_detection – ID of the first frame where track was detected.

Last_detection – ID of the last frame where track was detected.

Age – Integer specifying age estimate (hidden if analysis of age and gender is turned off).

Gender – String specifying gender estimate (hidden if analysis of age and gender is turned off).

5.3.2 Evaluation scripts

For purposes of MOT evaluation, testing script was developed using library *py-motmetrics* [38]. This script gets video sequence, object detector and feature extractor as input, and outputs evaluation of the system's performance using MOT metrics such as MOTA and MOTP (defined in section 1.3).

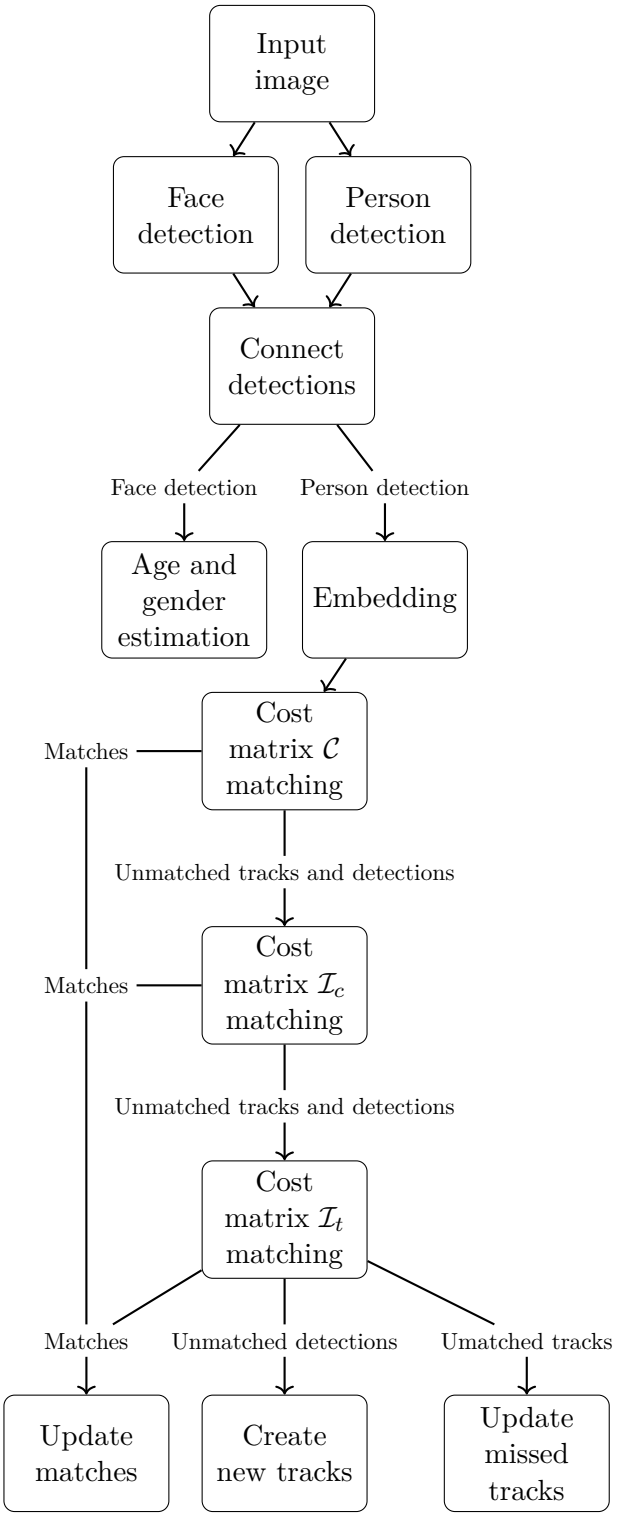

Figure 5.5: Illustration of algorithm iteration.

Chapter 6

Experiments

In this chapter, we describe experiments conducted while working on this thesis. These experiments consist of training our object detector and evaluation of our proposed algorithm on benchmark introduced in this chapter.

6.1 Detector training

To achieve all the requirements on our algorithm, we trained object detectors *CenterNet* and *Yolo-v4* on CrowdHuman dataset [14]. This leads to our detector being able to detect both heads and persons in one inference.

Achieved results on the CrowdHuman testing set can be found in table 6.1 with highlighted results being the best.

	Person AP	**Heads** AP
CenterNet	81.9	82.1
Yolo-v4	**82.47**	**83.03**

Table 6.1: Results achieved on CrowdHuman test set.

This is comparable with some of the slower detectors. For example, object detector *Faster R-CNN* achieved 84.95% on AP of person detection [14].

6.2 Evaluation

In this section, we provide evaluation of our proposed association algorithm on a testing set simulating the target enviroment and comparison with established algorithm used in *FairMOT*. We also provide experiments illustrating best possible combination of detector and feature extractor for our target enviroment.

Description of the testing set is described in following subsection.

6.2.1 Testing dataset

Our testing dataset consists of three scenes put together to target as much possible sources of confusion as possible. Dataset in total contains **6 identities** and over **1.8K frames**.

In the first scene, three identities enter and walk around while crossing trajectories. After a couple of seconds they leave the scene, each by different exit.

In the second scene, fourth identity, which will not reappear, is introduced to the gallery under slightly different lightning conditions.

In the last scene, five identities, with three of them reappearing from the first scene, enter the scene from one entrance. This entrance is close to exiting point of one of the identities in the first scene, but far from exiting points of the other reappearing identities. These five identities then walk around the scene, generating overlaps and occlusions and then leave the scene by the same place they entered.

6.3 Algorithm comparisons

In this section we compare performances of our proposed algorithm and the *FairMOT* algorithm on our testing set. For evaluation we use metrics **MOTA**, **IDSW**, **FP** and **FN** as defined in section 1.3.

We evaluate two versions of the proposed algorithm. In first version we do not save feature vectors from overlapped detections and use these vectors solely for associating.

In the second version we treat overlapped detections identically to unoverlapped ones and save feature vectors created from every matched detection.

For detector and feature extractor, we have chosen *Yolo-v4* and *OSNet* respectively. Results of this experiment can be seen in table 6.2 with highlighted results being the best.

	MOTA ↑	**IDSW** ↓	**FP** ↓	**FN** ↓
FairMOT association	0.907	41	61	**503**
Our association	**0.910**	9	**31**	544
Our assoc. + filtered overlaps	**0.910**	**7**	**31**	550

Table 6.2: Comparison of mentioned association algorithms with *Yolo-v4* and *OSNet* (↓ signals lower value is better and ↑ signals higher value is better).

We also include a variant of this experiment with *FairMOT* used for both object detection and extraction to make sure that there are not aspects of the *FairMOT* association specific to the *FairMOT* model. Results of this variant can be seen in table 6.3.

	MOTA ↑	IDSW ↓	FP ↓	FN ↓
FairMOT association	0.916	23	100	424
Our association	0.918	17	97	419
Our assoc. + filtered overlaps	**0.920**	**16**	**91**	**416**

Table 6.3: Comparison of mentioned association algorithms with *FairMOT* (↓ signals lower value is better and ↑ signals higher value is better).

As seen in both of these tables, our approach offers a major improvement over *FairMOT* approach on the benchmark simulating the target enviroment in terms of lowering the number of identity switches and false positives. While number of false negatives tends to be the same or even higher in some cases, as mentioned in chapter 5, we do not consider false negatives as important objective to our task.

From the results we can also see that the method of filtering overlapped detections tends to provide a minor improvement towards the performance of the system.

6.4 Optimal models

In this section, we compare several model combinations of object detectors and feature encodes to find the optimal model pair for our target enviroment.

We consider *CenterNet* and *Yolo-v4* for detectors and *ABDNet*, *OSNet*, with its variance *OSNet-AIN* and *PCB* for feature encoders. For evaluation we use metrics **MOTA**, **MOTP**, **IDSW** and **FP** and for association, we use our approach with the filtering of overlapped detections. Results of these comparison can be seen in table 6.4 with highlighted results being the best.

	MOTA ↑	MOTP ↓	IDSW ↓	FP ↓
FairMOT	**0.920**	0.175	16	91
Yolo-v4 + PCB	0.910	0.181	13	**25**
Yolo-v4 + ABDNet	0.910	0.181	14	31
Yolo-v4 + OSNet	0.910	0.181	7	31
Yolo-v4 + OSNet-AIN	0.917	0.184	**2**	27
CenterNet + PCB	0.870	0.173	22	130
CenterNet + ABDNet	0.869	**0.171**	30	130
CenterNet + OSNet	0.851	0.175	18	195
CenterNet + OSNet-AIN	0.851	0.174	27	175

Table 6.4: Comparison of mentioned association algorithms with *FairMOT* (↓ signals lower value is better and ↑ signals higher value is better).

From the results of the experiments we can see, that *Yolo-v4* tends to

perform better in terms of accuracy, while *CenterNet* tends to be slightly more precise. In terms of feature encoders, we can see that lowest count of identity switches is provided by *OSNet* with notable performance of its variant *OSNet-AIN* in synergy with *Yolo-v4*.

While it is noteworthy that *FairMOT* achieved the best results in terms of **MOTA**, from its higher values of **IDSW** and **FP** we can see that this is due to lower number of false negatives, which as we established before, is the least important type of error for our objective.

Since identity switches is the most important type of error for our objective, due to its long-term effects, we can consider combination of *Yolo-v4* and *OSNet-AIN* as the best combination for our target enviroment.

Conclusion

The goal of this thesis was designing and implementing a system that is able to track people in a video sequence by creating unique identities extracted from their image characteristics.

This system also estimates the age and gender of every detected person and stores this information to output statistics about the people encountered in the sequence.

In the first three chapters, we introduce the theoretical background needed to explore this task and conduct an analysis of approaches in this field.

After that we conducted an analysis of the target environment and based on this analysis we collected requirements on the algorithm.

In the next chapter, we introduce the designed algorithm, based on established approaches with modified association step and with additional features of age and gender estimation and pairing of face and person detections. The proposed algorithm uses three layers of matching, based on appearance information and distance.

The algorithm, as opposed to established approaches, also adds the feature of filtering out embeddings from overlapped detections to capture embeddings without visual noise and has a parametrizable way of filtering out detections in undesirable parts of the frames.

In the last chapter, we present results from training of our detectors to achieve detection of both head and person bounding boxes. We also present a benchmark simulating the target environment on which then we conduct experiments to compare the performance of our algorithm. We also conduct experiments to try to find the best possible combination of detectors and feature extractors for our target environment.

Our proposed approach can be improved upon mainly in the field of age and gender estimation by the following factors. Since our detectors are trained to detect bounding boxes of heads, it detects the head even when the person is turned away from the camera, thus sometimes giving misleading information to the age and gender estimator. We can improve upon this by, for example,

using a way to filter only faces from our head detections proposals.

Another way we can improve our algorithm is to parallelize age and gender estimation because since it is not necessary for our tracking problem, it can be done separately from the tracking loop, thus improving the speed of the algorithm.

Bibliography

[1] Wang, Z.; Zheng, L.; et al. Towards Real-Time Multi-Object Tracking. 2020. Available from: https://www.ecva.net/papers/eccv_2020/papers_ECCV/papers/123560103.pdf

[2] Zhang, Y.; Wang, C.; et al. FairMOT: On the Fairness of Detection and Re-Identification in Multiple Object Tracking. *arXiv preprint arXiv:2004.01888*, 2020.

[3] Jiao, L.; Zhang, F.; et al. A Survey of Deep Learning-Based Object Detection. *IEEE Access*, volume 7, 2019: pp. 128837–128868, doi: 10.1109/ACCESS.2019.2939201.

[4] Ren, S.; He, K.; et al. Faster R-CNN: Towards Real-Time Object Detection with Region Proposal Networks. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, volume 39, 06 2015, doi:10.1109/TPAMI.2016.2577031.

[5] Wang, C.-Y.; Mark Liao, H.-Y.; et al. CSPNet: A New Backbone that can Enhance Learning Capability of CNN. In *2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops (CVPRW)*, 2020, pp. 1571–1580, doi:10.1109/CVPRW50498.2020.00203.

[6] Zhou, X.; Wang, D.; et al. Objects as Points. 04 2019.

[7] Bochkovskiy, A. Yolo-v4. https://alexeyab84.medium.com/yolov4-the-most-accurate-real-time-neural-network-on-ms-coco-dataset-73adfd3602fe, 2020.

[8] Crowdhuman dataset [online]. https://www.crowdhuman.org, 2018.

[9] Zheng, L.; Yang, Y.; et al. Person Re-identification: Past, Present and Future. *arXiv preprint arXiv:1610.02984*, 10 2016.

[10] Ye, M.; Shen, J.; et al. Deep Learning for Person Re-identification: A Survey and Outlook. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, volume PP, 01 2021: pp. 1–1, doi:10.1109/TPAMI.2021.3054775. Available from: https://www.researchgate.net/publication/348799269_Deep_Learning_for_Person_Re-identification_A_Survey_and_Outlook

[11] Sun, Y.; Zheng, L.; et al. *Beyond Part Models: Person Retrieval with Refined Part Pooling (and A Strong Convolutional Baseline): 15th European Conference, Munich, Germany, September 8-14, 2018, Proceedings, Part IV*. 09 2018, ISBN 978-3-030-01224-3, pp. 501–518, doi:10.1007/978-3-030-01225-0_30. Available from: https://www.ecva.net/papers/eccv_2018/papers_ECCV/papers/Yifan_Sun_Beyond_Part_Models_ECCV_2018_paper.pdf

[12] Zhou, K.; Yang, Y.; et al. Omni-Scale Feature Learning for Person Re-Identification. In *ICCV*, 2019.

[13] Chen, T.; Ding, S.; et al. ABD-Net: Attentive but Diverse Person Re-Identification. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, October 2019.

[14] Shao, S.; Zhao, Z.; et al. CrowdHuman: A Benchmark for Detecting Human in a Crowd. 04 2018.

[15] Milan, A.; Leal-Taixe, L.; et al. MOT16: A Benchmark for Multi-Object Tracking. *arXiv preprint arXiv:1603.00831*, 2016.

[16] Kuhn, H. W. The Hungarian Method for the Assignment Problem. *Naval research logistics quarterly*, 1955. Available from: http://bioinfo.ict.ac.cn/~dbu/AlgorithmCourses/Lectures/Lec10-HungarianMethod-Kuhn.pdf

[17] The Hungarian algorithm [online]. https://www.hungarianalgorithm.com/hungarianalgorithm.php.

[18] Bewley, A.; Ge, Z.; et al. Simple online and realtime tracking. In *2016 IEEE International Conference on Image Processing (ICIP)*, 2016, pp. 3464–3468, doi:10.1109/ICIP.2016.7533003.

[19] Kalman, R. E. A New Approach to Linear Filtering and Prediction Problems. *Transactions of the ASME–Journal of Basic Engineering*, 1960.

[20] Wojke, N.; Bewley, A.; et al. Simple online and realtime tracking with a deep association metric. In *2017 IEEE International Conference on Image Processing (ICIP)*, 2017, pp. 3645–3649, doi:10.1109/ICIP.2017.8296962.

[21] He, K.; Zhang, X.; et al. Deep Residual Learning for Image Recognition. In *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2016, pp. 770–778, doi:10.1109/CVPR.2016.90.

[22] Simonyan, K.; Zisserman, A. Very Deep Convolutional Networks for Large-Scale Image Recognition. In *International Conference on Learning Representations*, 2015.

[23] Bochkovskiy, A.; Wang, C.-Y.; et al. YOLOv4: Optimal Speed and Accuracy of Object Detection. 04 2020.

[24] Lin, T.-Y.; Maire, M.; et al. Microsoft COCO: Common Objects in Context. *CoRR*, volume abs/1405.0312, 2014. Available from: `http://dblp.uni-trier.de/db/journals/corr/corr1405.html#LinMBHPRDZ14`

[25] Su, C.; Zhang, S.; et al. Deep Attributes Driven Multi-Camera Person Re-identification. In *ECCV*, 2016.

[26] Li, X.; Wu, A.; et al. Adversarial Open-World Person Re-Identification. 07 2018. Available from: `https://openaccess.thecvf.com/content_ECCV_2018/papers/Xiang_Li_Adversarial_Open-World_Person_ECCV_2018_paper.pdf`

[27] Zhou, K.; Yang, Y.; et al. Learning Generalisable Omni-Scale Representations for Person Re-Identification. *TPAMI*, 2021.

[28] Zheng, L.; Shen, L.; et al. Scalable Person Re-Identification: A Benchmark. *CVPR*, 2015. Available from: `https://www.cv-foundation.org/openaccess/content_iccv_2015/papers/Zheng_Scalable_Person_Re-Identification_ICCV_2015_paper.pdf`

[29] Li, W.; Zhao, R.; et al. DeepReID: Deep Filter Pairing Neural Network for Person Re-Identification. 2014. Available from: `https://openaccess.thecvf.com/content_cvpr_2014/html/Li_DeepReID_Deep_Filter_2014_CVPR_paper.html`

[30] Wei, L.; Zhang, S.; et al. Person Transfer GAN to Bridge Domain Gap for Person Re-identification. In *2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2018, pp. 79–88, doi:10.1109/CVPR.2018.00016.

[31] B, N. K.; Salis, D. V. E. Survey of Techniques to Estimate the Age and Gender of A Person using Face Images. 2019. Available from: `https://www.ijert.org/survey-of-techniques-to-estimate-the-age-and-gender-of-a-person-using-face-images`

[32] Tan, M.; Le, Q. EfficientNet: Rethinking Model Scaling for Convolutional Neural Networks. In *Proceedings of the 36th International Conference on Machine Learning*, *Proceedings of Machine Learning Research*, volume 97, edited by K. Chaudhuri; R. Salakhutdinov, PMLR, 09–15 Jun 2019, pp. 6105–6114. Available from: http://proceedings.mlr.press/v97/tan19a.html

[33] Paszke, A.; Gross, S.; et al. PyTorch: An Imperative Style, High-Performance Deep Learning Library. In *Advances in Neural Information Processing Systems 32*, edited by H. Wallach; H. Larochelle; A. Beygelzimer; F. d Alché-Buc; E. Fox; R. Garnett, Curran Associates, Inc., 2019, pp. 8024–8035. Available from: http://papers.neurips.cc/paper/9015-pytorch-an-imperative-style-high-performance-deep-learning-library.pdf

[34] Abadi, M.; Agarwal, A.; et al. TensorFlow: Large-Scale Machine Learning on Heterogeneous Systems. 2015, software available from tensorflow.org. Available from: https://www.tensorflow.org/

[35] Bradski, G. The OpenCV Library. *Dr. Dobb's Journal of Software Tools*, 2000.

[36] QT framework [online]. https://www.qt.io, 2012.

[37] Zhou, K.; Xiang, T. Torchreid: A Library for Deep Learning Person Re-Identification in Pytorch. 2019.

[38] cheind. MOT metrics [online]. https://github.com/cheind/py-motmetrics, 2020.

Appendix A

Acronyms

re-ID Re-identification

JSON JavaScript Object Notation

MOT Multiple object tracking

SORT Simple Online Realtime Tracking

SDE Separate Detection and Embedding

JDE Joint Detection and Embedding

RPN Region Proposal Network

FPS Frames per second

AI Artificial intelligence

ML Machine learning

IoU Intersection Over Union

MOTA Multiple object tracking accuracy

MOTP Multiple object tracking precision

IDSW Number of Identity switches

PCB Part-based convolutional baseline

IN Instance Normalisation

ABDNet Attentive but Diverse Network

YOLO You only look once

Appendix B

Contents of enclosed CD

- `readme.txt` ..... the file with CD contents description
- `src` ..... the directory of source codes
  - `moteel` ..... implementation sources
  - `thesis` ..... the directory of LaTeX source codes of the thesis
- `text` ..... the thesis text directory
  - `thesis.pdf` ..... the thesis text in PDF format