People Tracking and Re-Identification from Multiple Cameras

by

Ergys Ristani

Department of Computer Science
Duke University

Date: _____

Approved:

_____
Carlo Tomasi, Supervisor

_____
Ronald Parr

_____
Pankaj Agarwal

_____
Jiri Matas

Dissertation submitted in partial fulfillment of the requirements for the degree of
Doctor of Philosophy in the Department of Computer Science
in the Graduate School of Duke University
2018

Abstract

People Tracking and Re-Identification from Multiple Cameras

by

Ergys Ristani

Department of Computer Science
Duke University

Date: _____

Approved:

_____
Carlo Tomasi, Supervisor

_____
Ronald Parr

_____
Pankaj Agarwal

_____
Jiri Matas

An abstract of a dissertation submitted in partial fulfillment of the requirements for the degree of Doctor of Philosophy in the Department of Computer Science in the Graduate School of Duke University
2018

Abstract

In many surveillance or monitoring applications, one or more cameras view several people that move in an environment. Multi-person tracking amounts to using the videos from these cameras to determine who is where at all times. The problem is very challenging both computationally and conceptually. On one hand the amount of video to process is enormous while near real-time performance is desired. On the other hand people's varying appearance due to lighting, occlusions, viewpoint changes, and unpredictable motion in blind spots make person re-identification challenging.

This dissertation makes several contributions to person re-identification and multi-person tracking from multiple cameras. We present a weighted triplet loss for learning appearance descriptors which addresses both problems uniformly, doesn't suffer from the imbalance between positive and negative examples, and remains robust to outliers. We introduce the largest tracking benchmark to date, DukeMTMC, and adequate performance measures that emphasize correct person identification. A correlation clustering formulation for associating person observations is then introduced which maximizes agreements on the evidence graph. We assemble a tracker called DeepCC that combines an existing person detector, hierarchical and online reasoning, our appearance features and correlation clustering association. DeepCC achieves increased performance on two challenging sequences from the DukeMTMC benchmark, and ablation experiments demonstrate the merits of individual components.

To my parents

Contents

List of Tables

List of Figures

List of Abbreviations

Abbreviations

ATA	Average Tracking Accuracy
AWTL	Adaptive Weighted Triplet Loss
BIP	Binary Integer Program
CC	Correlation Clustering
CDSC	Constrained Dominant Set Clustering
CNN	Convolutional Neural Network
DPM	Deformable Parts Model
FN	False Negative
FNR	False Negative Rate
FRG	Fragments
FP	False Positive
FPR	False Positive Rate
GMCP	Generalized Minimum Clique Problem
GMMCP	Generalized Maximum Multi-Clique Problem
HNM	Hard Negative Mining
IDFN	False Negative ID
IDFP	False Positive ID
IDF_1	Identification F_1-score
IDP	Identification Precision

IDR	Identification Recall
IDTN	True Negative ID
IDTP	True Positive ID
ILP	Integer Linear Program
IoU	Intersection over Union
LMP	Lifted Multicuts Problem
mAP	Mean Average Precision
MCBM	Minimum Cost Bipartite Matching
MCNF	Minimum Cost Network Flow
MCTA	Multiple Camera Tracking Accuracy
ML	Mostly Lost
MOTA	Multiple Object Tracking Accuracy
MT	Mostly Tracked
MTT	Multi-Target Tracking
MTMC	Multi-Target Multi-Camera
MTMCT	Multi-Target Multi-Camera Tracking
MP	Multicut Problem
MPT	Multi-Person Tracking
MWDPC	Maximum Weight Disjoint Paths Cover
MWIS	Maximum Weight Independent Set
OP	Object Purity
PT	Partially Tracked
PCNF	Pairwise Costs Network Flow
ReID	Re-Identification
SDP	Subgraph Decomposition Problem
TP	True Positive / Track Purity

Acknowledgements

Thank you to my advisor, Carlo Tomasi, for many years of support and guidance. Being Carlo's student has been a privilege. Carlo taught me to value high quality work whether it applies to research, teaching, or leadership, to think creatively and ask the right questions, strive for deep understanding rather than incremental progress, and develop a principled methodology for research.

Thank you to my committee members Ronald Parr and Pankaj Agarwal for their service, teaching, valuable discussions and feedback. Thank you to my external committee member Jiri Matas for his genuine and inspiring conversations, and for always accommodating my needs to his busy schedule.

Thank you to the Computer Science Department at Duke University for giving me the opportunity to be part of its graduate program and support me throughout.

Thank you to my fellow graduate students Abhinandan, Cassi and Mayuresh for being fun officemates, and to Susanna, Shuzhi, Hannah, Swarna, Ben, Branka and DRIVE members, for valuable discussions during reading group meetings.

Thank you to my friend and collaborator Francesco Solera for long and productive conversations, and to Anton Milan and Laura Leal-Taixé for their continued help with improving MOTChallenge.

Thank you to my internship mentors Mehmet Tek and Nick Colonnese for their guidance.

Thank you to my friends Lorin, Stefan, Seth, and the volleyball team for making

Durham an enjoyable place to live.

Thank you to all my teachers throughout the years for their dedication.

None of my accomplishments would have been possible without the support of my family. Thank you to my parents Vasil and Liliana for nurturing me with passion for knowledge, arts, and languages. Thank you to my loving sister Ardita for her continuous care.

Finally, thank you to the following agencies for providing financial support during my research: Duke University, the National Science Foundation (through NSF grants CCF-1513816, IIS-10-17017, IIS-154372 and IIS-1420894), and the Army Research Office (through ARO grants W911NF-16-1-0392 and W911NF-10-1-0387).

1

Introduction

Multi-Target Multi-Camera Tracking (MTMCT) aims to determine the position of every person at all times from video streams taken by multiple cameras. The resulting multi-camera trajectories enable applications including visual surveillance, *e.g.*, suspicious activity and anomaly detection, sport player tracking, crowd behavior analysis, and cashierless stores. A problem closely related to MTMCT is person re-identification (ReID). Given a query image of a person, the goal is to retrieve from a database of images taken by different cameras the images where the same person appears.

MTMCT is a notoriously difficult problem: Cameras are often placed far apart to reduce costs, and their fields of view do not overlap. This distance between cameras causes extended periods of occlusion and large changes of viewpoint and illumination across different fields of view. In addition, the number of people is typically not known in advance, and the amount of data to process is enormous. This dissertation aims to address five challenging aspects of MTMC tracking.

The first challenge is designing good appearance features that can re-identify the same person at different occurrences in the videos. Good appearance features should

be invariant to changes in viewpoint or illumination, and robust to occlusion and detector localization errors. There is a vast literature on the topic of modeling human appearance and recent success is based on supervised deep learning methods. In this dissertation we design a new hard-negative mining scheme that yields improved features, and is generic enough to be used in combination with other work. We also generalize a popular definition of training loss, and propose a scheme to assign weights to samples which makes training stable to outliers and leads to fast convergence.

The second challenge is evaluating MTMC trackers. Performance evaluation is by no means straightforward for three reasons: a) It is difficult to quantify the performance of a complex tracker using a single score, b) different applications require different metrics, and c) matching ground truth and computed trajectories is a combinatorial problem. Previous efforts in the literature have produced metrics that are mostly suitable for researchers to pinpoint the errors of their trackers, but can fail to compare different systems fairly and uniformly in terms of performance from a user's perspective. In this dissertation we introduce a set of measures that are conceptually simple, useful for both researchers and the end-users of tracking systems, interpretable, and fair.

The third challenge in MTMC tracking is meaningful benchmarking. A good benchmark should be representative of realistic scenarios, provide accurate annotations, contain sufficient data to enable training of machine learning models, and present a level of difficulty suitable for accelerating progress in the field. Existing MTMC datasets have very simple camera topology, few cameras, are often recorded in lab conditions, can contain scripted motion performed by actors, and the number of identities is limited. This dissertation addresses these benchmarking challenges by introducing the largest MTMCT benchmark to date called DukeMTMC. It contains high quality video from 8 cameras, is recorded outdoors on a university campus, was carefully annotated over one year, has non-trivial camera topology, and features

more identities than all existing datasets combined.

The fourth challenge pertains to formulating the data association problem, that is, determining whether person detections correspond to the same identity or not. The problem is combinatorial and the identities and their number are unknown. In early work, correspondence had been addressed for consecutive video frames through greedy or bipartite matching. Subsequent formulations extended the idea to multiple frames through generalizations of bipartite matching, *e.g.*, finding path covers of a graph. Nodes in the graph represent observations, and the edge between two nodes carries a weight that depends on the similarity or correlation between the two observations at the edge's endpoints. Observations covered by a path belong to the same identity. In this dissertation we present a stronger formulation that partitions a correlation graph into sets, one set per identity, and maximizes agreement within sets and minimizes disagreement between sets. The superiority of this formulation and robustness to inaccurate evidence is demonstrated empirically on recent benchmarks.

Finally, the overarching challenge addressed in this dissertation is scalability. Can we design an MTMC tracker that can accurately and efficiently track people in large amounts of video of unbounded duration? We design a processing pipeline that feeds state-of-the-art detections to our state-of-the-art problem formulation and appearance features. The pipeline (DeepCC) processes multiple video streams efficiently utilizing online batch processing, hierarchical reasoning, and good initializations to the data association formulation. DeepCC achieves near real-time performance and sets a new state of the art in MTMC tracking with significant accuracy improvement over previous work.

Throughout this dissertation the terminology Multi-Object Tracking (MOT), Multi-Person Tracking (MPT), Multi-Target Tracking (MTT), and Multi-Target Multi-Camera Tracking is used interchangeably. MTT will often be used in the context of single-camera tracking whereas MTMCT will be used in the context of

multiple cameras with disjoint views. When cameras have overlapping views, the literature defines the problem as Multi-View Tracking. Also, the objects/targets of interest refer to people, even though in some cases the community is interested in tracking vehicles or animals.

This dissertation is organized as follows. Chapter 2 discusses features of appearance and is based on Ristani and Tomasi (2018). Chapter 3 reviews performance evaluation in MTMC tracking and Chapter 4 describes the DukeMTMC benchmarks for tracking and re-identification. They are based on Ristani et al. (2016). Chapter 5 discusses data association formulations and the proposed pipeline for scalable MTMC tracking. It is based on Ristani and Tomasi (2014) and Ristani and Tomasi (2018). All the above chapters are self-contained. They first review related work and preliminary concepts, and they rely on their own notation and symbols. Chapter 6 gives an overview of future directions, and at last Chapter 7 provides a summary of our insights and conclusions in chronological order.

1.1 Summary of Contributions

In this dissertation we make the following contributions:

- We formulate MTMC tracking as a correlation clustering problem which improves over methods that restrict evidence only along paths.
- We propose an adaptive weighted triplet loss that, unlike fixed-weight variants, is both accurate and stable, and does not suffer from the imbalance between positive and negative examples.
- We propose an inexpensive hard-identity mining scheme that helps learn better appearance features.
- We introduce the largest available benchmark for MTMC tracking, DukeMTMCT,

which poses significant challenges to existing trackers and provides large amounts of annotated trajectories to enable deep learning.

- We introduce a set of performance evaluation measures that emphasize correct identification rather than the number of individual mistakes. They are simple, fair, and are more suitable to end users of MTMCT scenarios.
- We propose an efficient processing pipeline that achieves state of the art MTMCT performance.
- We provide new insights on the relation between tracking and ranking accuracy on the DukeMTMCT benchmark.
- We show experimentally that our features yield state-of-the-art results on both MTMCT and Re-ID tasks.
- We share lessons learned throughout the years of work that went into making this dissertation.

2

Appearance Features

Appearance is by far the most informative cue when tracking people over time, within and across cameras. However, two conflicting challenges make appearance modeling for multi-person tracking and re-identification hard. On one hand observations are *ambiguous* in that different people that look alike may be confused with each other. Conversely, changing lighting, viewpoint, occlusions and other circumstances may cause *variance of appearance* for a given person, which may not be recognized to be the same in different observations.

Section 2.1 describes work related to modeling person appearance. Section 2.2 describes a supervised approach for learning discriminative appearance features. It includes an Adaptive Weighted Triplet Loss and a Hard Negative Mining scheme. The experiments in Section 2.3 demonstrate the benefits of these techniques.

2.1 Related Work

Human appearance has been described by color (Cai and Medioni (2014); Chen et al. (2011a, 2015, 2011b); Das et al. (2014); Gilbert and Bowden (2006); Javed et al. (2008); Jiuqing and Li (2013); Kuo et al. (2010); Zhang et al. (2015a,b)) and tex-

ture descriptors (Cai and Medioni, 2014; Chen et al., 2015; Daliyot and Netanyahu, 2013; Kuo et al., 2010; Zhang et al., 2015a,b). Lighting variations are addressed through color normalization (Cai and Medioni, 2014), exemplar-based approaches (Chen et al., 2015), or brightness transfer functions learned with (Das et al., 2014; Javed et al., 2008) or without supervision (Chen et al., 2011a; Gilbert and Bowden, 2006; Zhang et al., 2015a,b). Discriminative power is improved by *saliency* information (Martinel et al., 2014; Zhao et al., 2013) or by *learning* features specific to body parts (Cai and Medioni, 2014; Chen et al., 2015, 2011b; Daliyot and Netanyahu, 2013; Das et al., 2014; Jiuqing and Li, 2013; Kuo et al., 2010), either in the image (Bedagkar-Gala and Shah, 2011, 2012; Cheng et al., 2011) or back-projected onto an articulated (Baltieri et al., 2013; Cheng and Cristani, 2014) or monolithic (Baltieri et al., 2015)) 3D body model.

The current state of the art in person re-identification relies mainly on deep learning and the residual network architecture of He et al. (2016). Unlike standard networks where feature maps in consecutive layers can live in very different spaces, the identity connection of the residual unit adds a prior that consecutive feature maps should be similar, forcing convolution layers only to learn the residual.

To combat over-fitting and improve performance, hard training examples are typically generated by performing data augmentation (Barbosa et al., 2017; Zhong et al., 2017). Zheng et al. (2017b) follow an alternate approach and use a Generative Adversarial Network for mining difficult examples.

Sun et al. (2017) introduce an SVD layer that learns independent features on each dimension of the descriptor. Zheng et al. (2017a) use a spatial alignment layer to compensate for localization errors of the bounding box. Use of attributes has also proven beneficial for complementing appearance features (Layne et al., 2012; Lin et al., 2017; Schumann and Stiefelhagen, 2017)). More recently attention mechanisms have been employed to focus the learning process only on relevant features

(Zhou et al. (2018); Xu et al. (2018); Si et al. (2018). Kalayeh et al. (2018) use human semantic parsing for further feature improvement. Almazan et al. (2018) perform extensive experiments and discuss best practices for achieving high accuracy in person re-identification. The current state of the art model of Wang et al. (2018) learns discriminative features at multiple levels of granularity.

There has also been research on loss functions for training. Several methods rely on the categorical cross-entropy loss by casting appearance learning as a multi-class classification problem (Li et al., 2017; Zhang et al., 2017a). Appearance features are then extracted from the layer preceding the dense layer with softmax activation. The constrastive loss of Hadsell et al. (2006) was also used in Siamese networks. It has fallen out of favor to the triplet loss of Schroff et al. (2015) which can learn a more discriminative space. The triplet loss has been further improved to perform hard example mining within the batch either for each anchor (Hermans et al. (2017); Mishchuk et al. (2017)), or for the most difficult pair of positives and negatives (Xiao et al. (2017)). A loss that uses quadruples instead of triples has also been proposed by Chen et al. (2017a). More recently it has become common practice to combine multiple losses in the optimization objective. Other loss functions include the center loss of Wen et al. (2016) and the L2-constrained softmax loss of Ranjan et al. (2017).

2.2 Feature Learning

In this section we show how to learn appearance features useful for both MTMC tracking and person re-identification. MTMCT and ReID differ subtly but fundamentally, because ReID *ranks* distances to a query while MTMCT *classifies* a pair of images as being co-identical or not, and their performance is consequently measured by different metrics: ranking performance for ReID, classification error rates for MTMCT. This difference would seem to suggest that appearance features used for the two problems must be learned with different loss functions. Ideally, the ReID

loss ought to ensure that *for any query* a the largest distance between a and a feature that is co-identical to it is smaller than the smallest distance between a and a feature that is not co-identical to it. This would guarantee correct feature ranking for any given query. In contrast, the MTMCT loss ought to ensure that the largest distance between *any two* co-identical features is smaller that the smallest distance between *any two* non co-identical features, to guarantee a margin between within-identity and between-identity distances.

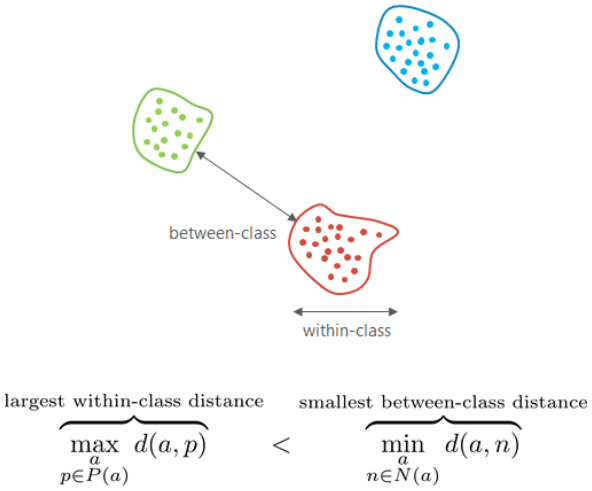


FIGURE 2.1: An ideal feature space where class samples are well-separated

With these criteria, zero MTMCT loss would imply zero Re-ID loss, but not *vice versa*. However, training with a loss of the MTMCT type is very expensive, because it would require using all pairs of features as input. More importantly, there would be a severe imbalance between the number of within-identity pairs and the much greater number of between-identity pairs. If the learned feature space satisfies the condition that features of different identities are *well-separated*, *i.e.*, the largest within class distance is smaller than the smallest between class distance, than a loss which enforces this condition can address appearance learning for both MTMCT and ReID jointly. An illustration is shown in Figure 2.1. In the next subsection we present such a loss.

2.2.1 Adaptive Weighted Triplet Loss

Given a large collection of labeled person snapshots we learn appearance features using an adaptive weighted triplet loss. For an anchor sample x_a, positive samples $x_p \in P(a)$ and negative samples $x_n \in N(a)$, we re-write the triplet loss in its most general form as:

$$L_3 = \left[m + \sum_{x_p \in P(a)} w_p d(x_a, x_p) - \sum_{x_n \in N(a)} w_n d(x_a, x_n) \right]_+ \tag{2.1}$$

where m is the given inter-person separation margin, d denotes distance of appearance, and $[\cdot]_+ = max(0, \cdot)$. The reformulation of the triplet loss from unweighted to weighted has two advantages. First it avoids the combinatorial process of triplet generation by using all the samples rather than a selection. Instead, the challenge of learning good features is to assign larger weights to difficult positive and negative samples. Second, the positive/negative class imbalance is easily handled by reflecting it in the weight distribution.

Hermans et al. (2017) and Mishchuk et al. (2017) have proposed the hard triplet loss with built-in hard sample mining. The hard triplet loss weights for Equation 2.1 are binary in their approach, as the loss considers only the most difficult positive and negative sample:

$$w_p = \left[x_p == \arg \max_{x \in P(a)} d(x_a, x) \right] \tag{2.2}$$

$$w_n = \left[x_n == \arg \min_{x \in N(a)} d(x_a, x) \right] \tag{2.3}$$

where $[\cdot]$ denotes the Iverson bracket. This loss gives better results than the original triplet loss of Schroff et al. (2015) with uniform weights because the latter washes

out the contribution of hard samples and is driven to worse local minima by easy samples. On the other hand, the uniformly weighted loss is more robust to outliers because they cannot affect the weights.

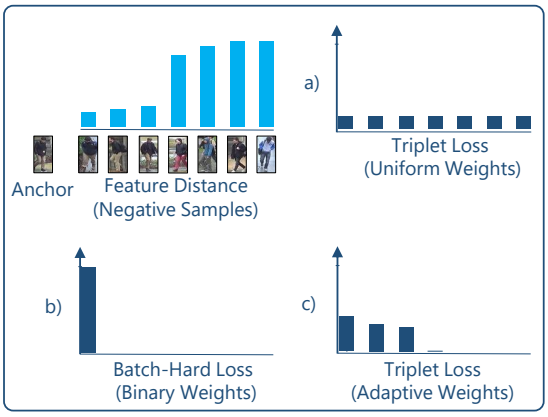


FIGURE 2.2: Triplet loss weighing schemes

Can we define weights such that L_3 converges to parameters at least as good as the hard triplet loss, yet remains robust to outliers? Our first improvement pertains to weights that achieve high accuracy and training stability *simultaneously*. Equations 2.2-2.3 assign full weight to the hardest positive/negative sample for each anchor while ignoring the remaining positive and negative samples. Instead, we assign adaptive weights using the softmax/min weight distributions as follows (see Figure 2.2):

$$w_p = \frac{e^{\frac{d(x_a,x_p)}{T}}}{\sum_{x \in P(a)} e^{\frac{d(x_a,x)}{T}}}, \quad w_n = \frac{e^{\frac{-d(x_a,x_n)}{T}}}{\sum_{x \in N(a)} e^{\frac{-d(x_a,x)}{T}}}. \tag{2.4}$$

where T is the temperature of the distribution, and defaults to 1. The adaptive weights in Equation 2.4 give little importance to easy samples and emphasize the most difficult ones. The lower the temperature, the more weight is assigned to the difficult samples. If the temperature instead was high, weights would approach the

uniform distribution. Lowering the temperature is useful when the number of samples increases.

When several difficult samples appear in a batch, they all get their fair share of the weight. This differs from the hard weight assignments of Equations 2.2-2.3 which give importance to the *single* most difficult sample. Adaptive weights are useful when the most difficult sample in a batch is an outlier, yet there exist other difficult samples to learn from. Experiments in such cases demonstrate the favorable properties of adaptive weights.

2.2.2 Hard Example Mining

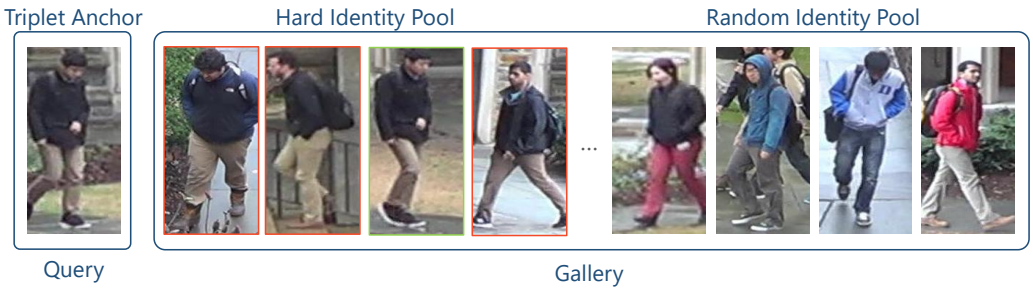


FIGURE 2.3: Hard identity mining: For each anchor identity, half of the $P-1$ identities in the batch are sampled from the hard identity pool, the other half from the random identity pool. Hard-negative identities (correct matches) are outlined in red (green).

For batch construction during training we leverage the idea of PK batches also introduced Hermans et al. (2017). In each batch there are K sample images for each of P identities. This approach has shown very good performance in similarity-based ranking and avoids the need to generate a combinatorial number of triplets. During a training epoch each identity is selected in its batch in turn, and the remaining $P-1$ batch identities are sampled at random. K samples for each identity are then also selected at random.

Our improvement is on the procedure that selects difficult identities. As the

size of the training set increases, sampling $P-1$ identities at random rarely picks the hardest negatives, thereby moderating batch difficulty. This effect can also be observed in the last few epochs of training, when many triplets within a batch exhibit zero loss.

To increase the chances of seeing hard negatives, we construct two sets to sample identities from. An example is shown in Figure 2.3. The hard identity pool consists of the H most difficult identities given the anchor, and the random identity pool consists of the remaining identities. Then in a PK batch of an anchor identity we sample the remaining $P-1$ identities from the hard or random identity pool with equal probability. This technique samples hard negatives more frequently and yet the batch partially preserves dataset statistics by drawing random identities. The pools can be constructed either after training the network for few epochs, or computed from a pre-trained network. We demonstrate the benefit of using the hard-identity mining scheme in the experiments section.

2.2.3 Data Augmentation

FIGURE 2.4: Examples of applying data augmentation to a person image

We augment the training images online with crops and horizontal flips to compensate for detector localization errors and to gain some degree of viewpoint/pose invariance. For illumination invariance we additionally apply contrast normalization, grayscale and color multiplication effects on the image. For resolution invariance we apply Gaussian blur of varying σ. For additional viewpoint/pose invariance we ap-

ply perspective transformations and small distortions. We additionally hide small rectangular image patches to simulate occlusion, similar to Zhong et al. (2017). We use the ImgAug library[1] and illustrate some examples in Figure 2.4.

2.3 Experiments

We run experiments on two person re-identification benchmarks to demonstrate the usefulness of the adaptive weighted triplet loss and hard negative mining. We report Rank-1 accuracy and mean average precision (mAP). Rank-1 accuracy measures the percentage of queries whose top retrieved image has the same identity as the query. mAP is the standard score used for comparing retrieval algorithms. It is the mean of the average precision scores for each query.

The first benchmark is Market 1501 introduced by Zheng et al. (2015). It is a large-scale person re-identification dataset with 1,501 identities observed by 6 near-synchronized cameras. The dataset was collected in the campus of Tsinghua University. It features 32,668 bounding boxes obtained using the deformable parts model detector. The dataset is challenging as the boxes are often misaligned and viewpoints can differ significantly. 751 identities are reserved for training and the remaining 750 for testing.

The second benchmark is DukeMTMC-reID. It features 1,404 identities appearing in more than two cameras and 408 identities who appear in only one camera are used as distractors. 702 identities are reserved for training and 702 for testing. More details on this dataset are discussed in Chapter 4.

We use the ResNet50 model of He et al. (2016) pre-trained on ImageNet and follow its *pool5* layer by a dense layer with 1024 units, batch normalization, and ReLU. Another dense layer yields 128-dimensional appearance features. We train the model with the adaptive weighted triplet loss, data augmentation, and hard-

[1] `https://github.com/aleju/imgaug`

identity mining. During training we set $P = 18$, $K = 4$, $m = 1$, $T = 1$, and the image is resolution 256×128. The learning rate is $3 \cdot 10^{-4}$ for the first 15000 iterations, and decays to 10^{-7} at iteration 25000. In experiments with hard identity mining we construct the hard and random pools once with features obtained at iteration 5000, then sample identities from these pools until the last iteration. The hard identity pool size H is set to 50 and we found that similar scores were obtained with 30-100 identities (4%-15% of all training identities). Extreme sizes yield little gain: A size of 1 contains a single hard identity which can be an outlier, a large HN pool nears random sampling.

2.4 Results

Table 2.1: Re-ID results on Market-1501

		Euclidean		SqEuclidean	
Author	Method	mAP	rank-1	mAP	rank-1
Zhang et al. (2016)	DNS	29.87	55.43	-	-
Varior et al. (2016)	GatedSiamese	39.55	65.88	-	-
Zhou et al. (2017)	PointSet	44.27	70.72	-	-
Barbosa et al. (2017)	SomaNet	47.89	73.87	-	-
Zheng et al. (2017a)	PAN	63.35	82.81	-	-
Hermans et al. (2017)	TriHard	66.63	82.99	64.47	82.01
Ristani and Tomasi (2018)	**AWTL**	**68.03**	**84.20**	**65.95**	**82.16**
Hermans et al. (2017)	TriHard (+Aug)	69.57	85.14	68.92	84.12
Ristani and Tomasi (2018)	**AWTL (+Aug)**	**70.83**	**86.11**	**69.64**	**84.71**
Hermans et al. (2017)	TriHard (+Aug+HNM)	71.13	86.40	0.16	0.36
Ristani and Tomasi (2018)	**AWTL (+Aug+HNM)**	**71.76**	**86.94**	**70.19**	**85.39**
Chen et al. (2017b)	DPFL (1-stream)	66.50	85.70	-	-
Chen et al. (2017b)	DPFL (2-stream)	72.60	88.06	-	-
Ristani and Tomasi (2018)	**AWTL (2-stream)**	**75.67**	**89.46**	**74.81**	**87.92**

Our Re-ID results for similarity-based ranking are shown in Tables 2.1 and 2.2. Scores are averages of five repetitions and no test-time augmentation is used. (a) Our Adaptive Weighted Triplet Loss (AWTL) consistently improves over the hard triplet loss of Hermans et al. (2017) and Mishchuk et al. (2017). (b) When training with square Euclidean distance to emphasize sensitivity to outliers our loss is robust

Table 2.2: Re-ID results on DukeMTMC-ReID

Author	Method	Euclidean mAP	Euclidean rank-1	SqEuclidean mAP	SqEuclidean rank-1
Zheng et al. (2015)	BoW+KISSME	12.17	25.13	-	-
Liao et al. (2015)	LOMO+XQDA	17.04	30.75	-	-
Zheng et al. (2016b)	Baseline	44.99	65.22	-	-
Zheng et al. (2017a)	PAN	51.51	71.59	-	-
Sun et al. (2017)	SVDNet	56.80	76.70	-	-
Hermans et al. (2017)	TriHard	54.60	73.24	0.28	0.89
Ristani and Tomasi (2018)	**AWTL**	**54.97**	**74.23**	**52.37**	**71.45**
Hermans et al. (2017)	TriHard (+Aug)	56.65	74.91	0.48	1.25
Ristani and Tomasi (2018)	**AWTL (+Aug)**	**57.28**	**75.31**	**55.94**	**75.04**
Hermans et al. (2017)	TriHard (+Aug+HNM)	54.90	74.23	0.30	0.94
Ristani and Tomasi (2018)	**AWTL (+Aug+HNM)**	**58.74**	**77.69**	**57.84**	**76.21**
Chen et al. (2017b)	DPFL (1-stream)	48.90	70.10	-	-
Chen et al. (2017b)	DPFL (2-stream)	60.60	79.20	-	-
Ristani and Tomasi (2018)	**AWTL (2-stream)**	**63.40**	**79.80**	**63.27**	**79.08**

in all scenarios, whereas the hard triplet loss shows to be unstable on the Duke dataset. (c) The proposed hard identity mining scheme (HNM) is also beneficial, and our adaptive weighted loss is both accurate and stable with difficult batches. (d) We also compare against a recent method by Chen et al. (2017b) that combines two network streams for better performance. When employing a similar technique (2-stream ensemble) we improve our ranking accuracy further, demonstrating that our loss can be used in different architectures. Qualitative results for our method are shown in Figure 2.5.

FIGURE 2.5: Qualitative results for our learned appearance features on the Duke dataset. Images outlined in green share the identity of the query image, those outlined in red do not.

3

Performance Evaluation

Evaluating the performance of a tracker is not straightforward. Some measure definitions can be ambiguous, tracking scenarios can be diverse, and different end-users have different needs. It is not surprising that different evaluation paradigms have appeared in the literature as well as multiple application-specific measures. Some measures report on very detailed aspects of tracker performance and are useful for analysis. Others look at bottom line performance and are used for ranking and tracker comparisons. In principle it is impossible to establish one single measure of performance that satisfies end-users with different needs.

This chapter begins with a thorough review of existing evaluation measures in Section 3.1 and analyzes each measure in detail. The interested reader car refer to Chapter 4 of Betke and Wu (2016) for an additional treatment of existing measures. Milan et al. (2013) also give a thorough review of challenges in performance evaluation. Section 3.2 discusses a new set of measures that capture how well the tracker knows who is where over all frames, regardless of when or why mistakes occur. These identity measures are appropriate for applications where correct identification is crucial, *e.g.*, visual surveillance, and are relevant to end-users interested in bottom-line

tracker performance.

3.1 Existing Measures

3.1.1 Preliminaries

Evaluating a multi-target tracker is typically done in two main steps:

(a) Mapping true and computed identities

(b) Computing scores given the true/computed trajectories and their mapping

There exist two popular types of mapping procedures in the literature for step (a). The first type maps true and computed detections optimally and bijectively at each frame. An example is the CLEAR MOT procedure which was introduced in Bernardin and Stiefelhagen (2008) and remains by far the most popular mapping procedure. At each frame, it first tries to preserve the mapping of detections from the previous frame if the match is still valid, and otherwise it matches the remaining true and computed detections optimally through bipartite matching. The second type of procedure maps trajectories to each other rather than detections. We refer to this mapping as identity mapping. Examples include the mapping in Kasturi et al. (2009), Leichter and Krupka (2013), or in Smith et al. (2005).

For step (b), a set of basic error types are accumulated through a scoring function. Below is a summary of a few error types and preliminary concepts.

The *match condition* requires a measure of distance or similarity between a true and a computed detection, and a threshold to decide if a match has occurred. For 2D image-plane evaluation, most benchmarks measure similarity by the intersection over union (IoU) ratio thresholded at 1/2. For evaluation in 3D world coordinates, it is common to use Euclidean distance between two positions on a reference ground plane, in meters, thresholded at 1 meter. Evaluators need to be careful when using

ground plane evaluation in 3D world coordinates: Due to detector localization errors the estimated 3D position of a person is more error-prone for far-away people.

The number of *false positives* fp_t is the number of times the tracker detects a target in frame t where there is none in the ground truth, the number of *false negatives* fn_t is the number of true targets missed by the tracker in frame t, and tp_t is the number of *true positive* detections at time t. The capitalized versions TP, FP, FN are the sums of tp_t, fp_t, and fn_t over all frames (and cameras, if more than one).

- A *fragmentation* occurs in frame t if the tracker switches the identity of a trajectory in that frame, but the corresponding ground-truth identity does not change. The number of fragmentations at frame t is ϕ_t, and $\Phi = \sum_t \phi_t$. Here and elsewhere, time subscripts for different cameras are considered to be different.

- A *merge* is the reverse of a fragmentation: The tracker merges two different ground truth identities into one between frames t' and t. The number of merges at frame t is γ_t, and $\Gamma = \sum_t \gamma_t$.

- A *mismatch* is either a fragmentation or a merge. The number of mismatches is $\mu_t = \phi_t + \gamma_t$, and $M = \sum_t \mu_t$.

When relevant, each of these error counts is given a superscript w (for "within-camera") when the frames t' and t in question come from the same camera, and a superscript h (for "handover") otherwise.

FIGURE 3.1: Two outputs that are equally good according to MOTA, yet the second tracker has better identification ability by explaining nearly all of the ground truth identity 'A' through computed identity 'b'.

3.1.2 Multiple Object Tracking Accuracy

Single-camera, multi-object tracking performance is typically measured by the Multiple Object Tracking Accuracy (MOTA):

$$\text{MOTA} = 1 - \frac{FN + FP + M}{T} \,. \tag{3.1}$$

MOTA penalizes detection errors ($FN + FP$) and mismatches (M) normalized by the total number T of true detections. If extended to the multi-camera case, MOTA and its companions under-report across-camera errors, because a trajectory that covers n_f frames from n_c cameras has only about n_c across-camera detection links between consecutive frames and about $n_f - n_c$ within camera ones, and $n_c \ll n_f$. The score lives in the range $(-\infty, 1]$ and relies on the CLEAR MOT mapping.

End-users interested in identity preservation do not find MOTA satisfactory. An example is given if Figure 3.1. The two trackers make one mismatch error and MOTA considers these trackers equally good. The end-users however expect the second tracker to be superior because it preserves the identity longer. There are scenarios in

which certain end-users find MOTA more suitable, for example in tracking the flow of ants. In this scenario MOTA is more suitable for evaluating tracklets that capture the general flow of motion, without heavily penalizing fragmentation errors.

A less discussed issue about MOTA is fairness. There has been no example in the literature to disprove that the optimal frame-by-frame matching gives the highest possible MOTA score to a tracker, neither a formal proof to show that the CLEAR MOT mapping procedure is globally optimal. While we believe the latter to be true, such proof goes beyond the scope of this dissertation.

3.1.3 Multi Camera Tracking Accuracy

Cao et al. (2015) introduced the multi-camera object tracking accuracy (MCTA) which condenses all aspects of system performance into one measure:

$$\text{MCTA} = \underbrace{\frac{2PR}{P+R}}_{F_1} \underbrace{\left(1 - \frac{M^w}{T^w}\right)}_{\text{within camera}} \underbrace{\left(1 - \frac{M^h}{T^h}\right)}_{\text{handover}} . \tag{3.2}$$

This measure multiplies the F_1 detection score (harmonic mean of precision and recall) by a term that penalizes within-camera identity mismatches (M^w) normalized by true within-camera detections (T^w) and a term that penalizes wrong identity handover mismatches (M^h) normalized by the total number of handovers. Consistent with our notation, T^h is the number of true detections (true positives TP^h plus false negatives FN^h) that occur when consecutive frames come from different cameras.

Comparing to MOTA, MCTA multiplies within-camera and handover mismatches rather than adding them. In addition, false positives and false negatives, accounted for in precision and recall, are also factored into MCTA through a product. This separation brings the measure into the range $[0, 1]$ rather than $[-\infty, 1]$ as for MOTA. However, the reasons for using a product rather than some other form of combination are unclear, and the measure has no intuitive interpretation. In particular, each error

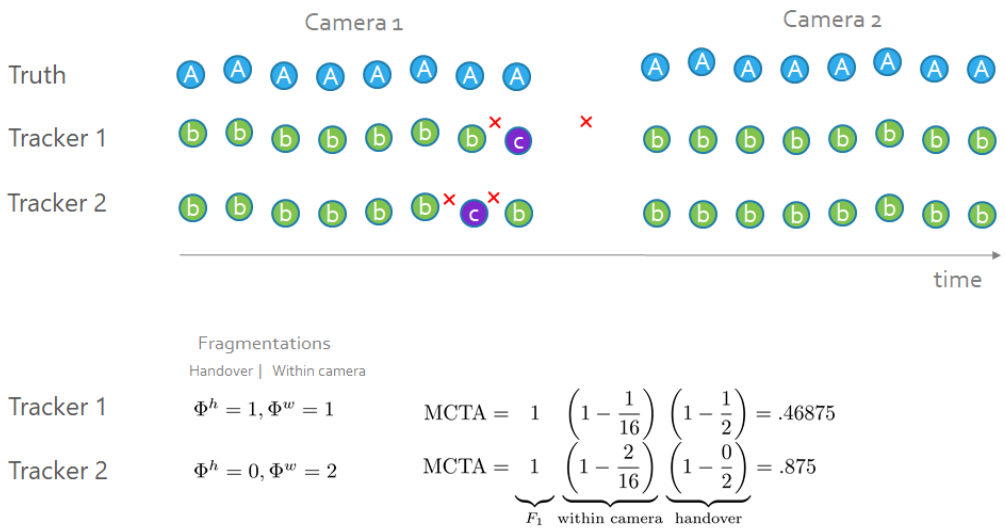


FIGURE 3.2: An example of how MCTA is very sensitive to the mapping of detections at the entry/exit frames. MCTA favors the second tracker much more, even though the two trackers are equally good at explaining identity.

in any of the three terms is penalized inconsistently, in that its cost is multiplied by the (variable) product of the other two terms. This makes MCTA very sensitive to the mapping of detections at entry/exit frames. An example is given in Figure 3.2. Even though the two trackers explain the ground truth identity equally well, MCTA favors the second tracker.

3.1.4 Handover Errors

Handover errors were introduced in Kuo et al. (2010) and capture only errors across cameras. These errors distinguish between fragmentations Φ^h and merges Γ^h. Fragmentations and merges are divided further into crossing (Φ^h_X and Γ^h_X) and returning (Φ^h_R and Γ^h_R) errors. These more detailed handover error scores help understand different types of tracker failures, and within-camera errors are quantified separately by standard measures.

A drawback of handover errors is their sensitivity to the detections' mapping at entry/exit frames. An example is given in Figure 3.3.

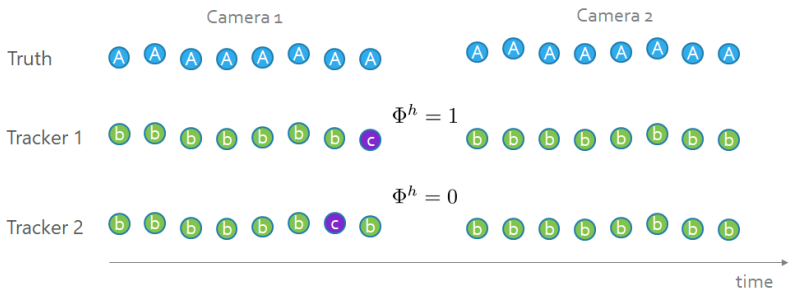


FIGURE 3.3: Handover errors are very sensitive to how the detections are mapped in the entry/exit frames of the camera transition.

3.1.5 PR-MOTA

PR-MOTA was introduced by Wen et al. (2015) to evaluate tracker behavior under different input detections. Specifically, object detections are generated for different values of precision and recall, the tracker is run on these detections, and the MOTA score is computed for the precision-recall pair. PR-MOTA is then the integral of MOTA scores along the PR curve. As the authors show, the concept is applicable to other measures, and helps understand how robust a tracker is when the quality of the input changes.

3.1.6 Track Length/Area Curve

Solera et al. (2015) also argue that evaluating trackers under one set of detections is not sufficient. They propose a reproducible procedure to generate a set of input detections of different quality. The procedure begins with ground truth detections and degrades them through false positives, false negatives, bounding box resizing, and occlusions. Trackers are then evaluated for all detection sets.

They also propose two measures, the Track Length (TL) Curve and Track Area (TA) Curve. In these plots, a point (x, y) indicates how many tracks x were correctly and continuously tracked for at least y frames.

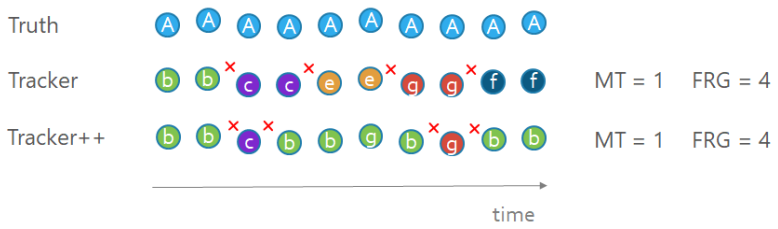


FIGURE 3.4: The popular trajectory score MT is oblivious to correct identification. Similarly, FRG doesn't account for identity recovery. Even though the second tracker is better at re-identifying the ground truth, these measures rank both trackers equally.

3.1.7 Trajectory Scores

Wu and Nevatia (2006) were interested in the quality of tracker produced trajectories and introduced four scores: Mostly Tracked (MT), Partially Tracked (PT), Mostly Lost (ML), and Fragments (FRG). MT (PT, ML) is the number of ground truth trajectories with true positives in the range 80%-100% (20%-80%, 0%-20%). FRG is the number of times that ground truth trajectories are interrupted in their mapping to tracker output. These measures have been popular in single-camera multi-target tracking.

In the context of these measures *tracked* means covered by detections regardless of their identity, so end-users interested in how well identities are explained do not find these measures satisfactory. In addition FRG ignores identities that switch back to the correct ground truth label. Examples are given in Figure 3.4.

3.1.8 Average Tracking Accuracy

The Average Tracking Accuracy (ATA) was introduced in Kasturi et al. (2009) as part of the VACE metrics. It first computes an optimal mapping between true and computed identities, and then reports a single score, ATA, that captures all errors. ATA is very similar to the IDF_1 score. However, it differs by how the penalty of a mismatch is computed. ATA penalizes a mismatch by the distance between the

mismatched true and computed detection, rather than a binary penalty. Leichter and Krupka (2013) argue that this makes the score not error type differentiable, meaning that it is ambiguous whether ATA measures identification or localization quality.

3.1.9 Track and Object Purity

Smith et al. (2005) introduce a set of measures for evaluating a tracker's identification ability. Specifically, they are named Track Purity (TP) and Object Purity (OP). Track Purity is the fraction of correctly identified ground truth detections and Object Purity is the ratio of correctly identified computed detections. Which true track is mapped to which object, and vice-versa, is determined through a mapping procedure that relies on the majority vote of true positive detections. TP (OP) is similar to Identification Recall (Precision) discussed in the next section. However, the identity mapping of Smith et al. (2005) is not necessarily bijective, and maximizes no global objective.

3.1.10 Track Clustering Scores

Wu et al. (2017) introduce a set of measures that evaluates a tracker's identification ability during inter-camera association. These measures are similar to ours as they utilize precision, recall and F_1-score. However, the authors show that these measures correlate more with re-identification mean average precision (mAP). The main difference between this work and other measures, including ours, is that the measures capture the fraction of correctly identified *links* between detections, instead of correctly identified *detections*. The authors claim that this paradigm is more useful for inter-camera tracking evaluation.

3.1.11 Monotonic and Error Type Differentiable Measures

Leichter and Krupka (2013) argue that defining performance measures for multi-target tracking is an ill-posed problem in the sense that there is no single "correct"

or "best" set of measures. However, different measures should be *monotonic* in that the score should not decrease if an error is eliminated or a success is added. Measures should also be *error type differentiable* in that each measure accounts for errors of the same type. They define five measures that satisfy both of these properties, namely False Negative Rate (FNR), False Positive Rate (FPR), Fragmentations, Mergers, and Mean Deviation. All five measures are computed on top of a bijective mapping between true and computed identities that maximizes true positives, guaranteeing fairness in evaluation.

FNR is analogous to identification recall, and the mapping of Leichter and Krupka (2013) is the same as ours. FPR is conceptually different from identification precision because it normalizes mistakes by the sequence length and image area. Overall, our measures also contain the identification F_1-score which gives bottom-line performance for ranking trackers. Our measures are also both monotonic and error type differentiable.

3.1.12 Identity-Aware Measures

Yu et al. (2016b) introduce performance scores for the task of identity aware multi-target tracking. The problem is defined as computing trajectories for a known set of true identities from a database. This implies that the truth-to-result match is determined during tracking and not evaluation. Instead, our evaluation applies to the more general MTMC setting where the tracker is agnostic to the true identities.

3.2 Identity Measures

We propose to measure performance not by *how often* mismatches occur, but by *how long* the tracker correctly identifies targets. To this end, ground-truth identities are first matched to computed ones. More specifically, a bipartite match associates one ground-truth trajectory to exactly one computed trajectory by minimizing the num-

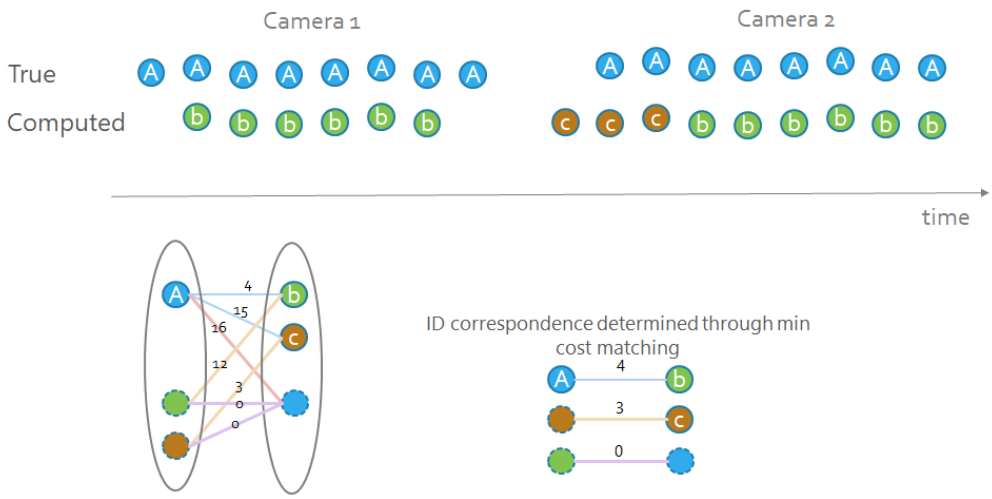


FIGURE 3.5: An illustration of the Truth-To-Result Match

ber of mismatched frames over all the available data—true and computed. Standard measures such as precision, recall, and F_1-score are built on top of this truth-to-result match. These scores then measure the number of mismatched or unmatched detection-frames, regardless of where the discrepancies start or end or which cameras are involved.

3.2.1 The Truth-To-Result Match

To compute the optimal truth-to-result match, we construct a bipartite graph $G = (V_T, V_C, E)$ as follows. Vertex set V_T has one "regular" node τ for each true trajectory and one "false positive" node f_γ^+ for each computed trajectory γ. Vertex set V_C has one "regular" node γ for each computed trajectory and one "false negative" node f_τ^-, for each true trajectory τ. Two regular nodes are connected with an edge $e \in E$ if their trajectories overlap in time. Every regular true node τ is also connected to its corresponding f_τ^-, and every regular computed node γ is also connected to its corresponding f_γ^+. An example is shown in Figure 3.5.

The cost on an edge $(\tau, \gamma) \in E$ tallies the number of false negative and false positive frames that would be incurred if that match were chosen. Specifically, let

$\tau(t)$ be the sequence of detections for true trajectory τ, one detection for each frame t in the set $\mathcal{T}_\tau$ over which τ extends, and define $\gamma(t)$ for $t \in \mathcal{T}_\gamma$ similarly for computed trajectories. The two simultaneous detections $\tau(t)$ and $\gamma(t)$ are a *miss* if they do not overlap in space, and we write

$$m(\tau, \gamma, t, \Delta) = 1 \ . \tag{3.3}$$

More specifically, when both τ and γ are regular nodes, spatial overlap between two detections can be measured either in the image plane or on the reference ground plane in the world. In the first case, we declare a miss when the area of the intersection of the two detection boxes is less than Δ (with $0 < \Delta < 1$) times the area of the union of the two boxes. On the ground plane, we declare a miss when the positions of the two detections are more than $\Delta = 1$ meter apart. If there is no miss, we write $m(\tau, \gamma, t, \Delta) = 0$. When either τ or γ is an irregular node (f_τ^- or f_γ^+), any detections in the other trajectory are misses. When both τ and γ are irregular, m is undefined. We define costs in terms of binary misses, rather than, say, Euclidean distances, so that a miss between regular positions has the same cost as a miss between a regular position and an irregular one. Matching two irregular trajectories incurs zero cost because they are empty.

With this definition, the cost on edge $(\tau, \gamma) \in E$ is defined as follows:

$$c(\tau, \gamma, \Delta) = \underbrace{\sum_{t \in \mathcal{T}_\tau} m(\tau, \gamma, t, \Delta)}_{\text{False Negatives}} + \underbrace{\sum_{t \in \mathcal{T}_\gamma} m(\tau, \gamma, t, \Delta)}_{\text{False Positives}} \ . \tag{3.4}$$

A minimum-cost solution to this bipartite matching problem determines a one-to-one matching that minimizes the cumulative false positive and false negative errors, and the overall cost is the number of mis-assigned detections for all types of errors. Every (τ, γ) match is a True Positive ID ($IDTP$). Every (f_γ^+, γ) match is a False

Positive ID ($IDFP$). Every (τ, f_τ^-) match is a False Negative ID ($IDFN$). Every (f_γ^+, f_τ^-) match is a True Negative ID ($IDTN$).

The matches (τ, γ) in $IDTP$ imply a *truth-to-result* match, in that they reveal which computed identity matches which ground-truth identity. In general not every trajectory is matched. The sets

$$MT = \{\tau \mid (\tau, \gamma) \in IDTP\} \quad \text{and} \quad MC = \{\gamma \mid (\tau, \gamma) \in IDTP\} \tag{3.5}$$

contain the *matched ground-truth trajectories* and *matched computed trajectories*, respectively. The pairs in $IDTP$ can be viewed as a bijection between MT and MC. In other words, the bipartite match implies functions $\gamma = \gamma_m(\tau)$ from MT to MC and $\tau = \tau_m(\gamma)$ from MC to MT.

3.2.2 Identification Precision, Identification Recall, and F_1 Score

We use the $IDFN$, $IDFP$, $IDTP$ counts to compute identification precision (IDP), identification recall (IDR), and the corresponding F_1 score IDF_1. More specifically,

$$IDFN = \sum_{\tau \in AT} \sum_{t \in \mathcal{T}_\tau} m(\tau, \gamma_m(\tau), t, \Delta) \tag{3.6}$$

$$IDFP = \sum_{\gamma \in AC} \sum_{t \in \mathcal{T}_\gamma} m(\tau_m(\gamma), \gamma, t, \Delta) \tag{3.7}$$

$$IDTP = \sum_{\tau \in AT} \text{len}(\tau) - IDFN = \sum_{\gamma \in AC} \text{len}(\gamma) - IDFP \tag{3.8}$$

where AT and AC are all true and computed identities in MT and MC.

$$IDP = \frac{IDTP}{IDTP + IDFP} \tag{3.9}$$

$$IDR = \frac{IDTP}{IDTP + IDFN} \tag{3.10}$$

$$IDF_1 = \frac{2\,IDTP}{2\,IDTP + IDFP + IDFN} \tag{3.11}$$

Identification precision (recall) is the fraction of computed (ground truth) detections that are correctly identified. IDF_1 is the ratio of correctly identified detections over the average number of ground-truth and computed detections. ID precision and ID recall shed light on tracking trade-offs, while the IDF_1 score allows ranking all trackers on a single scale that balances identification precision and recall through their harmonic mean.

This performance evaluation approach based on the truth-to-result match addresses all the weaknesses mentioned earlier in a simple and uniform way, and enjoys the following desirable properties: (1) *Bijectivity:* A correct match (with no fragmentation or merge) between true identities and computed identities is one-to-one. (2) *Optimality:* The truth-to-result matching is the most favorable to the tracker, maximizing IDF1. (3) *Consistency:* Errors of any type are penalized in the same currency, namely, the number of misassigned or unassigned frames. This approach also handles overlapping and disjoint fields of view in exactly the same way—a feature absent in all previous measures.

3.2.3 Measures of Handover Difficulty

Handover errors in current measures are meant to account for the additional difficulty of tracking individuals across cameras, compared to tracking them within a single camera's field of view. If a system designer were interested in this aspect of performance, a similar measure could be based on the difference between the total number of errors for the multi-camera solution and the sum of the numbers of single-camera errors:

$$E_M - E_S \quad \text{where} \quad E_M = IDFP_M + IDFN_M \quad \text{and} \quad E_S = IDFP_S + IDFN_S \; . \tag{3.12}$$

The two errors can be computed by computing the truth-to-result mapping twice: Once for all the data and once for each camera separately (and then adding the single-

camera errors together). The difference above is nonnegative, because the multi-camera solution must account for the additional constraint of consistency across cameras. Similarly, simple manipulation shows that ID precision, ID recall, and IDF_1 score are sorted the other way:

$$IDP_S - IDP_M \geqslant 0 \quad , \quad IDR_S - IDR_M \geqslant 0 \quad , \quad F_{1S} - F_{1M} \geqslant 0$$

and these differences measure how well the overall system can associate across cameras, given within-camera associations.

3.2.4 Additional Considerations

Comparison with CLEAR MOT. The first step in performance evaluation matches true and computed identities. In CLEAR MOT the event-based matching defines the best mapping sequentially at each frame. It minimizes Euclidean distances (within a threshold Δ) between unmatched detections (true and computed) while matched detections from frame $t-1$ that are still within Δ in t are preserved. Although the per-frame identity mapping is 1-to-1, the mapping for the entire sequence is generally many-to-many.

In our identity-based measures, we define the best mapping as the one which minimizes the total number of mismatched frames between true and computed IDs for the entire sequence. Similar to CLEAR MOT, a match at each frame is enforced by a threshold Δ. In contrast, our reasoning is not frame-by-frame and results in an ID-to-ID mapping that is 1-to-1 for the entire sequence.

The second step evaluates the goodness of the match through a scoring function. This is usually done by aggregating mistakes. MOTA aggregates FP, FN and M while we aggregate IDFP and IDFN counts. The notion of fragmentation is not present in our evaluation because the mapping is strictly 1-to-1. In other words our evaluation only checks whether every detection of an identity is explained or not,

consistently with our definition of tracking. Also, our aggregated mistakes are binary mismatch counts instead of, say, Euclidean distances. This is because we want all errors to be penalized in the same currency. If we were to combine the binary IDFP and IDFN counts with Euclidean distances instead of IDTP, the unit of error would be ambiguous: We won't be able to tell whether the tracker under evaluation is good at explaining identities longer or following their trajectories closer.

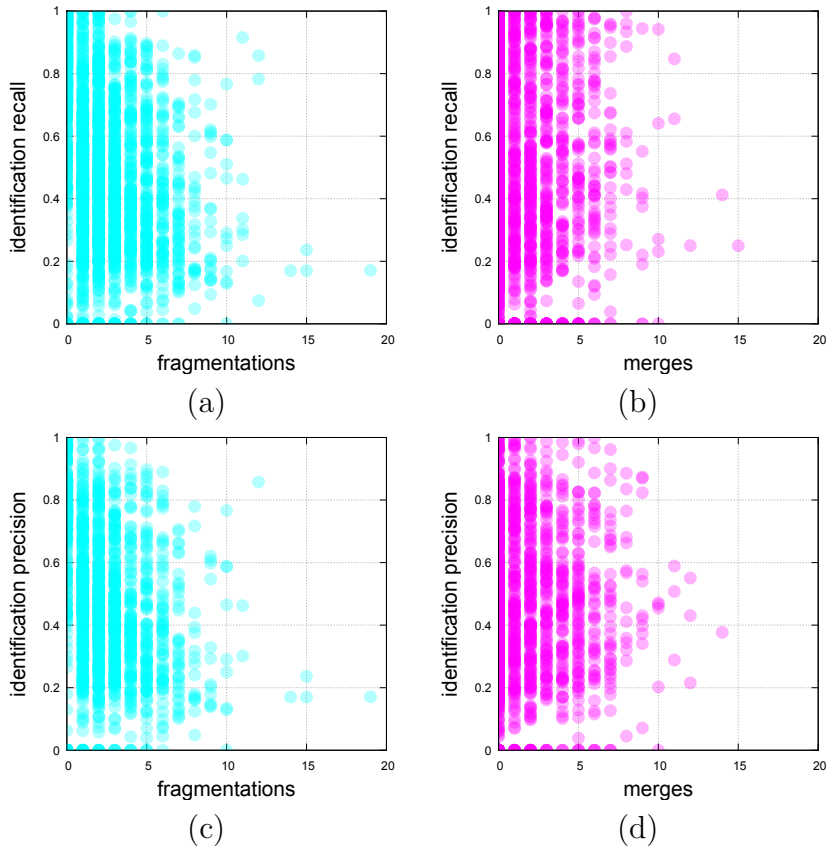
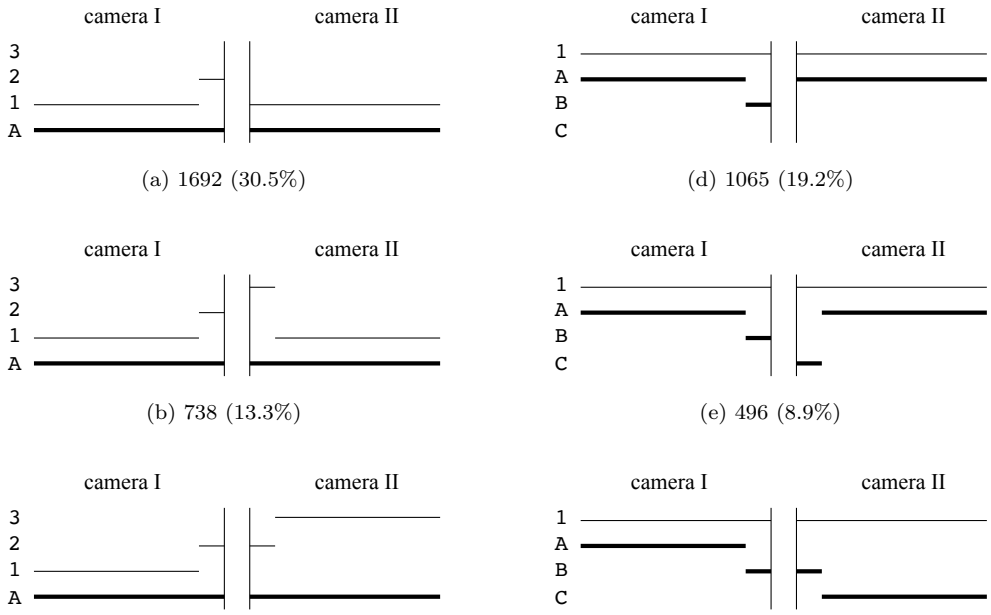


FIGURE 3.6: Scatter plots of ground-truth trajectory ID recall (a, b) and ID precision (c, d) versus the number of trajectory fragmentations (a, c) and merges (b, d). Correlation coefficients are -0.24, -0.05, -0.38 and -0.41. This confirms that event- and identity-based measures quantify different aspects of tracker performance.

ID Recall, ID Precision and Mismatches. Figure 3.6 shows that fragmentations and merges correlate poorly with ID recall and ID precision, confirming that event- and identity-based measures quantify different aspects of performance.



(a) 1692 (30.5%)

(d) 1065 (19.2%)

(b) 738 (13.3%)

(e) 496 (8.9%)

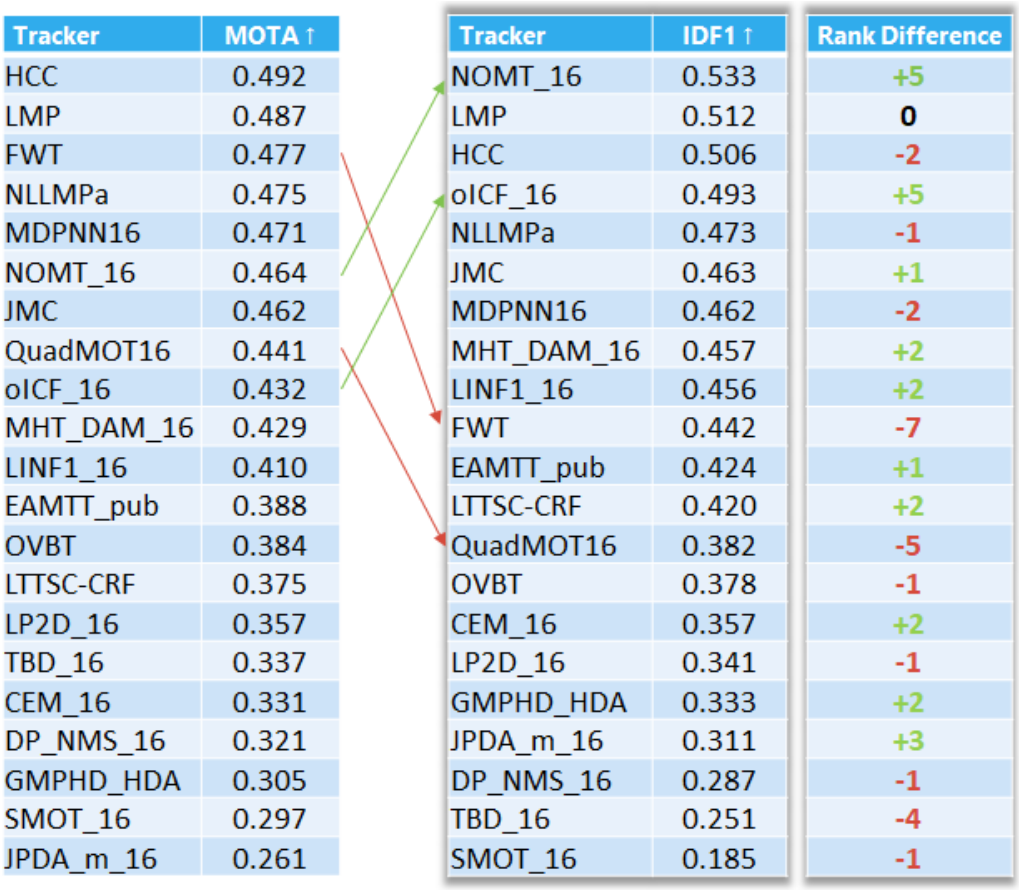
(c) 70 (1.3%)

(f) 58 (1.0%)

FIGURE 3.7: Thick lines represent true identities and thin lines computed identities. In about 74% (4,119 out of 5,549) of the handovers output by our reference system on the DukeMTMC data set, a short trajectory close to the handover causes a marked discrepancy between event-based, traditional performance measures and our identity-based measures. A handover fragmentation error (a, b) or merge error (d, e) is declared where the handover is essentially correct. A handover fragmentation error (c) or merge error (f) is not declared where the handover is essentially incorrect. Each caption shows the number of occurrences and the percentage of the total number of computed handovers.

Truth-to-Result Mapping. Traditional event-based performance measures handle handover errors differently from ours. Figure 3.7 shows that these discrepancies are frequent in our results.

Ranking by MOTA vs IDF$_1$. The popular MOTA score and our IDF_1 score rank trackers differently on the popular MOT16 dataset of Milan et al. (2016). Figure 3.8 shows that changes in ranking are frequent, and methods that do well according to MOTA are not necessarily the best at correct identification.



Tracker	MOTA ↑
HCC	0.492
LMP	0.487
FWT	0.477
NLLMPa	0.475
MDPNN16	0.471
NOMT_16	0.464
JMC	0.462
QuadMOT16	0.441
oICF_16	0.432
MHT_DAM_16	0.429
LINF1_16	0.410
EAMTT_pub	0.388
OVBT	0.384
LTTSC-CRF	0.375
LP2D_16	0.357
TBD_16	0.337
CEM_16	0.331
DP_NMS_16	0.321
GMPHD_HDA	0.305
SMOT_16	0.297
JPDA_m_16	0.261

Tracker	IDF1 ↑	Rank Difference
NOMT_16	0.533	+5
LMP	0.512	0
HCC	0.506	-2
oICF_16	0.493	+5
NLLMPa	0.473	-1
JMC	0.463	+1
MDPNN16	0.462	-2
MHT_DAM_16	0.457	+2
LINF1_16	0.456	+2
FWT	0.442	-7
EAMTT_pub	0.424	+1
LTTSC-CRF	0.420	+2
QuadMOT16	0.382	-5
OVBT	0.378	-1
CEM_16	0.357	+2
LP2D_16	0.341	-1
GMPHD_HDA	0.333	+2
JPDA_m_16	0.311	+3
DP_NMS_16	0.287	-1
TBD_16	0.251	-4
SMOT_16	0.185	-1

FIGURE 3.8: MOTA and IDF_1 rank the trackers differently on the popular MOT16 dataset of Milan et al. (2016). Trackers with the highest identification ability do not necessarily have the highest MOTA score.

4

The DukeMTMC Benchmarks

A good MTMCT benchmark data set should test trackers at least on bottom-line accuracy, efficiency, scalability, robustness to crowded scenes, changes in appearance, unconstrained camera transitions and extended occlusions. Section 4.1 discusses existing benchmarks and why they are unsatisfactory for a thorough performance evaluation. Section 4.2 describes the DukeMTMC data set, which aims to address these limitations and accelerate progress in the field. Section 4.3 discusses several benchmarks that build on DukeMTMC.

4.1 Related Data Sets

Existing multi-camera data sets allow only for limited evaluation of MTMC systems. Parameters of existing data sets are summarized in Table 4.1. Some have fully overlapping views and are restricted to short time intervals and controlled conditions. Some sports scenarios provide quality video with many cameras, but their environments are severely constrained and there are no blind spots between cameras. Data sets with disjoint views come either with low resolution video, a small number of cameras placed along a straight path, or scripted scenarios. Most importantly, all

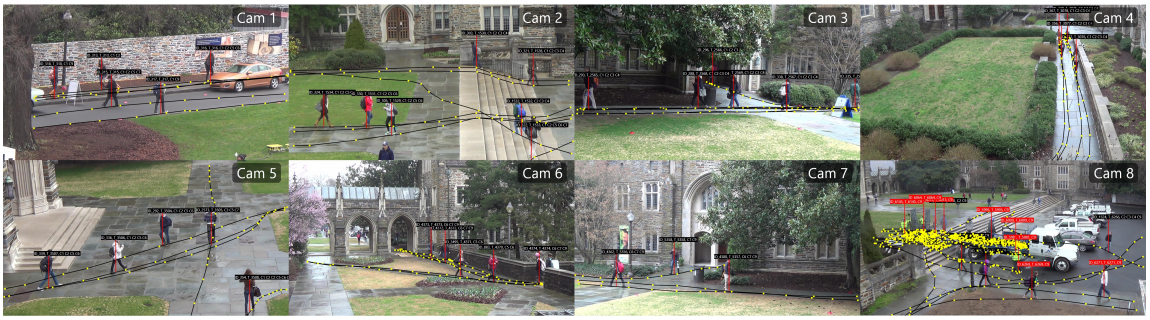


FIGURE 4.1: Camera views and annotations of the DukeMTMC data set

existing data sets only have a small number of identities. DukeMTMC is shown in the last row. It contains more identities than all previous data sets *combined*, and was recorded over the longest time period at the highest temporal resolution (60 fps).

4.2 DukeMTMC Data Set

DukeMTMC is a manually annotated, calibrated, multi-camera data set recorded outdoors on the Duke University campus with 8 synchronized cameras (Fig. 4.1). It consists of 6,791 trajectories for *2,834 different identities* (distinct persons) over *1 hour and 25 minutes for each camera*, for a total of more than 10 video hours and more than 2 million frames. There are on average 2.5 single-camera trajectories per identity, and up to 7 in some cases.

The cumulative trajectory time is more than 30 hours. Individual camera density varies from 0 to 54 people per frame, depending on the camera. There are 4,159 hand-overs and up to 50 people traverse blind spots at the same time. More than 1,800 self-occlusion events happen (with 50% or more overlap), lasting 60 frames on average. Videos are recorded at 1080p resolution and 60 fps to capture spatial and temporal detail. Two camera pairs (2-8 and 3-5) have small overlapping areas, through which about 100 people transit, while the other cameras are disjoint. Full annotations are provided in the form of trajectories of each person's foot contact point with the ground. Image bounding boxes are also available and have been

Table 4.1: Summary of existing data sets for MTMC tracking. Ours is in the last row.

Dataset	Author	IDs	Duration	Cams	Actors	Overlap	Blind Spots	Calib.	Resolution	FPS	Scene	Year
Laboratory	Fleuret et al. (2008)	3	2.5 min	4	Yes	Yes	No	Yes	320x240	25	Indoor	2008
Campus	Fleuret et al. (2008)	4	5.5 min	3	Yes	Yes	No	Yes	320x240	25	Outdoor	2008
Terrace	Fleuret et al. (2008)	7	3.5 min	4	Yes	Yes	No	Yes	320x240	25	Outdoor	2008
Passageway	Berclaz et al. (2011)	4	20 min	4	Yes	Yes	No	Yes	320x240	25	Mixed	2011
Issia Soccer	D'Orazio et al. (2009)	25	2 min	6	No	Yes	No	Yes	1920x1080	25	Outdoor	2009
Apidis Basket.	De Vleeschouwer et al. (2008)	12	1 min	7	No	Yes	No	Yes	1600x1200	22	Indoor	2008
PETS2009	Ferryman and Shahrokni (2009)	30	1 min	8	Yes	Yes	No	Yes	768x576	7	Outdoor	2009
NLPR MCT 1	Cao et al. (2015)	235	20 min	3	No	No	Yes	No	320x240	20	Mixed	2015
NLPR MCT 2	Cao et al. (2015)	255	20 min	3	No	No	Yes	No	320x240	20	Mixed	2015
NLPR MCT 3	Cao et al. (2015)	14	4 min	4	Yes	Yes	Yes	No	320x240	25	Indoor	2015
NLPR MCT 4	Cao et al. (2015)	49	25min	5	Yes	Yes	Yes	No	320x240	25	Mixed	2015
Dana36	Per et al. (2012)	24	N/A	36	Yes	Yes	Yes	No	2048x1536	N/A	Mixed	2012
USC Campus	Kuo et al. (2010)	146	25 min	3	No	No	Yes	No	852x480	30	Outdoor	2010
CamNeT	Zhang et al. (2015a)	50	30 min	8	Yes	Yes	Yes	No	640x480	25	Mixed	2015
DukeMTMC	Ristani et al. (2016)	2834	85 min	8	No	Yes	Yes	Yes	1920x1080	60	Outdoor	2016

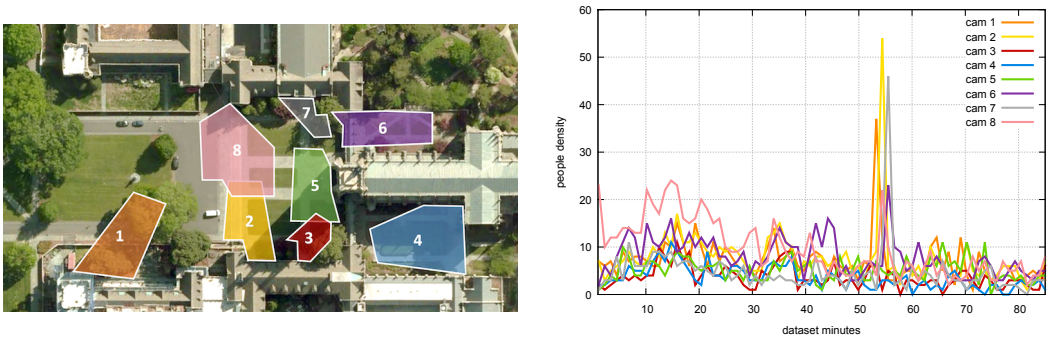


FIGURE 4.2: Camera placement with fields of view and people density over time

semi-automatically generated.

Unlike many multi-camera data sets, DukeMTMC is not scripted and cameras have a wider field of view. Unlike single-camera benchmarks where a tracker is tested on very short videos of different challenging scenarios, this data set is recorded in a fixed environment, and the main challenge is persistent tracking under occlusions and blind spots. Figure 4.2 shows the fields of view of the cameras (left) and the density variation of the dataset over time and across cameras (right).

People often carry bags, backpacks, umbrellas, or bicycles. Some people stop for long periods of time in blind spots and the environment rarely constrains their paths. So transition times through blind spots are often but not always informative. 891 people walk in front of only one camera—a challenge for trackers that are prone to false-positive matches across cameras. People detections and foreground masks are released along with the videos.

4.2.1 Annotation

The DukeMTMC annotation process was carried in three phases: a) single-camera trajectory annotation, b) multi-camera trajectory correspondence, and c) bounding box height annotation.

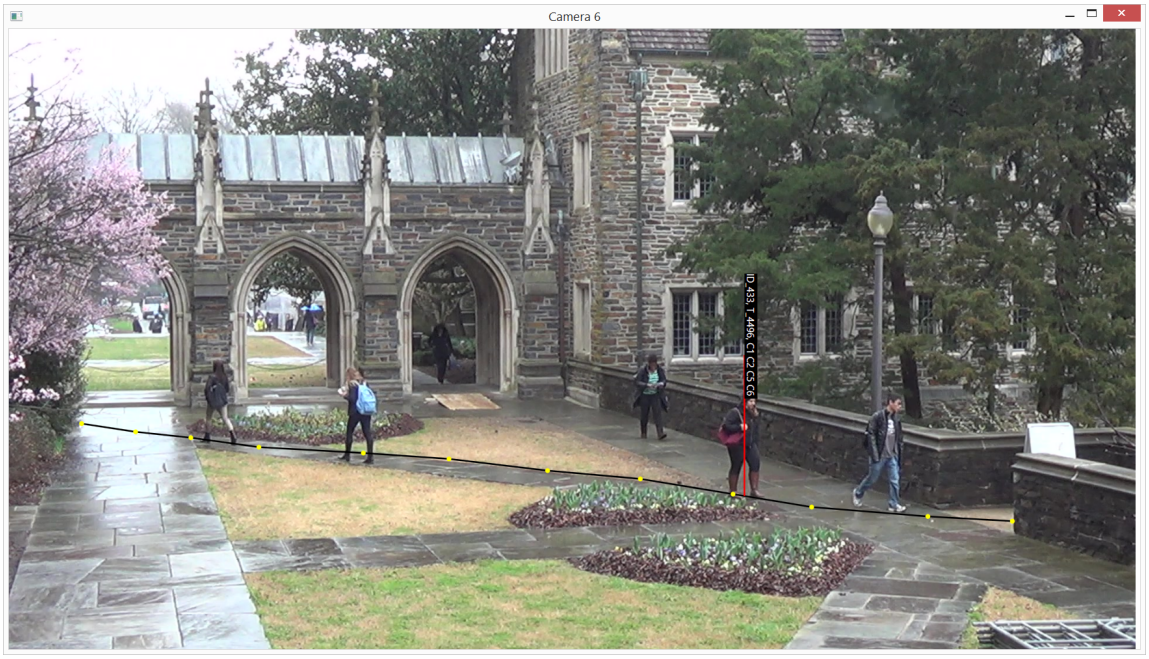


FIGURE 4.3: Annotation example of a single-camera trajectory through keypoints

4.2.1.1 Single-Camera Annotation

We considered several options for annotating single-camera trajectories. The first alternative was to run a multi-target tracker and generate a set of preliminary trajectories. The annotators would then inspect and correct the mistakes of the tracker. The second alternative was to use a single-target tracker for each person to generate trajectories independently. For both options the time required to correct and update the tracker predictions was similar to annotating the trajectories from scratch. Therefore we did not use these alternatives.

We evaluated a third option, crowdsourcing the annotation on Amazon Mechanical Turk through the VATIC tool of Vondrick et al. (2012). This approach is massively scalable but has the disadvantage that annotation style is not consistent among annotators, and still requires some degree of quality supervision. We also noted that VATIC required specifying an initial bounding box and adjusting it on every keyframe. The large number of clicks made this procedure very slow for annotating

trajectories with cumulative length of more than 30 hours.

We instead designed our own annotation interface that requires clicking on people's feet every few seconds and linearly interpolates the other trajectory points. An illustration is given in Figure 4.3. We also restricted the number of annotators to 5 and they hand-labeled all person trajectories including those that were often occluded. We observed that the annotation process required high concentration effort from the annotators and was mentally demanding. At the end of this phase we reviewed all trajectories by replaying the videos at various speeds and fixed minor issues. This phase was the most time consuming part of the annotation process.

4.2.1.2 Multi-Camera Annotation

Once we were satisfied with the quality of single-camera trajectories, we had to establish their correspondence across cameras. This step was assisted by a Re-ID system to propose candidate matches. For simplicity we used HSV features as descriptors in the Re-ID system. We extended the annotation tool to account for multi-camera association, and for ambiguous situations we referred to the stream of camera 8 that overlaps with several views. In some cases we had to use our own judgment based on context or gait. An illustration of the interface for across-camera correspondences is shown in Figure 4.4. At the end of this phase we had collected all single-camera trajectories and their correspondences across cameras. An illustration of a fully annotated frame is shown in Figure 4.6.

4.2.1.3 Bounding Boxes from Point Trajectories

The final stage required generating bounding boxes from the point trajectory annotations of each person. We specified one snapshot for each single-camera trajectory and extrapolated the width and height to all other frames using the homography information between the world and image plane. An example is shown in Figure 4.5.

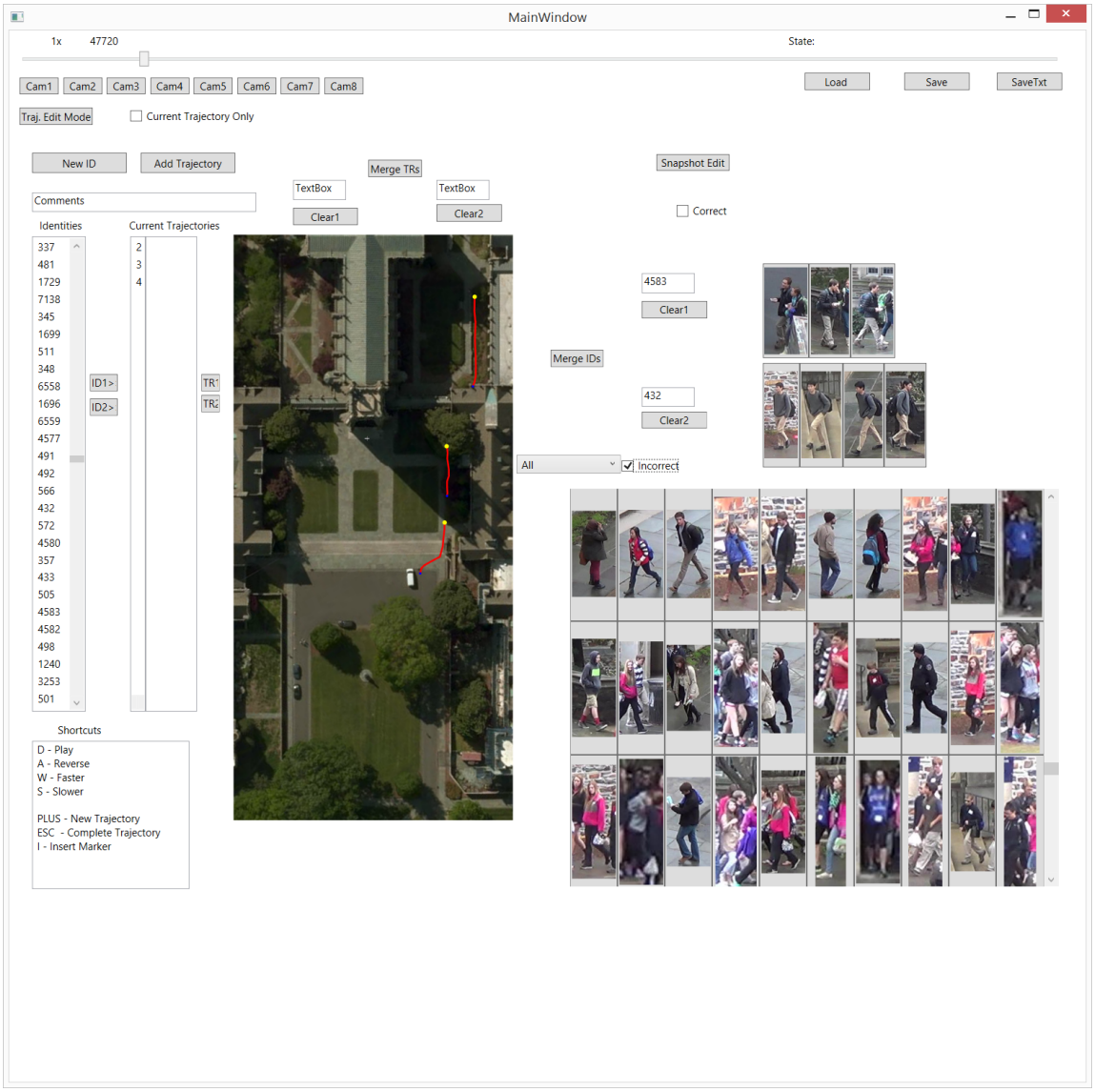


FIGURE 4.4: An illustration of the annotation tool for DukeMTMC

4.2.1.4 Potential Improvements

Current multi-target trackers have significantly improved since the creation of this dataset. A semi-supervised approach for dataset annotation is now reasonable. An example is the large-scale PathTrack dataset introduced by Manen et al. (2017). Also, carrying the annotation process on a screen with pen input rather than mouse input can significantly speed up the labeling process. Re-ID systems have also improved

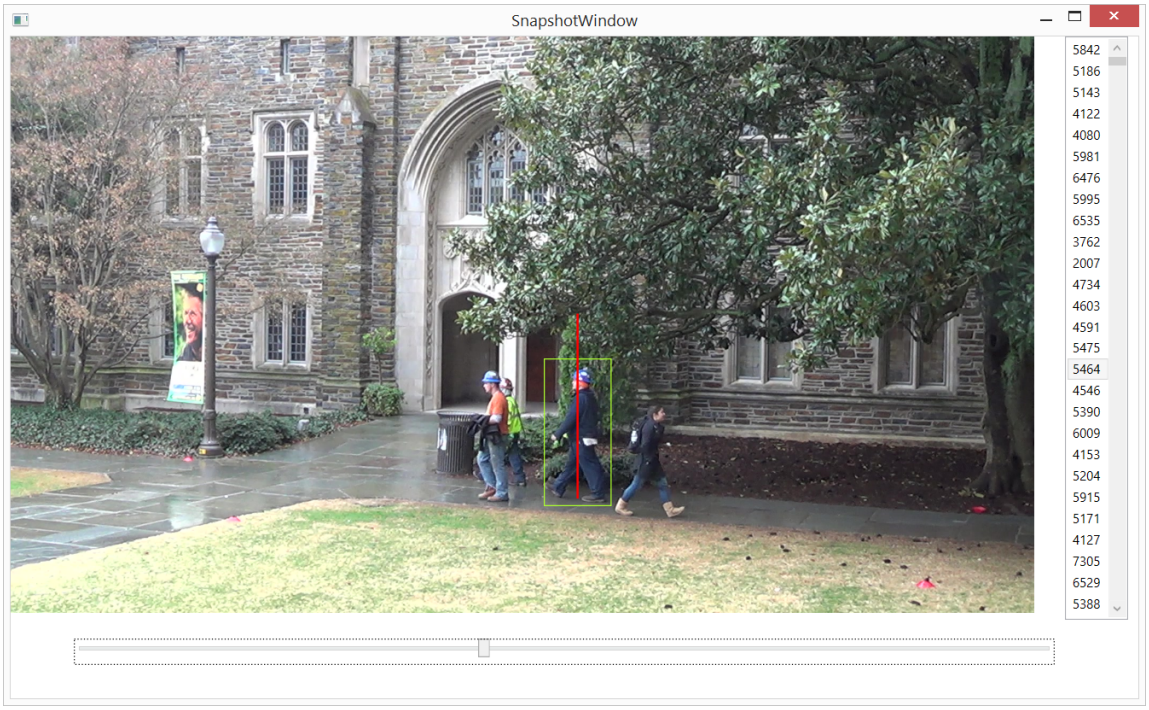


FIGURE 4.5: A bounding box is specified for a trajectory. The width and height are extrapolated for all other frames

both in accuracy and efficiency which makes the correspondence annotation more efficient. With all such recent improvements, it is now possible to generate other large-scale datasets in less time than for DukeMTMC.

4.2.2 Limitations

DukeMTMC covers a single outdoor scene from fixed cameras. Soft lighting from overcast weather may make tracking easier than otherwise. Views are mostly disjoint, which disadvantages methods that exploit data from overlapping views.

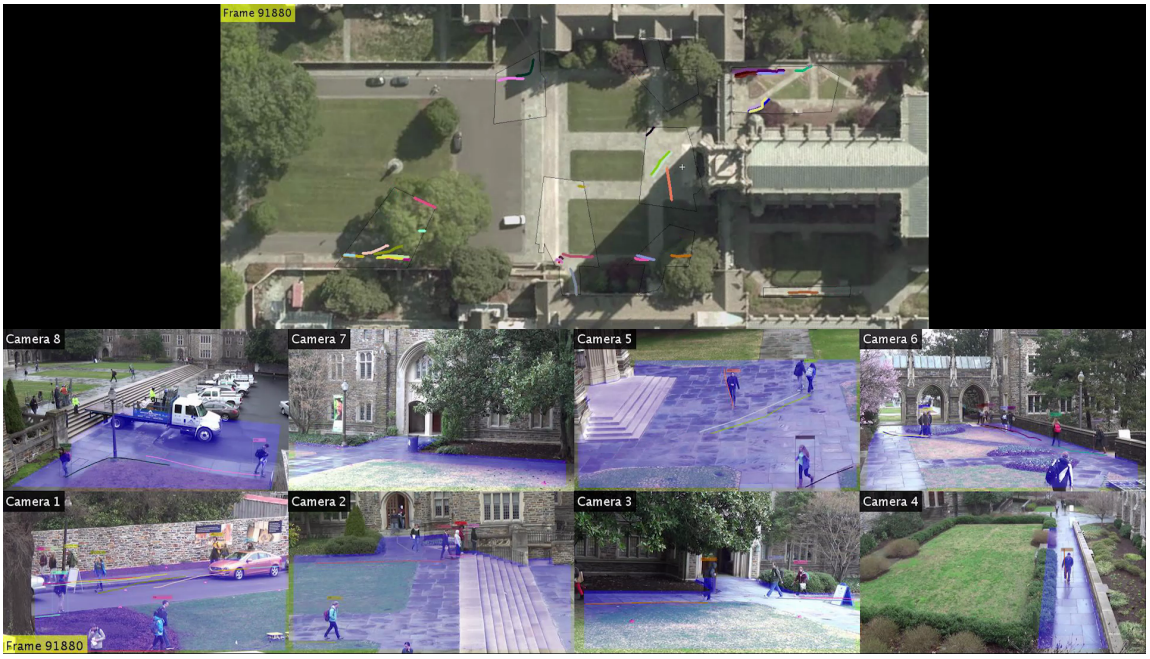


FIGURE 4.6: Trajectory annotations shown on both image and world plane. Annotations are restricted to the shaded blue regions that have no overlap across cameras.

4.3 DukeMTMC Benchmarks

4.3.1 MTMC Tracking

The DukeMTMC benchmark for single- and multi-camera tracking is hosted online on MOTChallenge [1]. It consists of a training/validation set and two test sets, easy and hard. The ground truth for training/validation set is publicly available whereas the ground truth for the test sets is withheld. Figure 4.7 illustrates the camera synchronization and dataset splits. A maximum of 4 submissions per tracker is allowed on the server, and ranking is done using the IDF_1 score introduced in the previous chapter.

The training/validation set is 50 minutes long to provide sufficient data for training deep learning models. The easy test set is 15 minutes long, and the hard test set is 10 minutes long. The hard test set consists of a group of 60 people traversing 4

[1] `https://motchallenge.net`

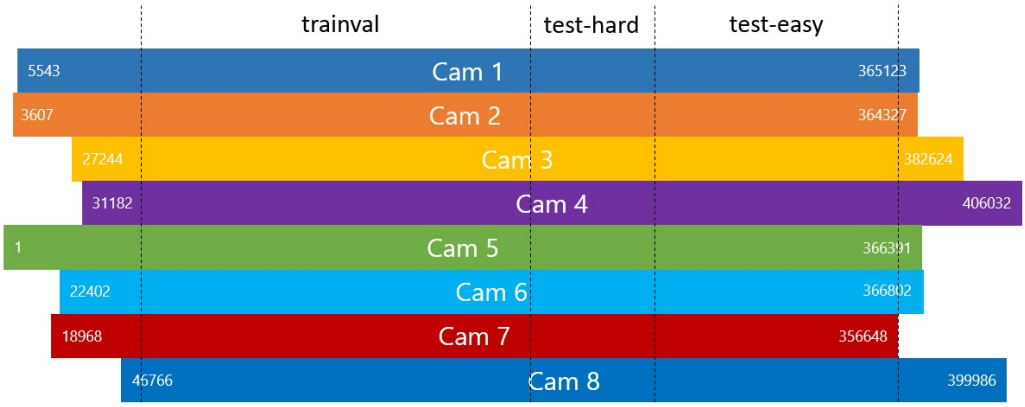


FIGURE 4.7: Camera frame synchronization and data set split for the DukeMTMC tracking benchmark

cameras, with frequent and long occlusions that make this sequence challenging. An illustration of the hard test set is shown if Figure 4.8.

FIGURE 4.8: A group of 60 people traversing 4 cameras on the DukeMTMC hard test sequence.

The DukeMTMC tracking benchmark has received more than 10 tracker submissions between its release and the time of this writing. The score distribution is shown in Figure 4.9.

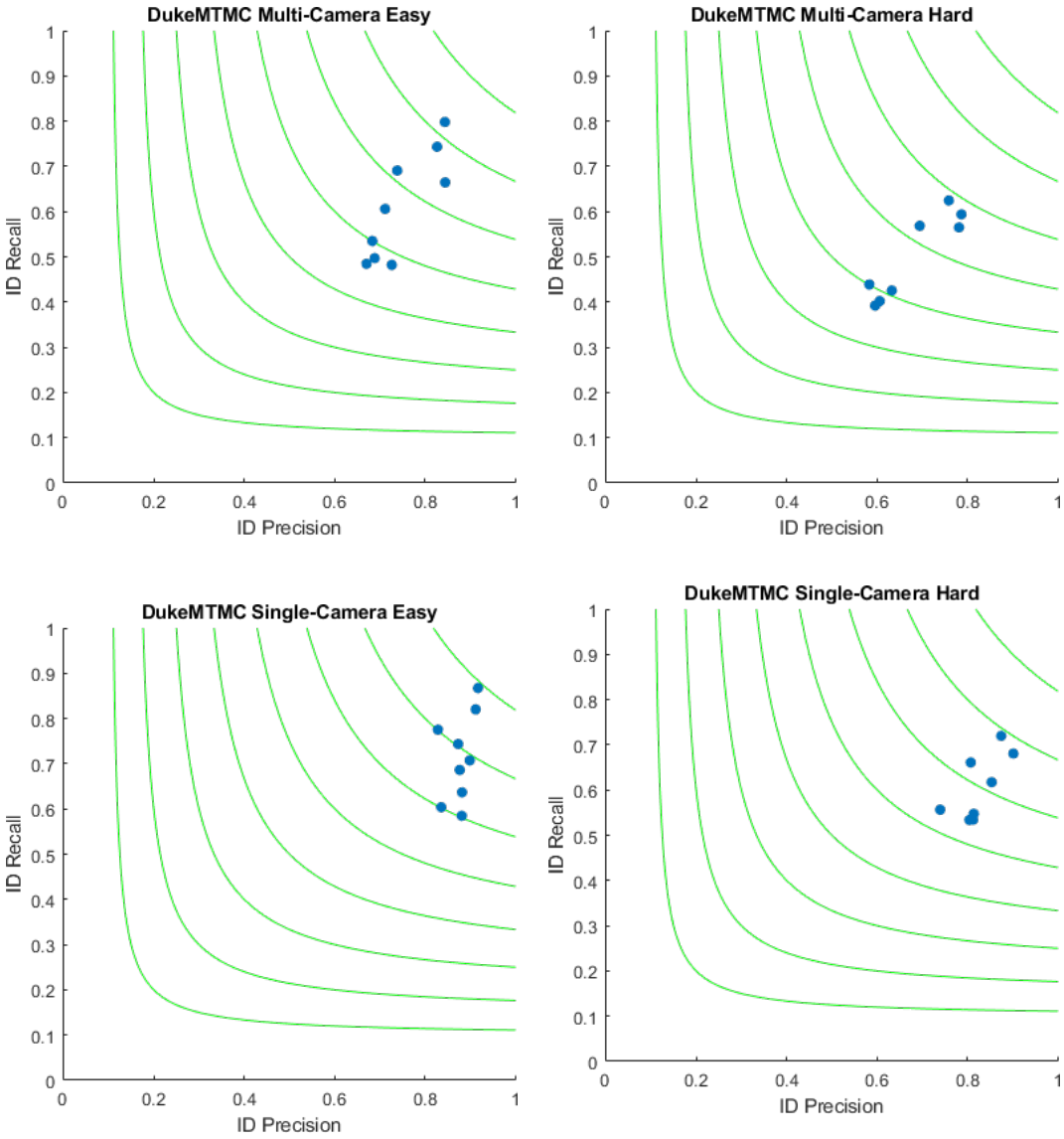


FIGURE 4.9: Identification Precision/Recall plots of tracker submissions for single- (top) and multi-camera (bottom) scenarios, for both the easy (left) and hard (right) test sets in the DukeMTMC tracking benchmark.

Both the baseline and the top performing method are our own and we discuss them in the next chapter.

4.3.2 Image-Based Person Re-Identification

The community has extended the DukeMTMC data set for image-based person re-identification. The most popular benchmark is DukeMTMC-reID by Zheng et al. (2017b). It consists of 702 identities for training and 702 for testing, while 408 identities are used as distractors. The pedestrian images are cropped from the training/validation set trajectories every 120 frames. The evaluation protocol picks one query image for each identity in each camera and puts the remaining images in the gallery. Re-ID systems are then evaluated using the Rank-1 accuracy and mean average precision (mAP) metrics. To date more than 30 systems have been submitted.

4.3.3 Video-Based Person Re-Identification

Wu et al. (2018) have extended DukeMTMC for video-based person re-identification. The DukeMTMC-VideoReID dataset consists of 702 identities for training, 702 identities for testing, and 408 identities as distractors. In total there are 2,196 videos for training and 2,636 videos for testing. Each video contains person images sampled every 12 frames. During testing, a video for each ID is used as the query and the remaining videos are placed in the gallery. The evaluation protocol and identity split is the same as in Zheng et al. (2017b).

5

Multi-Target Multi-Camera Tracking

Tracking multiple targets across multiple cameras remains a challenging problem. In addition to difficulties from varying appearance within and between cameras, a tracker has to deal with another pair of challenges. Person *occlusions*—whether caused by limited field of view, visual obstacles between camera and person, or algorithm failure—generate gaps in the input observations that make tracking harder. Conversely, overactive person detectors may generate *spurious observations* that confuse the tracker.

In Section 5.1 we discuss related work. The discussion concentrates around person detection, additional issues that arise when tracking across cameras rather than within, and how learning has been recently incorporated in tracking. The last subsection gives a broad broad overview of existing optimization algorithms for data association.

In Section 5.2 we discuss our method. In particular, we discuss how to compute an evidence graph from appearance and motion information, and how to partition the graph into sets that correspond to identities through correlation clustering. For efficiency, our method reasons hierarchically and uses a temporal sliding window for

online processing. Experiments in Sections 5.3-5.4 demonstrate the usefulness of our approach and analyze the various trade-offs in MTMC tracking.

5.1 Related Work

This Section summarizes prior research on person detection, methods to account for multiple views of the scene, the use of machine learning methods in tracking, and different optimization techniques used in MTMC tracking, which we compare to our correlation clustering method. The interested reader can refer to Luo et al. (2014) and Betke and Wu (2016) for additional surveys of related work.

5.1.1 Person Detection

The most popular framework in MTMC tracking is that of tracking-by-detection. Input observations are generated by a person detector that outputs a set of bounding box predictions and their confidence for every video frame.

The popular Deformable Parts Model (DPM) detector of Felzenszwalb et al. (2010) has been the most common detector before the emergence of deep learning. DPM was used as the public detector for MOTChallenge sequences MOT15 (Leal-Taixé et al., 2015) and MOT16 (Milan et al., 2016), and in labeling Re-ID datasets such as Market-1501 and MARS (Zheng et al., 2016a, 2015).

Since the MOT17 challenge, trackers have shown increased accuracy by utilizing detectors that rely on deep learning. These include Faster R-CNN by Ren et al. (2015), SSD by Liu et al. (2016), KDNT by Yu et al. (2016a), or pose-based detectors such as OpenPose by Cao et al. (2017) or DeeperCut by Insafutdinov et al. (2016). In our experiments we have experimented with the DPM and OpenPose detectors.

There is also work which doesn't use a detector, but assumes that single-camera trajectories are given as input (Bredereck et al., 2012; Cai and Medioni, 2014; Chen et al., 2011a, 2015, 2011b; Daliyot and Netanyahu, 2013; Das et al., 2014; Gilbert

and Bowden, 2006; Javed et al., 2008; Kuo et al., 2010; Makris et al., 2004; Zhang et al., 2015a). Such work evaluates association across cameras only, not taking into account detection errors that give rise to inaccurate single-camera trajectories.

5.1.2 Multiple Cameras

Tracking from multiple cameras comes with its own challenges compared to single-camera tracking. In many cases it is useful to know where people can possibly enter or leave the overall area under observation, to ease the correspondence problem. Some previous work has taken advantage of external knowledge of aspects like these. For instance, in many cases the topology of the cameras is assumed to be given, and the spatial relations between cameras are explicitly mapped in 3D (Chen et al., 2011a; Zhang et al., 2015a; Ristani et al., 2016; Ristani and Tomasi, 2018). In other more challenging cases the camera topology is not know and needs to be estimated from video. For this purpose some methods track known identities (Calderara et al., 2008; Javed et al., 2008; Jiuqing and Li, 2013) or compare entry/exit rates across pairs of cameras (Cai and Medioni, 2014; Kuo et al., 2010; Makris et al., 2004).

Knowing when people will possibly re-appear in a different camera is also useful for improving across-camera association. Many methods rely on modeling travel time either parametrically (Jiuqing and Li, 2013; Zhang et al., 2015a) or not (Chen et al., 2011a; Gilbert and Bowden, 2006; Javed et al., 2008; Kuo et al., 2010; Makris et al., 2004). In our method we use time constraints to rule out unlikely inter-camera associations, and use motion predictions across cameras as soft constraints. In particular, we decay correlations to zero as the time distance between observations increases. Correlation decay ensures that time-distant observations are associated if there is a chain of positively-correlated observations that connect them. The idea is similar to lifted multicuts by Tang et al. (2017), although we employ no threshold or hard constraints.

The problem of matching observations from multiple cameras could benefit from the redundancy deriving from any overlap between the fields of view of the cameras, both to ease occlusion problems and to improve the localization accuracy of the people being tracked. In scenarios where camera views overlap, pre-processing method may fuse data from partially overlapping views (Zhang et al., 2015b). Some systems rely on completely overlapping and unobstructed views and the main challenge is accurate data fusion (Ayazoglu et al., 2011; Berclaz et al., 2011; Bredereck et al., 2012; Hamid et al., 2010; Kamal et al., 2013).

5.1.3 Learning to Track

There have been several attempts to learn multi-target tracking data association in a supervised way. Milan et al. (2017) employ a recurrent neural network for end-to-end prediction of trajectories and Schulter et al. (2017) learn data association by back-propagating through a network-flow solution. These two methods have been pushing in the direction of end-to-end learning which has shown very good results in other computer vision problems such as image classification (He et al., 2016). Even though end-to-end learning for multi-target tracking has not yet topped single-camera tracking benchmarks, it remains a promising direction to pursue.

In our method we follow a different approach. We learn features for correlations independently, without resorting to an end-to-end trainable tracking pipeline that assesses trajectory quality as its loss and back-propagates the loss through a combinatorial optimization layer. Our argument is that if correlations are good, even greedy association suffices. This idea has been shown to work for person detection in Cao et al. (2017), and implicitly pursued in single-camera trackers (Tang et al., 2016, 2017; Yu et al., 2016a) and Re-ID methods (Hermans et al., 2017; Zheng et al., 2016b, 2017a) that improve features to increase accuracy. Learning good correlations makes training simpler and less expensive, and we show that it achieves state-of-the-art

performance for MTMCT.

5.1.4 Optimization

Almost all MTMC trackers employ *optimization* to maximize the coherence of observations for predicted identities. They first summarize spatial, temporal, and appearance information into a graph of *weights* that express the affinity of node observations, and then partition the nodes into identities either greedily through bipartite matching or, more generally, by finding either paths, cliques, or sets with maximal internal weights. We discuss several optimization formulations for MTMC tracking that have appeared in the literature.

5.1.4.1 Minimum Cost Bipartite Matching

Perhaps the most well know data association algorithm is Minimum Cost Bipartite Matching (MCBM). It is mostly used in online tracking settings. In the bipartite graph the nodes represent observations and edges have weights that denote the affinity between observations. One partition contains observations from the past and the other partition contains observations from the current frame. The first partition often includes observations from several previous frames to help account for identities that were occluded.

Minimum cost bipartite matching of a graph with n nodes and m edges can be solved efficiently in $O(n^2m)$ time with the Hungarian algorithm (Kuhn, 1956). Compared to our formulation, MCBM and all formulations equivalent to it have the disadvantage that they only consider consecutive edge weights in the formulation, rather than all pairwise evidence. This restriction results in more fragmentations and lower quality trajectories.

Some methods include node as well as entry/exit costs in the optimization to form an extended bipartite graph. An example is shown in Figure 5.1.

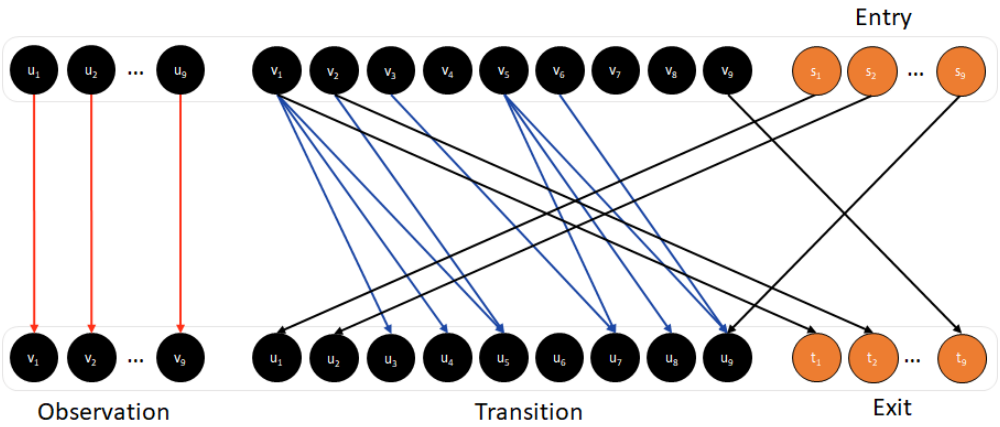


FIGURE 5.1: An extended bipartite graph to include detection costs as well as entry/exist costs. This instance is constructed from the graph shown in Figure 5.5. A red edge represents the cost of observation u_i being valid. Blue edges represent the costs of matching consecutive observations v_i and u_j. Black edges connected to black nodes represent the cost of the black node appearing or disappearing at that frame.

Some single-camera methods that employ MCBM include Shu et al. (2012); Wu and Nevatia (2007). Some multi-camera trackers also use the same optimization, including Kuo et al. (2010); Cai and Medioni (2014); Chen et al. (2015); Daliyot and Netanyahu (2013).

5.1.4.2 Maximum Weight Independent Set

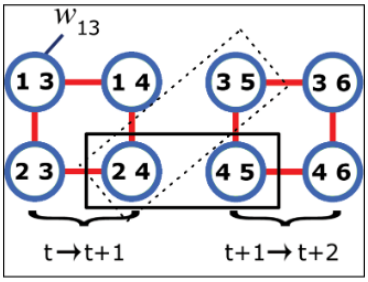


FIGURE 5.2: Nodes (tracklets) are shown in blue and edges (forbidden associations) in red. The solid rectangle is a plausible track whereas the dashed rectangle is not a single track. Image from Brendel et al. (2011).

Brendel et al. (2011) formulate data association as a Maximum Weight Inde-

pendent Set (MWIS) problem. Given a set of detections, a graph is constructed such that nodes represent pairs of consecutive detections ("tracklets"), and edges forbid the association of tracklets (a, b) and (c, d) when either a and c or b and d are co-identical. An illustration is given in Figure 5.2. The optimal association of detections into tracklets is then found by computing the MWIS of the graph. The authors provide an $O(m^2)$ algorithm algorithm for finding a local maximum to the general MWIS problem, where m is the number of tracklets.

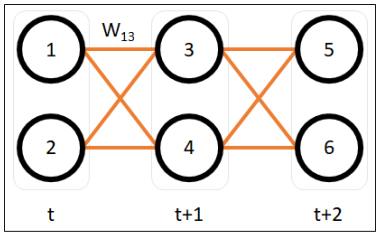


FIGURE 5.3: The instance from Figure 5.2 is converted to a bipartite matching problem.

We note that the 2-frame version of the MWIS problem is identical to MCBM. The reduction converts every tracklet of the MWIS graph into two consecutive detection nodes which are connected by an edge. This edge has weight equal to the node (tracklet) weight from the MWIS graph. An illustration is shown in Figure 5.3. Therefore 2-frame MWIS can be solved optimally and efficiently in $O(n^2m)$ time for n detections and m tracklets.

5.1.4.3 Maximum Weight Disjoint Paths Cover

Javed et al. (2008) formulate data association as a Maximum Weight Disjoint Paths Cover (MWDPC) problem. Given a set of detections, a graph is constructed such that edges between nodes are directed in time. The weight of an edge is a measure of affinity between two detections. The authors show that finding the Maximum Weight Disjoint Paths Cover is no different than minimum cost bipartite matching when the graph is directed and acyclic. Each node can be duplicated into entry (+)

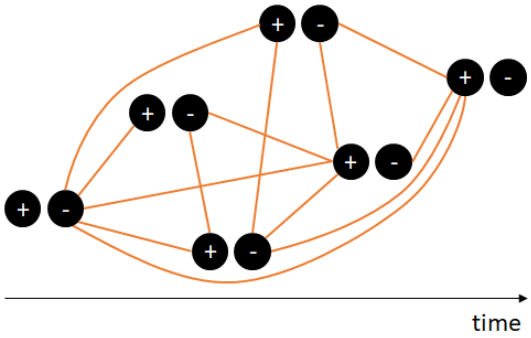


FIGURE 5.4: An example of the split graph for finding the maximum weight path disjoint cover. Each node from the original graph is split into entry (+) and exit (-) nodes, and edges are directed in time.

and exit (-), so the optimization is done between all (+;-) time-ordered edges. An example is shown in Figure 5.4.

5.1.4.4 Minimum Cost Network Flow

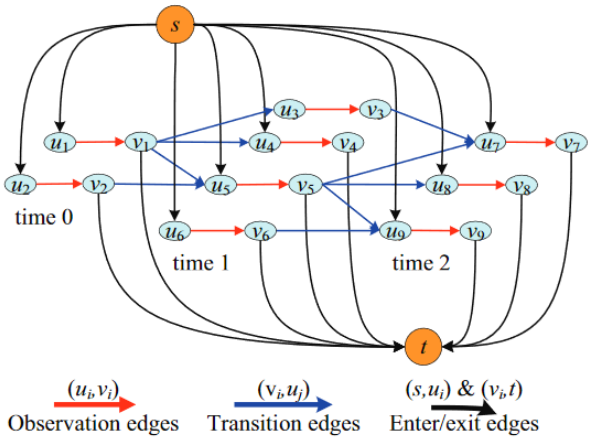


FIGURE 5.5: An example of the cost flow network with 3 timesteps and 9 observations. Image courtesy of Zhang et al. (2008).

Zhang et al. (2008) and Pirsiavash et al. (2011) formulate data association as a Minimum Cost Network Flow (MCNF) problem. Given a set of detections, a graph is constructed such that edges between nodes are directed in time. The weight of an edge is a measure of affinity between two detections. The cost of a detection d is

encoded by expanding the detection into an edge (u_d, v_d). Two additional nodes are added, source and sink. The source (sink) is connected to all edge start (end) nodes. An illustration is given in Figure 5.5. The authors give an algorithm with running time $O(n^2 m \log n)$ for a graph with n nodes and m edges.

Pirsiavash et al. (2011) note that if n and m scale linearly with the number of frames N, then the algorithm of Zhang et al. (2008) runs in $O(N^3 \log N)$ to find K tracks. Noting that the graph has unit capacity edges and is directed and acyclic, Pirsiavash et al. (2011) give an improved $O(KN \log N)$ algorithm involving Successive Shortest Paths and bisection search over K. They also give a dynamic programming greedy approximation algorithm with $O(KN)$ complexity. This algorithm proves as effective as the one involving Successive Shortest Paths.

The Minimum Cost Network Flow graph of Zhang et al. (2008); Pirsiavash et al. (2011) is richer than the standard MCBM formulations as it includes costs for detections, and costs for starting/ending a trajectory at each detection.

Minimum Cost Network Flow is more general than Maximum Weight Path Disjoint Cover or Minimum Cost Bipartite Matching, *e.g.*, when edges have capacity different from 1. It is important to emphasize that in tracking, and when edges still have capacity 1, MCNF is still more general. This is because a feasible solution to MCBM does not necessarily translate to a feasible flow in MCNF. MCNF can choose not to route flow through particular nodes if that decision produces a lower cost.

5.1.4.5 Graph Multicuts

Tang et al. (2016) formulate data association as a graph multicut problem (MP). Graph multicut was introduced after our correlation clustering formulation in Ristani and Tomasi (2014). Nonetheless, we note that the two optimization problems are equivalent as was shown in Demaine et al. (2006). The difference is only semantic: Correlation clustering maximizes positive correlations within clusters, graph multicut

minimizes negative correlations between clusters. Graph multicuts is an NP-hard problem and the authors use the Kernighan and Lin (1970) approximation algorithm.

5.1.4.6 Consistent Re-Identification Binary Integer Program

Das et al. (2014) formulate data association as an NP-hard binary integer program. Given a set of person detections from different cameras, the binary integer program forms multi-camera sets that maximize the correlation between sets from pairwise terms. A special constraint doesn't allow person detections from the same camera to be placed in the same set. We note that this binary integer program is a special case of correlation clustering. We allow associations of people returning to the same camera, while Das et al. (2014) do not.

5.1.4.7 Generalized Minimum Clique

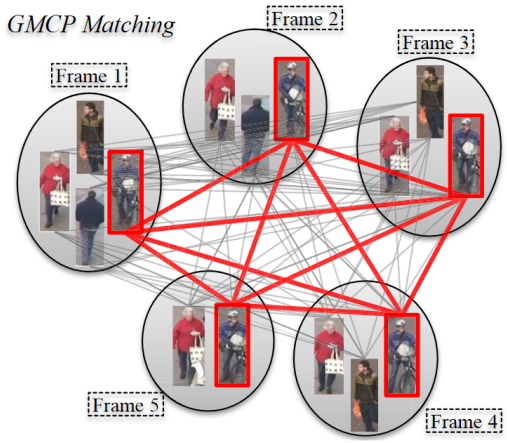


FIGURE 5.6: An illustration of the Minimum Weight Clique (red). Image from Zamir et al. (2012).

Zamir et al. (2012) formulate data association as a Generalized Minimum Clique Problem (GMCP). A graph is constructed where nodes represent detections and edges represent their affinity. GMCP then associates detections into identities iteratively, one clique (identity) at a time. Every time the minimum clique is computed,

its edges and nodes are removed from the graph, and the process is repeated until the graph is empty. An illustration of one iteration is shown in Figure 5.6 where one detection from each frame is selected to produce the minimum weight clique.

GMCP is NP-hard and Zamir et al. (2012) rely on RANSAC optimization. We note that iterating GMCP solves the correlation clustering problem approximately. In our correlation clustering formulation, we solve for all detections jointly and optimally.

5.1.4.8 Constrained Dominant Set

Tesfaye et al. (2017) formulate data association as a Constrained Dominant Set Clustering (CDSC) Problem. The problem is NP-hard and the authors rely on quadratic optimization to solve it. The graph construction and iterative association procedure is similar to GMCP. The difference is that during an iteration, CDSC requires that a predetermined node must belong to the dominant set. In GMCP on the other hand the minimum weight clique is not tied to any node.

5.1.4.9 Generalized Maximum Multi-Clique

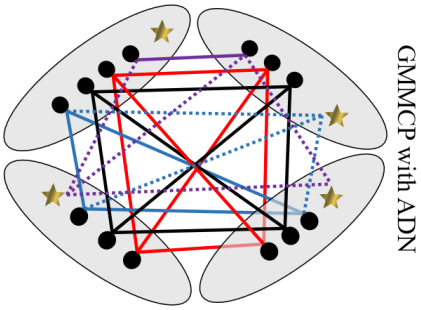


FIGURE 5.7: An illustration of the Maximum Weight Multi-Clique problem. In this instance 4 cliques are found, each shown in a different color. The method uses aggregated dummy nodes (stars) to efficiently account for occlusions. Image from Dehghan et al. (2015).

Dehghan et al. (2015) formulate data association as Generalized Maximum Multi-

Clique Problem (GMMCP). This work extends GMCP to solve the assignment problem jointly and optimally for all identities. An illustration is shown in Figure 5.7.

GMMCP is NP-Hard and similar to our method Dehghan et al. (2015) solve GMMCP optimally through a Binary Integer Program. Unlike our correlation clustering formulation, GMCCP also considers unary terms. However, GMCCP forbids the association of simultaneous detections. Correlation clustering doesn't add such a constraint, which makes it useful for the non-maximum suppression of overlapping bounding boxes, or associating detections from overlapping views.

5.1.4.10 Pairwise Costs Network Flow

Chari et al. (2015) formulate data association as a Pairwise Costs Network Flow problem (PCNF). They use the same graph as in the Minimum Cost Network Flow problem but extend the optimization objective to account for all pairwise costs. This means that unlike MCNF, now the cost of a source-tank path (trajectory) accounts for all pairwise costs in the trajectory, rather than only costs of edges along the trajectory. Unlike MCNF, PCNF is NP-hard, and the authors rely on Integer Linear Programming to obtain a solution.

5.1.4.11 Subgraph Decomposition

Tang et al. (2015) formulate data association as a subgraph decomposition problem (SDP). The graph consists of nodes that represent detections and edges that represent affinities, and the graph is decomposed into identities. An illustration is shown in Figure 5.8. Unlike GMMCP and PCNF this formulation allows the association of simultaneous detections if the costs allow. SDP is more general than our correlation clustering formulation because it accounts for detection costs. However, we have found in our experiments that unary terms play negligible role in the quality of the trajectories.

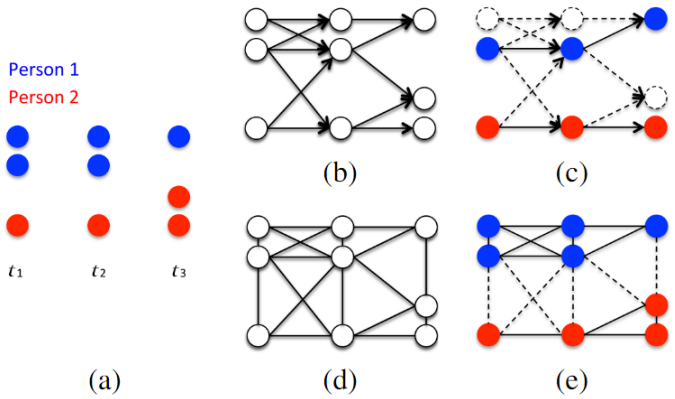


FIGURE 5.8: (a) Ground truth detections for two identities. (b) A directed graph and (c) its maximum weight path disjoint cover which accounts only for time consecutive edges and forbids association of simultaneous observations. (d) An undirected graph and (e) the maximum weight subgraph decomposition. Image from Tang et al. (2015).

5.1.4.12 Lifted Multicuts

Tang et al. (2017) introduce the Lifted Multicut Problem (LMP) which generalizes multicut by categorizing edges into regular and lifted. Edges are regular when they connect detections no more than T frames apart. Edges are lifted when they connect detections separated by more than T frames. The formulation enforces motion constraints directly in the graph. It requires that (a) two nodes connected by lifted edges are connected in the solution iff the two nodes are also connected by a path or regular edges. Two examples are shown in Figure 5.9.

When T equals the video length L, all edges are regular and the problem is equivalent to graph multicuts and correlation clustering. When $T < L$ the feasible solutions of lifted multicuts are a subset of feasible solutions in regular multicuts with $T = L$. That is because any feasible multicut set that violates conditions (a) is infeasible in lifted multicuts.

Tang et al. (2017) have empirically shown that LMP performs better than correlation clustering for single-camera tracking. It is not yet known if the LMP formulation

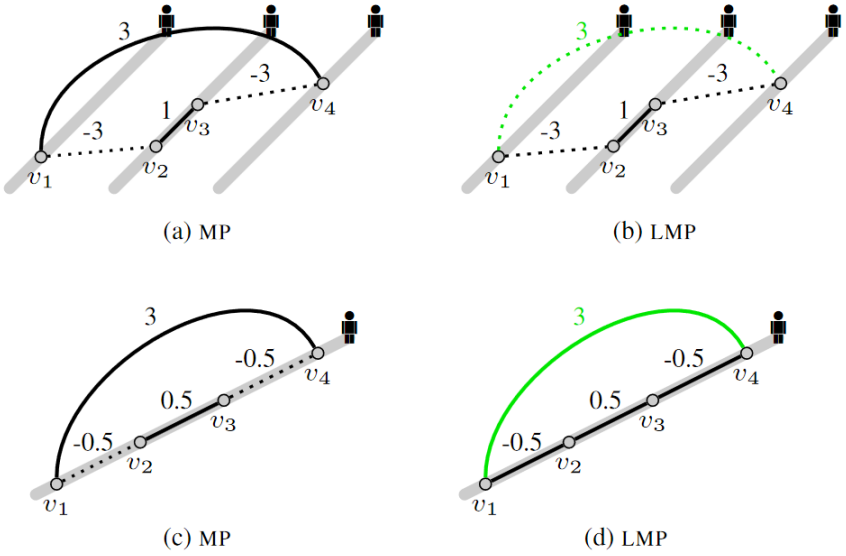


FIGURE 5.9: Ground truth trajectories are shown in grey, regular edges in black, and lifted edges in green. In the optimal solution solid lines indicate co-identity, and dashed lines indicate cuts. Correlations are shown on each edge. (a) An example where MP incorrectly merges v_1 and v_3 whereas (b) LMP does not have evidence of regular path connectivity for this long range association. (c) MP incorrectly fragments the true trajectory whereas (d) LMP makes a correct assignment due to the lifted edge. Image from Tang et al. (2017).

works well in MTMC tracking. It could happen that a person appears in camera A, walks through cameras B and C unobserved, and after T frames appears in camera D. The direct edge (lifted) would capture the similarity of the two appearances yet condition (a) cannot be satisfied because the person was not observed in cameras B and C (lack of motion evidence). Correlation clustering would be able to make this association despite some of its drawbacks in single-camera tracking that were shown in Figure 5.9.

5.1.4.13 Other Methods

Henriques et al. (2011) propose a simple yet effective approach to track targets in groups that merge or split. They rely on minimum cost bipartite matching and minimum cost flow circulation to infer the trajectories of individuals. Andriyenko

et al. (2012) formulate multi-target tracking as a discrete-continuous optimization problem. The discrete optimization solves the combinatorial problem of data association, and the continuous optimization refines the estimated trajectories. Butt and Collins (2013) formulate data association as a weight maximization problem over frame triplets. The reason for selecting triplets instead of pairs is motivated by the constant velocity motion model. Kumar et al. (2014) also formulate tracking as a graph partitioning problem. They employ unary and pairwise terms, and also include triplets for curvature information. They use the Tree Re-Weighted Message Passing algorithm for approximate but efficient inference. Wen et al. (2014) use an undirected hypergraph for data association, claiming that hyper-edges can better capture long-term temporal dependencies. Henschel et al. (2017) improve the Frank-Wolfe optimization algorithm and show improved performance in the task of multi-detector multi-target tracking.

In a different type of approach, Milan et al. (2015) jointly track and segment moving targets. This approach proves more adequate in crowded scenes where standard detectors suppress low confidence detectors.

More recently Xiang et al. (2015) use a Markov Decision Process formulation trained through reinforcement learning to make online decisions on whether to associate existing tracks, terminate them, or initialize new tracks. Schulter et al. (2017) use bi-level optimization within a network flow framework to learn the affinities between detections using a deep network. Maksai et al. (2017) use global behavioral patterns to improve the results of any tracker through optimization.

Summary. Many data association methods have been proposed in recent years and rely on combinatorial optimization. The trend has been to account for all pairwise evidence given the weak performance of algorithms that consider only limited evidence, and to use optimal solvers instead of approximation algorithms. We were the first to formulate single- and multi-camera tracking as a correlation clustering

problem (Ristani and Tomasi, 2014; Ristani et al., 2016), and to avoid approximate algorithms with no guarantees by solving Binary Integer Programs optimally.

5.2 Method

In our approach we take as input a set of synchronized image streams from different cameras, and estimate the position of each person at any point in time. Formally, the input is a set of videos $V = \{V_1, \ldots, V_n\}$ from n different cameras, and the ground truth is a set of multi-camera trajectories $T = \{T_1, \ldots, T_\ell\}$. A trajectory $T_k = \{(\mathbf{x_i}, t_i)\}$ is an ordered set of tuples consisting of target state $\mathbf{x}$ (usually a bounding box) and time t. Times are assumed to be unique. MTMCT could be cast as a supervised learning problem: Find the optimal parameters Θ^* of a function $f(\Theta, V)$ that estimates the true trajectories as well as possible:

$$\Theta^* = \arg\min_{\Theta} \mathcal{L}(f(\Theta, V), T) \tag{5.1}$$

where the loss function $\mathcal{L}$ could be derived from the multi-camera tracking accuracy measure IDF_1.

However, end-to-end training as is customary in the literature relies on back-propagating the loss through a combinatorial optimization layer that performs data association, and this is expensive as shown in Schulter et al. (2017). We avoid this complexity by noting that if the correlations were positive for co-identical pairs and negative for non co-identical pairs, then combinatorial optimization would be trivial. Thus, we aim to learn features that produce good correlations during training, while at test time we employ correlation clustering to maximize agreement between potentially erroneous correlations.

An additional source of difficulty during training is model depth, as weight updates can fail to propagate back to early layers responsible for person detection. If the network is monolithic and trained with a single loss, training becomes more difficult.

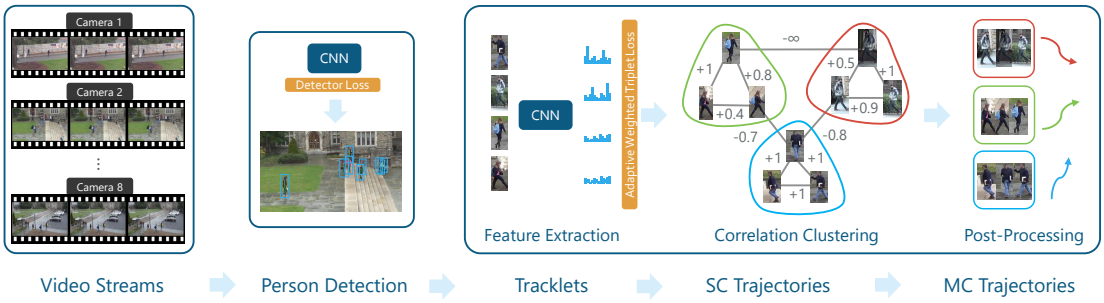


FIGURE 5.10: An illustration of our pipeline for Multi-Target Multi-Camera Tracking. Given video streams, a person detector extracts bounding box observations from video. For trajectory inference, a feature extractor extracts motion and appearance features from observations. These are in turn converted into correlations and labeled using correlation clustering optimization. Finally, post-processing interpolates missing detections and discards low confidence tracks. Multi-stage reasoning repeats trajectory inference for 1-second long tracklets, single- and multi-camera trajectories. At training time the detector is trained independently, and the feature loss penalizes features that yield wrong correlations.

We therefore separate detection and association as is customary in the literature. An illustration of our processing pipeline is shown in Figure 5.10. In the following we describe how we learn appearance features, and the different parts of the tracker.

Given a $k \times o_d$ array O_d of k-dimensional detections as input, the tracker outputs a $(k+1) \times o_t$ array $O_t = f_t(\Theta_t, O_d)$ of o_t detections. The added dimension is the identity label assignment to the input observations. An observation $o = \{\boldsymbol{\varphi}, \mathbf{p}, t, \mathbf{v}\}$ constists of appearance feature $\boldsymbol{\varphi}$, position $\mathbf{p}$, time stamp t, and estimated velocity[1] $\mathbf{v}$.

In our design, the tracker first computes features for all o_d input observations, then estimates correlation between all pairs of features, and finally solves a correlation clustering problem to assign identities to observations. Two post-processing steps, interpolation and pruning, interpolate detections to fill gaps and remove trajectories with low confidence. For this reason, the number o_t of output detections can differ from the number o_d of input detections.

[1] Velocity is a vector, and its norm is called the *speed*.

5.2.1 Person Detection

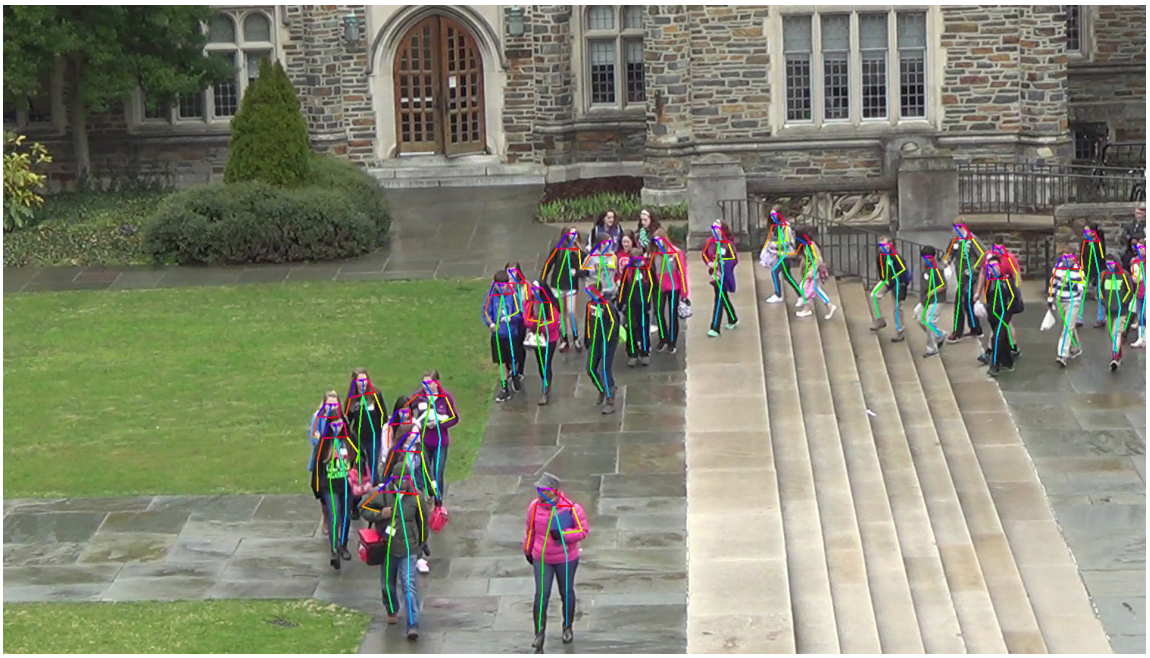


FIGURE 5.11: Sample detections from OpenPose on a crowded scene of the DukeMTMC data set

We use the off-the-shelf OpenPose person detector by Cao et al. (2017) which achieves good performance. This detector learns part affinity fields to capture the relation between body parts and applies greedy parsing to combine part affinities into bounding boxes. During training it is supervised directly on part affinities rather than bounding box accuracy. Detection results on a crowded scene are shown in Figure 5.11.

Since the output of this detector is a skeleton, we regress a bounding box by registration. Specifically, we estimate the bounding box center, width, and height which minimize the least squares error between the skeleton's visible joints and the corresponding joints on a template bounding box.

5.2.2 Appearance Correlation

We use the ResNet50 model pre-trained on ImageNet to extract appearance features. We follow its *pool5* layer by a dense layer with 1024 units, batch normalization, and ReLU. Another dense layer yields 128-dimensional appearance features. We train the model with the adaptive weighted triplet loss, data augmentation, and hard-identity mining. Details on appearance feature learning are given in Section 2.2.

We define the appearance correlation between two detections as $w_{ij} = \frac{t_a - d(\boldsymbol{\varphi}_i, \boldsymbol{\varphi}_j)}{t_a}$ (a number $\leqslant 1$) where the threshold $t_a = \frac{1}{2}(\mu_p + \mu_n)$ separates the means of positive and negative distances μ_p and μ_n of all training pairs, $d(\cdot, \cdot)$ denotes Euclidean distance, and $\boldsymbol{\varphi}_i$ denotes the appearance descriptor of observation o_i.

5.2.3 Motion Correlation

We use a linear motion model to predict motion correlation. For two observations $o_1 = \{\boldsymbol{\varphi}_1, \mathbf{p}_1, t_1, \mathbf{v}_1\}$ and $o_2 = \{\boldsymbol{\varphi}_2, \mathbf{p}_2, t_2, \mathbf{v}_2\}$, we define the motion error to be $e_m = e(o_1, o_2) + e(o_2, o_1)$, where $e(o_1, o_2) = \|\mathbf{q}_1 - \mathbf{p}_2\|_2$ measures the error between the position $\mathbf{p}_2$ of observation o_2 and the estimated position $\mathbf{q}_1 = \mathbf{p}_1 + \mathbf{v}_1(t_2 - t_1)$ of observation o_1 at time t_2.

We use the trajectories from the training set to learn a threshold t_m that separates positive and negative evidence, and a scaling factor α to convert errors to correlations: $w_m = \alpha(t_m - e_m)$. Associations that are physically impossible, *e.g.*, distant observations at the same time, receive correlation $w_m = -\infty$.

5.2.4 Correlation Clustering

Our tracker reasons about evidence for or against any two observations being *co-identified*, that is, being assigned to the same identity. If the appearance descriptors of two observations are similar and their times and locations are consistent with typical walking speeds, evidence for their co-identity is positive. If two observations look

different or occur at nearby time instants at faraway locations, evidence is negative. In the limit, "hard" evidence may be available: Simultaneous observations at faraway locations cannot possibly correspond to the same person, and yield "infinitely negative" evidence. "Infinitely positive" evidence, on the other hand, denotes an irreversible commitment to a co-identification. Infinite evidence, positive or negative, is equivalent to hard constraints on the solution.

We associate evidence to the edges of an *evidence graph* that has one node per observation and one edge for each pair of observations for which co-identity evidence is available. Multi-person tracking then becomes a *correlation clustering* problem. Specifically, the set of nodes is partitioned into subsets—one subset per person identity—such that edges within sets accumulate high positive evidence, and edges between sets accumulate high negative evidence. As known from Bansal et al. (2002), the resulting problem is Binary Integer Program (BIP) when the input is finite.

5.2.4.1 Formulation

Consider a set V of n observations that could be individual outputs from a person detector, or the results of aggregating co-identical detections with some other method. For a pair of observations u, v in V, let w_{uv} be a measure of the evidence for or against the hypothesis that u and v are co-identical. For now, it suffices to say that evidence is quantified by a *correlation*, that is, a number in the set $\{-\infty, [-1, 1], +\infty\}$. Positive values indicate evidence for co-identity, negative values indicate evidence against, zero denotes indifference, and infinite values correspond to definitive evidence (hard constraints).

Let the *evidence graph* $G = (V, E, W)$ be a weighted graph[2] on V. If a correlation

[2] The graph can be directed from past to future in time, if simultaneous observations cannot be co-identical, or undirected otherwise.

is available for a pair of observations in V, an edge is added to set E for that pair, and its correlation is added to set W. In the following, we think of G as being a complete graph, but nothing in our formulation depends on this.

A multi-person tracker partitions V into sets believed to refer to distinct identities. Specifically, the partition maximizes the sum of the rewards w_{uv} assigned to edges that connect co-identical observations and the penalties $-w_{uv}$ assigned to edges that straddle identities. This graph partition problem can be rephrased as the following BIP:

$$\arg\max_{X} \sum_{(u,v)\in E} w_{uv} x_{uv} \tag{5.2}$$

subject to

$$x_{uv} \in \{0,1\} \qquad \forall (u,v) \in E \tag{5.3}$$

$$x_{uv} + x_{vt} \leqslant 1 + x_{ut} \qquad \forall (u,v),(v,t),(u,t) \in E\ . \tag{5.4}$$

The set X is the set of all possible combinations of assignments to the binary variables x_{uv}, with the interpretation that x_{uv} is 1 iff the observations u and v are co-identical. The constraints in Equation (5.4) enforce co-identity to be transitive: If u and v are co-identical and so are v and t, then u and t must be co-identical as well.

Considering all pairwise correlations has the advantage that even when some edges carry negative correlation, the nodes they connect can still be co-identical if the overall reward of the set is positive, and *vice versa*.

Finding an optimal solution to this BIP is NP-hard Bansal et al. (2002) and the problem is also hard to approximate Tan (2008). The best known approximation algorithm as shown in Swamy (2004) achieves an approximation ratio of 0.7664, but its semi-definite program formulation makes it slow for practical consideration.

Other algorithms exist and we describe one of them next, but they come with no quality guarantees. Nonetheless they are useful for finding a good initialization to reach a globally high-quality solution.

5.2.4.2 Approximate Initialization

Three algorithms for graph partitioning have been recently discussed in Bagon and Galun (2011), namely: Expand-and-Explore, Swap-and-Explore, and Adaptive Label Iterative Conditional Modes (AL-ICM). Other approximate correlation clustering algorithms include Kernighan and Lin (1970) and Fiduccia and Mattheyses (1988). We use AL-ICM to initialize the BIP solution because of its speed and ability to scale to large problems, then we solve the BIP optimally using the *Branch and Cut* algorithm of Padberg and Rinaldi (1991).

Given a labeling vector $L = \{1, 2, \ldots\}^n$ the AL-ICM algorithm assigns a label l_u to observation u so as to minimize the following energy function:

$$E(L) = \sum_{uv} w_{uv} \vec{1}_{[l_u \neq l_v]} \tag{5.5}$$

where $\vec{1}_{[P]}$ is 1 when P is true and 0 otherwise. Minimizing this energy function is equivalent up to a constant to maximizing rewards and minimizing penalties in Equation (5.2). This energy is lowered when observations supported by negative correlation are labeled differently and when observations supported by positive correlation are labeled identically. This discrete energy minimization formulation has the advantage that the labeling vector L consists of n variables whereas the co-identity matrix X in our formulation consists of n^2 variables. This allows AL-ICM to scale to $n \geqslant 100,000$ observations.

AL-ICM is a greedy search algorithm. In each iteration, every variable is assigned the label that minimizes the energy, conditioned on the current label of the other variables. While ICM by Besag (1986) requires a fixed number of labels, AL-ICM

handles a varying number of labels as follows: conditioned on the current labeling, each observation is assigned to the most rewarding partition, or to a new partition if penalized by all current partitions. The algorithm terminates either when the energy cannot be minimized further or when a predefined number of iteration is reached.

5.2.5 Hierarchical Reasoning

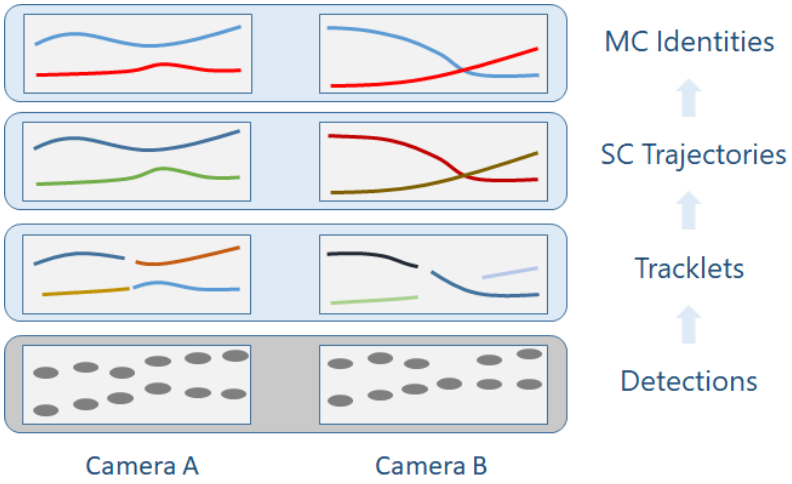
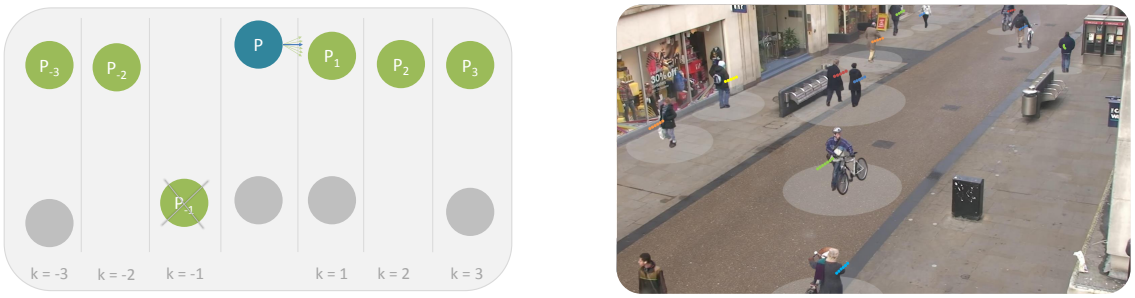


FIGURE 5.12: Our method computes multi-camera identities over three levels of hierarchy to reduce computational complexity. First tracklets are estimated from detections. Then single-camera trajectories are estimated from tracklets. Finally, multi-camera identities are estimated from single-camera trajectories.

We reason over three levels of hierarchy to break down computational complexity as is customary in the literature. Our hierarchy is shown in Figure 5.12.

5.2.5.1 Tracklets

In the first stage we associate detections into tracklets for every 1-second interval. We compute appearance correlations and motion correlations as described previously. Because detections come with no velocity information, we estimate a detection's velocity as the mean velocity between itself and its nearest neighbor in each of the neighboring frames. An example is show in Figure 5.13(a).



(a) (b)

FIGURE 5.13: (a) Velocity estimation of the blue detection for $m = 3$. Circles are detections, the horizontal dimension is time, and the vertical one stands for 2D space. Green detections are the nearest detections in space to the blue detection for each k. Detections in grey are not considered for velocity estimation. Detection p_{-1} is discarded because the speed required to reach the blue detection from it exceeds a predefined limit. The green vectors are the velocities computed for each blue-green detection pair and the blue vector is the estimated velocity. (b) Circles enclose disjoint space-time groups, found from assumed bounds on walking speed.

For efficiency we use hierarchical agglomeration (Anderberg, 1973) to group detections within each interval into *space-time groups* (Figure 5.13(b)). This allows solving the correlation clustering problem separately for each space-time group, rathern than solving one big instance which contains all detections in the interval.

In hierarchical agglomeration each detection is initially placed in a separate group. The algorithm then repeatedly merges the pair of groups that are closest to each other in space until k_i space-time groups are formed for time interval i. We set k_i to one half of the expected number of visible people in the given time interval, estimated as the ratio between the total number of detections and the number of frames in the interval. Because of the conservative choice of k_i, it is unlikely that observations that belong together end up in different groups. Even if they do, one person will end up split into different identities, and the single-camera trajectory stage, described later, has an opportunity to undo the split.

5.2.5.2 Single-Camera Trajectories

We compute single-camera trajectories in windows of 10 seconds. The length of the window should be longer than the typical occlusion length to correctly recover the identity of the target after it reappears. For the extraction of the appearance descriptor and position information, we use the middle detection of each tracklet as reference.

5.2.5.3 Multi-Camera Identities

The top level of the hierarchy estimates multi-camera identities from single-camera trajectories. The temporal window length for association should be sufficiently long to capture typical cross-camera transitions. We set this window to be 1.5 minutes. The appearance feature of each single-camera trajectory is the average of the features of its tracklets. At this level we switch the motion correlation estimation from the image plane to the ground plane to provide more meaningful evidence.

5.2.6 Post-Processing

We employ post-processing to interpolate and smooth all trajectory estimates as well as discard spurious tracks. At the tracklet level, we fit a polynomial of degree 2 to all bounding box coordinates of detections to produce smooth bounding box tracklets. To smooth trajectories we linearly interpolate all bounding box coordinates between tracklet centers over time. For discarding spurious tracks, we remove all single- and multi-camera trajectories shorter than 2 seconds due to their low confidence.

5.2.7 Unlimited Time Horizon

We employ a temporal sliding window to associate tracklets into single-camera trajectories, and single-camera trajectories into multi-camera identities. The window spans 10 seconds for single-camera trajectories, and 1.5 minutes for multi-camera identities.

Every time a window is processed, its estimated partial trajectories are fixed, the window is advanced by 50% of its length and the process is repeated. Processing of a window consists of running correlation clustering and trajectory post-processing. All the trajectories that are at least partially contained in the first window are considered for association. Partial trajectories are never undone, but they can be extended from data in subsequent windows.

5.3 Experiments

We run several experiments on benchmarks for MTMCT and Re-ID. (a) We measure overall MTMCT performance, (b) measure the impact of improved detector and features during tracking, (c) study the relation between measures of accuracy for ranking and tracking, and (d) analyze tracker failures.

We benchmark our method on the DukeMTMC tracking benchmark described in Chapter 4. We use the 25 minute long *test-easy* sequence and 15 minute long *test-hard* sequence hosted on MOTChallenge for testing. *test-hard* features a large group of people traveling through 4 cameras. We use the first 30 minutes of the training set for training and the 17 minute long validation sequence for ablation experiments.

Code implementation for the tracker is publicly available[3].

5.3.1 Evaluation

For MTMCT evaluation we use the ID measures of performance introduced in Chapter 3 which indicate how well a tracker identifies who is where regardless of where or why mistakes occur. We use Identity Precision (IDP), Identity Recall (IDR), and Identity F_1-score. IDP (IDR) is the fraction of computed (true) detections that are correctly identified. IDF_1 is the ratio of correctly identified detections over the average number of true and computed detections. IDF_1 is used as the principal measure

[3] `http://vision.cs.duke.edu/DukeMTMC`

for ranking MTMC trackers. ID measures first compute a 1-1 mapping between true and computed identities that maximizes true positives, and then compute the ID scores.

For single-camera evaluation we also report the Multi Object Tracking Accuracy (MOTA), which counts mistakes by how often, not how long, incorrect decisions are made. MOTA is based on the CLEAR-MOT mapping of Bernardin and Stiefelhagen (2008) which under-reports multi-camera errors, therefore we report it only in single camera experiments.

5.4 Results

We discuss results for MTMC tracking, where our proposed method outperforms previous and concurrent work in IDF_1 score and identity recall IDR; study the influence of different components; and analyze typical tracking failures.

5.4.1 Impact of Learning

We evaluate how detector and feature choice impact multi-camera IDF_1 on the DukeMTMC validation set. Results are shown in Table 5.1. Results for single-camera appear in the text and when relevant.

Table 5.1: Impact of improving detector and features on multi-camera performance for the validation sequence.

	IDF_1	IDP	IDR
DPM + HSV (BIPCC)	54.98	62.67	48.97
OpenPose + HSV	58.24	60.60	56.06
DPM + ResNet	65.68	74.87	58.50
OpenPose + ResNet (**DeepCC**)	**80.26**	**83.50**	**77.25**

First we compare the behavior of our baseline method BIPCC with and without deep features. BIPCC uses part based color histograms as appearance features. Our

learned features play an important role in improving IDF_1 by 10.7 points (third row) in multi-camera performance.

Second we measure the impact of the deep learned detector. We substituted the baseline's DPM detections (first row) with those obtained from OpenPose (second row). Although single-camera IDF_1 increases from 75.0 to 85.5, multi-camera IDF_1 increases by only 3.26 points (from 54.98 to 58.24%). This indicates that the detector plays an important role in single-camera tracking by reducing false negatives, but in multi-camera tracking weak features take little advantage of better single-camera trajectories.

These results imply that good features are crucial for MTMC tracking, and that a good detector is most useful for improving single-camera performance. The best MTMCT performance is achieved by combining both.

5.4.2 MTMC Tracking

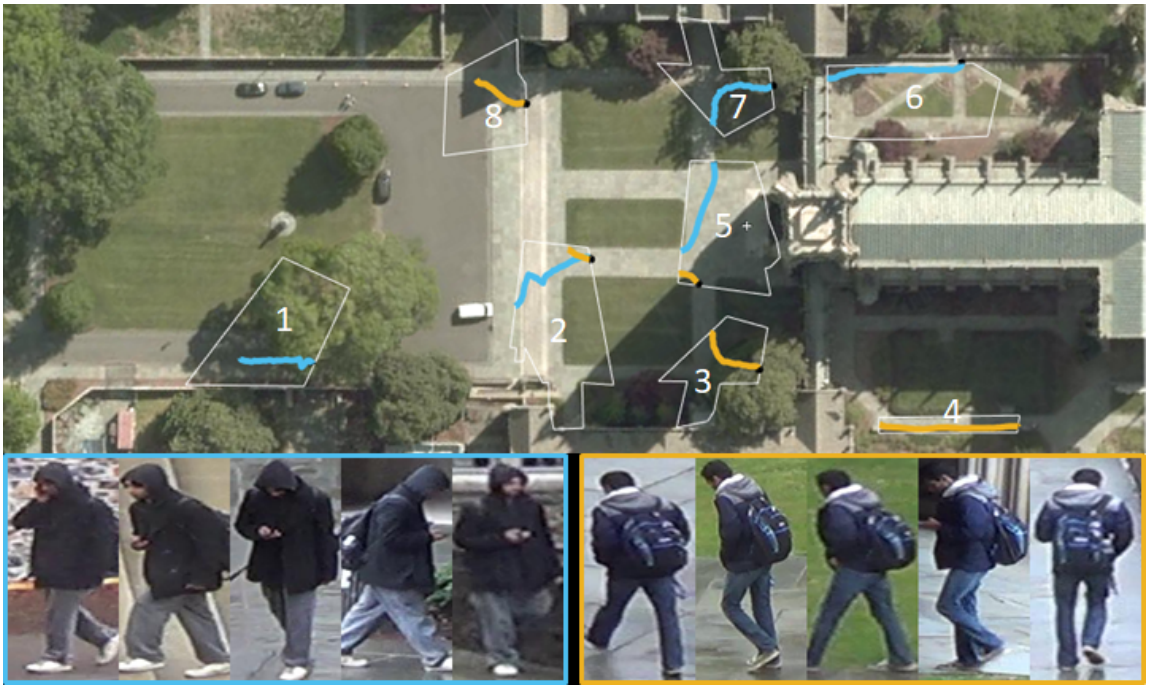


FIGURE 5.14: Two example multi-camera results from our tracker on the DukeMTMC dataset

Table 5.2: DukeMTMCT results. Methods in † are unrefereed submissions.

	Multi-Camera Easy			Multi-Camera Hard			Single-Camera Easy				Single-Camera Hard			
	IDF_1	IDP	IDR	IDF_1	IDP	IDR	IDF_1	IDP	IDR	MOTA	IDF_1	IDP	IDR	MOTA
BIPCC	56.2	67.0	48.4	47.3	59.6	39.2	70.1	83.6	60.4	59.4	64.5	81.2	53.5	54.6
Liang and Zhou (2017)	58.0	72.6	48.2	48.3	60.6	40.2	70.3	88.1	58.5	61.3	64.2	80.4	53.4	53.6
Maksai et al. (2017)	-	-	-	-	-	-	71.2	84.8	61.4	59.3	65.0	81.8	54.0	54.4
Tesfaye et al. (2017)	60.0	68.3	53.5	50.9	63.2	42.6	77.0	87.6	68.6	70.9	65.5	81.4	54.7	59.6
Yoon et al. (2018)	64.8	70.8	59.8	47.3	55.6	41.2	80.0	87.5	73.8	77.7	63.4	74.5	55.2	59.0
Zhang et al. (2017b)†	78.3	82.6	74.3	67.7	**78.6**	59.4	86.3	91.2	82.0	83.6	77.6	**90.1**	68.1	69.6
DeepCC	**82.0**	**84.3**	**79.8**	**68.5**	75.8	**62.4**	**89.2**	**91.7**	**86.7**	**87.5**	**79.0**	87.4	**72.0**	**70.0**

Table 5.3: Detailed DukeMTMCT single-camera tracking results for the *test-easy* and *test-hard* sequences. Methods in † are unrefereed submissions.

	MOTA						IDP						IDR						IDF_1					
	BIPCC	Maksai et al. (2017)	Tesfaye et al. (2017)	Yoon et al. (2018)	Zhang et al. (2017b)†	**DeepCC**	BIPCC	Maksai et al. (2017)	Tesfaye et al. (2017)	Yoon et al. (2018)	Zhang et al. (2017b)†	**DeepCC**	BIPCC Ristani et al. (2016)	Maksai et al. (2017)	Tesfaye et al. (2017)	Yoon et al. (2018)	Zhang et al. (2017b)†	**DeepCC**	BIPCC Ristani et al. (2016)	Maksai et al. (2017)	Tesfaye et al. (2017)	Yoon et al. (2018)	Zhang et al. (2017b)†	**DeepCC**
Easy-all	59.4	59.3	70.9	77.7	83.6	**87.5**	83.6	84.8	87.6	87.5	91.2	**91.7**	60.4	61.4	68.6	73.8	82.0	**86.7**	70.1	71.2	77.0	80.0	86.3	**89.2**
Cam1	43.0	42.9	69.9	84.9	87.4	**93.3**	91.2	91.9	89.1	89.7	91.1	**95.6**	41.8	42.2	67.7	79.6	86.2	**93.0**	57.3	57.8	76.9	84.3	88.6	**94.3**
Cam2	44.8	44.7	71.5	78.4	84.2	**87.1**	69.3	70.4	90.9	88.9	92.4	**93.6**	67.1	68.0	73.4	75.9	82.9	**87.4**	68.2	69.2	81.2	81.9	87.4	**90.4**
Cam3	57.8	57.8	67.4	65.7	**82.4**	79.7	78.9	78.2	76.3	76.2	**87.8**	86.2	48.8	48.4	56.0	63.5	**79.7**	77.7	60.3	59.8	64.6	69.3	**83.6**	81.8
Cam4	63.2	63.2	76.8	79.8	**91.9**	91.8	88.7	91.7	91.2	84.1	**97.7**	96.3	62.8	64.9	79.0	77.6	93.1	**94.4**	73.5	76.0	84.7	80.7	**95.4**	95.3
Cam5	72.8	72.6	68.9	76.6	80.8	**86.2**	83.0	83.0	76.1	81.4	**87.2**	83.6	65.4	65.6	61.9	67.3	75.8	**77.7**	73.2	73.3	68.3	73.7	**81.1**	80.6
Cam6	73.4	73.4	77.0	82.8	83.1	**88.7**	87.5	91.7	91.6	88.9	91.7	**93.4**	69.1	72.4	75.3	78.8	82.5	**92.2**	77.2	80.9	82.7	83.5	86.9	**92.8**
Cam7	71.4	71.4	73.8	77.0	80.8	**82.2**	93.6	93.6	**94.0**	91.4	92.8	93.7	70.6	70.6	72.5	73.5	80.1	**83.7**	80.5	80.5	81.8	81.5	86.0	**88.5**
Cam8	60.7	60.9	63.4	71.6	79.9	**85.0**	**92.2**	**92.2**	89.1	90.8	91.1	89.4	59.6	60.0	61.8	71.3	78.6	**82.4**	72.4	72.7	73.0	79.9	84.4	**85.8**
Hard-all	54.6	54.4	59.6	59.0	69.6	**70.0**	81.2	81.8	81.4	74.5	**90.1**	87.4	53.5	54.0	54.7	55.2	68.1	**72.0**	64.5	65.0	65.5	63.4	77.6	**79.0**
Cam1	37.8	37.4	63.2	61.1	74.4	**79.6**	92.5	91.9	83.0	72.2	92.3	**94.7**	36.8	36.7	56.4	58.4	76.1	**80.1**	52.7	52.5	67.1	64.6	83.4	**86.8**
Cam2	47.3	46.6	54.8	50.4	**70.9**	57.9	65.7	66.0	78.8	61.2	**89.1**	77.5	56.1	56.7	53.1	52.6	66.7	**67.3**	60.6	61.0	63.4	56.6	**76.3**	72.0
Cam3	46.7	46.7	68.8	70.3	**87.1**	84.2	**96.1**	96.1	91.1	86.9	94.9	90.8	46.5	46.5	73.7	74.1	**89.2**	87.1	62.7	62.7	81.5	80.0	**91.9**	88.9
Cam4	85.3	85.5	75.6	81.2	**95.0**	90.3	86.0	93.6	87.1	84.4	**97.3**	93.0	82.7	91.0	78.1	82.2	**97.7**	97.0	84.3	92.3	82.3	83.3	**97.5**	94.9
Cam5	78.3	78.3	78.6	81.9	77.2	**86.0**	90.1	90.1	91.5	**93.3**	88.4	90.9	75.1	75.1	75.7	79.2	75.3	**85.5**	81.9	81.9	82.8	85.7	81.3	**88.1**
Cam6	59.4	59.4	53.3	56.1	58.4	**63.3**	81.7	82.4	71.2	70.0	86.3	**87.0**	52.7	53.3	42.3	44.9	55.4	**62.2**	64.1	64.7	53.1	54.7	67.5	**72.5**
Cam7	50.8	50.6	50.8	49.8	60.3	**61.4**	81.2	81.4	84.7	74.7	**91.4**	85.2	47.1	47.2	47.1	44.4	59.7	**61.3**	59.6	59.8	60.6	55.7	**72.2**	71.3
Cam8	73.0	73.0	70.0	71.5	**85.6**	85.0	**94.9**	94.9	90.3	93.5	92.2	92.3	72.8	72.8	73.9	70.5	83.7	**87.7**	82.4	82.4	81.3	80.4	87.7	**89.9**

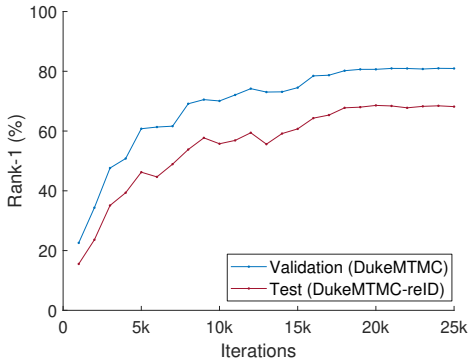


FIGURE 5.15: Relation between number of training iterations and rank-1 accuracy for validation and test sets

Overall results are presented in Tables 5.2 and 5.3. Our method DeepCC improves the multi-camera IDF_1 accuracy w.r.t to the previous state of the art of Tesfaye et al. (2017) by 22 and 17.6 points for the *test-easy* and *test-hard* sequences, respectively. For the single-camera easy and hard sequences, the IDF_1 improvement is 12.2 and 13.5 points, and MOTA improves by 16.6 and 10.4 points. Two multi-camera results from our tracker are shown in Figure 5.14.

Compared to unrefereed submissions, we perform slightly worse on IDP on the hard sequence. This could be due to a choice of detector that works better for crowded scenarios, a detector that is more conservative, and/or more conservative association. We nonetheless outperform all methods on IDF_1, IDR and MOTA.

It is worth noting that our method achieves the highest identity recall IDR on all scenarios, and on nearly all single-camera sequences. Identity recall is Achille's heel for modern multi-target trackers, as they commonly fail to re-identify targets after occlusions. This issue has also been discussed by Leal-Taixé et al. (2017). We believe that this improvement is a combination of better detections, joint optimization, and a discriminative feature embedding.

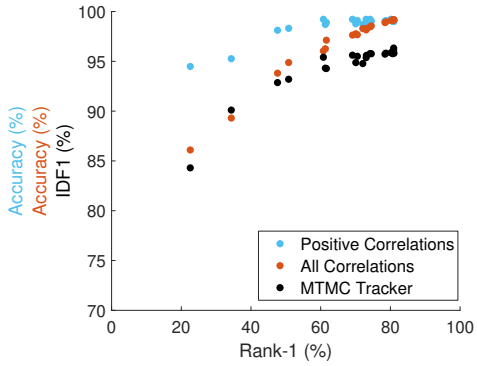


FIGURE 5.16: Relation of tracking, correlation, and rank accuracy

5.4.3 Accuracy of Tracking vs. Ranking

MTMCT and ReID differ subtly but fundamentally, because ReID *ranks* distances to a query while MTMCT *classifies* a pair of images as being co-identical or not, and their performance is consequently measured by different metrics: ranking performance for ReID, classification error rates for MTMCT. As more and more re-identification methods are being applied to multi-target tracking, we study the relation between ID measures for MTMC tracking and rank measures for ReID. In this experiment, we use ground truth single-camera trajectories and perform across-camera tracking with features at different times during training, resulting in different levels of ranking accuracy. Appearance features are learned from scratch using the 461 DukeMTMC-reID training IDs that do not appear in the validation sequence. Tracking accuracy is evaluated on the DukeMTMC validation sequence (241 IDs), and rank-1 accuracy on both DukeMTMC-reID test (702 IDs) and DukeMTMC validation. Results are shown in Figures 5.15-5.16.

We observe the following: Figure 5.15: Rank-1 accuracy for DukeMTMC-reID test and DukeMTMC validation correlate and we observe a similar pattern in how rank-1 accuracy improves with more training iterations. Figure 5.16: (a) Features with modest rank-1 performance can still do well in MTMCT because of more limited and diverse identities to compare between, and because tracking is also helped by

motion information. (b) MTMCT IDF_1 performance improves with rank-1 accuracy. However, after a point, further improvement in rank-1 accuracy yields diminishing returns in IDF_1.

Our interpretation for this saturation effect is as follows. Initially, the Re-ID model learns to separate positive and negative samples, and tracking performance increases linearly with rank-1 performance. Once enough correlations have the correct sign, correlation clustering can infer the remaining missing agreements by enforcing transitivity (inequality 5.4). Therefore, correcting the sign of the remaining correlations has a smaller effect on IDF_1. Even beyond that point, the Re-ID model tries to satisfy the separation margin of the Adaptive Weighted Triplet loss L_3 (Equation 2.1) by further pulling co-identical samples together and non co-identical ones apart. These changes do not affect the correlation signs and have little influence on IDF_1.

5.4.4 Weakness Analysis

We analyze the one-to-one ID mapping between true and computed trajectories to understand failures on the DukeMTMC validation sequence. During evaluation, each true trajectory that is mapped to an actual computed trajectory (not a false positive) has its own ID recall, as some of its detections could be missed by the tracker. Similarly, the computed trajectories have their own precision, as they can contain false positive detections.

We rank computed trajectories by ID precision and true trajectories by ID recall, then inspect the trajectories with the lowest scores. This helps clarify which situations are difficult in single- and multi-camera scenarios. Single- and multi-camera scenarios are analyzed separately because their ID mapping is different.

Failure cases are illustrated in Figures 5.18-5.17. In single-camera tracking, correlations are poor when there is significant pose change, significant occlusion, and/or abrupt motion, resulting in low identity recall (left in the Figure). In multi-camera

FIGURE 5.17: Two example multi-camera trajectories with low identity precision

FIGURE 5.18: Example ground truth trajectories with poor identity recall in single camera tracking. Red indicates failure.

tracking, fragmentation is mostly caused by delays in blind spots and unpredictable motion. Merge errors happen in cases where people dress similarly and their inter-camera motion is plausible.

The example in Figure 5.18 (bottom-right) highlights one of the most difficult situations in the validation sequence, where several construction workers share similar appearance. They enter and exit the field of view a few times, and both appearance and motion correlations are weak, resulting in poor identity recall during tracking.

6

Current and Future Work

Multi-Target Multi-Camera tracking remains a very challenging problem in computer vision. During this decade there has been very fast progress on this problem due to general combinatorial formulations and improved optimization methods, supervised deep learning techniques, large amounts of training data, better evaluation metrics, and standardized benchmarks. We have contributed to several of these aspects.

The rapid progress in the field has contributed to exceptional performance in single-camera tracking settings with low person density. Moreover, now there exist commercial grade sport player trackers that work in constrained settings using multiple overlapping camera views, as well person trackers deployed in pilot grocery stores with constrained layouts. Tracking multiple people in crowded scenarios and re-identifying them across cameras remains an open and challenging problem due to frequent occlusions, and all the challenges that come with modeling appearance and motion across cameras. This Chapter speculates on some possible directions for future research.

The notion of relying on large scale datasets for appearance learning which are manually annotated is probably not going to keep scaling to problems of increasing

complexity. An old idea that will soon start to materialize is that of generating large amounts of synthetic data through computer graphics techniques. What has changed in recent years is the quality of photo-realism during rendering which makes this approach appealing, as well as available 3D person models and human motion capture data. Computer graphics techniques allow the generation of person videos from arbitrary viewpoints, lighting conditions, background scenes, varying clothing, and different levels of occlusion without expensive camera setups, and they provide complete knowledge of the ground truth. Related to this aspect, it is not clear that we need large amounts of training data to build effective trackers. This question applies to many computer vision problems that rely on deep learning. Techniques such as one-shot learning which are able to reason from few examples as well as carefully designed, problem specific models, would be able to give preliminary insights on the necessity of large amounts of training data.

In another consideration, an interesting future direction especially during the wave of incorporating 3D geometry in deep learning is to employ parametric 3D body models for person re-identification from video. The 3D body model could include an articulated body model through a skeleton, a 3D surface mesh that is attached to the bones and is adjusted parametrically to match the shape of the person as observed in the video, a texture map that needs to be consistent with the visual features on each video frame, and a motion model that enforces smooth limb motion across frames. Person re-identification can then be reduced to comparing deep visual features mapped onto a canonical body shape as opposed to comparing deep features extracted from 2D images. Estimating parameters in 3 dimensions however can become complicated. An easier approach is to use RGBD sensors that provide depth information, and 3D information through structure from motion could also be useful. The computational price tag that comes with 3D modeling might be nonetheless worth paying. This line of research can lead to more consistent reasoning

about about partial observability and occlusion, provide better temporal feature integration in the texture map, and result in higher matching accuracy. We believe such reasoning will better address cross dataset generalization performance, a severe limitation of existing work.

The idea of reasoning in 3D can also be extended to multi-target tracking. Currently trackers reason about the targets' state through bounding boxes on the image plane, 3D positions on the ground plane, or more recently about person segments in video. If we could reason about the entire scene and people in 3D, accuracy could increase at the expense of additional modeling complexity, and we might be on the verge of affording this added complexity. Occlusion reasoning would be greatly simplified and would lead to consistent tracking after partial or full occlusions. Trackers would reason about re-projection errors on the image plane given a scene estimate, and a good sanity check would be to slightly perturb the 3D position of the real camera in a virtual scene, and expect that the tracker's state estimate of the targets would remain consistent.

Concerning a different aspect, data association, there are two plausible directions for progress that are orthogonal to each other. On one hand, if appearance and motion features become sufficiently good, simple techniques like greedy matching would suffice. This is signaled by recent work, including ours, that improves appearance features and uses simple data association algorithms. On the other hand, evidence is always corrupted by noise, and current techniques are still expensive and consider limited evidence, by-and-large restricted to pairs or triplets of observations. A potential direction is to use a two step algorithm where during the first stage a set of plausible trajectories is proposed, and the second stage finds a maximum weight set cover on the detections using the proposed trajectories. The quality of the set can be learned or modeled, and is more general than the pairwise evidence typically used.

Regarding evaluation, the community is still lacking a fast way to diagnose track-

ers. A researcher typically cycles through a first phase of idea prototyping, followed by bottom-line performance evaluation, visual inspection of videos for mistakes, code/data inspection for errors, and finally tracker refinement. Hundreds of hours of research time would be saved if there exists a diagnostic tool that evaluates a tracker's performance, automatically renders trajectories in the video along with ground truth, points the researcher back to where mistakes occur, and provides sufficient diagnostic information interactively, both visually and quantitatively, on why incorrect associations were made.

Another interesting direction is learning multi-target tracking in a weakly supervised or completely unsupervised fashion. Assuming a static background and a fixed camera that observes the same environment for a long period of time, it is possible to focus attention only on moving objects. Person appearance models could be learned by exploiting the large amounts of negative examples from simultaneous observations, or assuming that observations too distant in time are likely to belong to different people. Positive examples can be collected conservatively either through isolated trajectories, or through observations that are close in space and time. This data collection process can allow training an appearance model with little or no supervision, not requiring manual annotation. Such trackers could be validated on large scale tracking datasets like ours that provide ground truth. A more challenging problem is how to extend this idea to tracking across cameras, where collecting positive examples from different views is not straightforward without making some assumptions.

A curious idea to pursue is re-identification of people not by how they look, but how they walk. This is relevant to applications where the same people are observed during different days and wear different clothes, making appearance-based methods infeasible. It is worth exploring the gait pattern of a person's skeleton motion to learn appearance invariant features.

In the long term MTMC tracking computation ought to efficient to support running the tracker on low-power compute devices embedded in the cameras. Current trackers that rely on combinatorial optimization are not adequate for large-scale deployment in unconstrained settings without expensive infrastructure. Cameras should be self sufficient by being able to track targets themselves, removing the necessity to upload all video streams on one server but rather communicate only the person trajectories and necessary appearance models.

7

Conclusions

In this dissertation we have contributed to several aspects of the Multi-Target, Multi-Camera Tracking problem that have lead to accelerated progress in the field. Below we summarize our work and insights in chronological order.

The work behind this dissertation began in 2013 by framing MTMCT as a correlation clustering problem. Due to unavailable video benchmarks with more than 3 cameras at the time, our research effort focused on showing the benefits of correlation clustering in single-camera settings. The formulation proved to be significantly more accurate than methods which computed paths of maximal internal weights, and also more accurate than trackers which considered all pairwise evidence but used suboptimal solvers. We were the first to officially formulate MTT as a correlation clustering problem in Ristani and Tomasi (2014) and showed that our formulation was state of the art in single-camera tracking, without using deep learning.

One of the main insights from this work is that partitioning the correlations' graph of a large tracking instance into smaller subgraphs conservatively, solving for the small instances optimally and combining their solutions post-facto, results in improved accuracy compared to solving the full graph with approximate algorithms.

To this end, we have found that if the graphs are restricted to no more than 150 nodes then our tracker can efficiently process video frames at a 30 fps rate. Combinatorial methods that followed ours have also converged to our approach to tracking: They use an optimal solver for small graph instances and combine the solutions post-facto.

Due to a lack of multi-camera benchmarks with complex topology, in 2014 we made preparations to collect our own video sequences. We initially collected multi-camera sequences through computer graphics simulation of people and soon switched to deploying 9 cameras on the Duke University campus. This was motivated by the lack of realism in the synthetic sequences. Two major developments were occurring in 2015 at the time when we determined the camera placement and the time of filming for our data: MOTChallenge had come into existence and was unifying the protocol of multi-target tracking benchmarking; data-hungry deep learning methods on the other hand had began to gain traction. Our ambition to create a very large tracking benchmark was already in line with these developments, and the main challenge was to obtain high quality trajectory annotations with minimal effort. After we had collected enough enough multi-camera annotations to benchmark of our tracker, we realized that existing evaluation protocols were not aligned with our expectations of performance evaluation. In particular, existing measures were not helping showcase the main benefit of correlation clustering, correct identification. Thus we revisited our old ideas on improving performance evaluation and the effort converged with the identity measures of performance. The new data set, identity measures of performance, and correlation clustering multi-target multi-camera tracker were published in Ristani et al. (2016).

During this work we learned several lessons. The first pertains to performance measures. It is difficult to convince a research community that is already accustomed to a set of measures to accept a new evaluation paradigm. On the other hand the community always appreciates efforts gone into large-scale data sets. For data

annotation, we learned that existing annotation tools are not always adequate for new tasks. For instance the VATIC tool was not designed with multiple cameras in mind. Our takeaway message is that it is always best to invest time in a new tool if it is going to significantly reduce the annotation effort. We also learned that the annotation process is very slow if high annotation quality is a priority. In terms of video recording, we learned that filming outdoors is weather dependent and it is not always possible to predict the day with the ideal conditions. Camera placement is also a strong factor in determining the recording location, where some of the main challenges include gaining access to buildings at the desired time, synchronizing disjoint views, and obtaining permission to record people.

In late 2016 our main research effort was focused towards bridging the gap between the multi-target tracking and person re-identification communities. We specifically organized a workshop at CVPR 2017 to promote the cross-fertilization between the two fields. We promoted the idea that the two problems share many similarities and should not be solved separately, and in our most recent work Ristani and Tomasi (2018) we demonstrated that is it possible to learn appearance features for both MTMCT and ReID using the same techniques, an adaptive weighted triplet loss and hard negative mining. By embracing deep learning, our features give state of the art results on both MTMCT and ReID benchmarks. Also, using deep learning in tracking both in the person detection stage and appearance feature learning, our correlation clustering tracker significantly improves its older version that doesn't use deep learning, both in single- and multi-camera tracking, setting a new state of the art.

One of the main insights from this work is that in tracking the fraction of correlations with correct sign matters more than how strong correlations are. This is immediately reflected on the fact that models with weaker rank-1 ReID performance that produce a similar number of correct correlations achieve similar tracking perfor-

mance. We also found that in tracking, improved detections do not give considerable improvements in across-camera accuracy when features are weak, they help mostly with single-camera tracking. This is because in across-camera association motion information is weak, so a weak appearance model is very limited in making correct associations.

During the CVPR 2017 MTMCT and ReID workshop[1] we shared our view that the MTMCT and ReID problems are clearly different. First, the gallery in target re-identification may or may not be a set of videos and both targets and queries are assumed to have been isolated ahead of time. The input to multi-target multi-camera tracking on the other hand is necessarily a set of videos. Second, multi-target multi-camera tracking matches all targets symmetrically while target re-identification distinguishes between query and response explicitly. However, MTMCT and ReID share several common aspects as well. They assume a semantic notion of "identity" in that only ground truth can tell if the targets in two bounding boxes share the same identity. They could both benefit from a detector that separates targets from non-targets. When the gallery in target re-identification is a set of videos, both problems can use similar pre-processing techniques. These might rely on knowing the camera topology, considering time elapsed between observations, modeling changes in illumination and viewpoint across cameras, etc. Both problems require annotated databases of videos or images and some databases can work for both problems. Some components of the solution to either problem can be used to solve the other.

To conclude, we have shared our code and data for easier reproducibility of this work and for the pursuit of several future directions, some of which this dissertation has enabled. Multi-Target Multi-Camera Tracking remains a very challenging problems in computer vision with significant room for new advances. We hope that new large-scale data sets will be introduced to further validate our ideas, and in the near

[1] `https://reid-mct.github.io/`

future simple and effective methods will address existing limitations of MTMCT and ReID uniformly.

Bibliography

Almazan, J., Gajic, B., Murray, N., and Larlus, D. (2018), "Re-ID done right: towards good practices for person re-identification," *arXiv preprint arXiv:1801.05339*.

Anderberg, M. R. (1973), "Cluster analysis for applications," Tech. rep., DTIC Document.

Andriyenko, A., Schindler, K., and Roth, S. (2012), "Discrete-Continuous Optimization for Multi-Target Tracking," in *CVPR*.

Ayazoglu, M., Li, B., Dicle, C., Sznaier, M., and Camps, O. (2011), "Dynamic subspace-based coordinated multicamera tracking," in *2011 IEEE International Conference on Computer Vision (ICCV)*, pp. 2462–2469.

Bagon, S. and Galun, M. (2011), "Large scale correlation clustering optimization," *arXiv preprint arXiv:1112.2903*.

Baltieri, D., Vezzani, R., and Cucchiara, R. (2013), "Learning Articulated Body Models for People Re-identification," in *Proceedings of the 21st ACM International Conference on Multimedia*, MM '13, pp. 557–560, New York, NY, USA, ACM.

Baltieri, D., Vezzani, R., and Cucchiara, R. (2015), "Mapping Appearance Descriptors on 3D Body Models for People Re-identification," *International Journal of Computer Vision*, 111, 345–364.

Bansal, N., Blum, A., and Chawla, S. (2002), "Correlation clustering," in *Foundations of Computer Science*.

Barbosa, I. B., Cristani, M., Caputo, B., Rognhaugen, A., and Theoharis, T. (2017), "Looking beyond appearances: Synthetic training data for deep cnns in re-identification," *arXiv preprint arXiv:1701.03153*.

Bedagkar-Gala, A. and Shah, S. (2011), "Multiple person re-identification using part based spatio-temporal color appearance model," in *Computer Vision Workshops (ICCV Workshops), 2011 IEEE International Conference on*, pp. 1721–1728.

Bedagkar-Gala, A. and Shah, S. K. (2012), "Part-based spatio-temporal model for multi-person re-identification," *Pattern Recognition Letters*, 33, 1908 – 1915, Novel Pattern Recognition-Based Methods for Re-identification in Biometric Context.

Berclaz, J., Fleuret, F., Türetken, E., and Fua, P. (2011), "Multiple Object Tracking using K-Shortest Paths Optimization," *IEEE Transactions on Pattern Analysis and Machine Intelligence*.

Bernardin, K. and Stiefelhagen, R. (2008), "Evaluating multiple object tracking performance: the CLEAR MOT metrics," *EURASIP Journal on Image and Video Processing*, pp. 1–10.

Besag, J. (1986), "On the statistical analysis of dirty pictures," *Journal of the Royal Statistical Society. Series B (Methodological)*, pp. 259–302.

Betke, M. and Wu, Z. (2016), "Data association for multi-object visual tracking," *Synthesis Lectures on Computer Vision*, 6, 1–120.

Bredereck, M., Jiang, X., Korner, M., and Denzler, J. (2012), "Data association for multi-object Tracking-by-Detection in multi-camera networks," in *2012 Sixth International Conference on Distributed Smart Cameras (ICDSC)*, pp. 1–6.

Brendel, W., Amer, M., and Todorovic, S. (2011), "Multiobject tracking as maximum weight independent set," in *Computer Vision and Pattern Recognition (CVPR), 2011 IEEE Conference on*, pp. 1273–1280, IEEE.

Butt, A. A. and Collins, R. T. (2013), "Multiple target tracking using frame triplets," in *Computer Vision–ACCV 2012*, pp. 163–176, Springer.

Cai, Y. and Medioni, G. (2014), "Exploring context information for inter-camera multiple target tracking," in *2014 IEEE Winter Conference on Applications of Computer Vision (WACV)*, pp. 761–768.

Calderara, S., Cucchiara, R., and Prati, A. (2008), "Bayesian-Competitive Consistent Labeling for People Surveillance," *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 30, 354–360.

Cao, L., Chen, W., Chen, X., Zheng, S., and Huang, K. (2015), "An equalised global graphical model-based approach for multi-camera object tracking," *ArXiv:11502.03532 [cs]*.

Cao, Z., Simon, T., Wei, S.-E., and Sheikh, Y. (2017), "Realtime Multi-Person 2D Pose Estimation using Part Affinity Fields," in *CVPR*.

Chari, V., Lacoste-Julien, S., Laptev, I., and Sivic, J. (2015), "On Pairwise Costs for Network Flow Multi-Object Tracking," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 5537–5545.

Chen, K.-W., Lai, C.-C., Lee, P.-J., Chen, C.-S., and Hung, Y.-P. (2011a), "Adaptive Learning for Target Tracking and True Linking Discovering Across Multiple Non-Overlapping Cameras," *IEEE Transactions on Multimedia*, 13, 625–638.

Chen, W., Chen, X., Zhang, J., and Huang, K. (2017a), "Beyond triplet loss: a deep quadruplet network for person re-identification," in *The Conference on Computer Vision and Pattern Recognition*.

Chen, X., Huang, K., and Tan, T. (2011b), "Direction-based stochastic matching for pedestrian recognition in non-overlapping cameras," in *2011 18th IEEE International Conference on Image Processing (ICIP)*, pp. 2065–2068.

Chen, X., An, L., and Bhanu, B. (2015), "Multitarget Tracking in Nonoverlapping Cameras Using a Reference Set," *IEEE Sensors Journal*, 15, 2692–2704.

Chen, Y., Zhu, X., and Gong, S. (2017b), "Person re-identification by deep learning multi-scale representations," .

Cheng, D. and Cristani, M. (2014), "Person Re-identification by Articulated Appearance Matching," in *Person Re-Identification*, eds. S. Gong, M. Cristani, S. Yan, and C. C. Loy, Advances in Computer Vision and Pattern Recognition, pp. 139–160, Springer London.

Cheng, D., Cristani, M., Stoppa, M., Bazzani, L., and Murino, V. (2011), "Custom Pictorial Structures for Re-identification," in *Proceedings of the British Machine Vision Conference*, pp. 68.1–68.11, BMVA Press, http://dx.doi.org/10.5244/C.25.68.

Daliyot, S. and Netanyahu, N. S. (2013), "A Framework for Inter-camera Association of Multi-target Trajectories by Invariant Target Models," in *Computer Vision - ACCV 2012 Workshops*, eds. J.-I. Park and J. Kim, no. 7729 in Lecture Notes in Computer Science, pp. 372–386, Springer Berlin Heidelberg.

Das, A., Chakraborty, A., and Roy-Chowdhury, A. K. (2014), "Consistent re-identification in a camera network," in *Computer Vision–ECCV 2014*, pp. 330–345, Springer.

De Vleeschouwer, C., Chen, F., Delannay, D., Parisot, C., Chaudy, C., Martrou, E., Cavallaro, A., et al. (2008), "Distributed video acquisition and annotation for sport-event summarization," in *NEM summit 2008:: Towards Future Media Internet*.

Dehghan, A., Assari, S. M., and Shah, M. (2015), "GMMCP Tracker: Globally Optimal Generalized Maximum Multi Clique Problem for Multiple Object Tracking," in *CVPR*, vol. 1, p. 2.

Demaine, E. D., Emanuel, D., Fiat, A., and Immorlica, N. (2006), "Correlation clustering in general weighted graphs," *Theoretical Computer Science*, 361, 172–187.

D'Orazio, T., Leo, M., Mosca, N., Spagnolo, P., and Mazzeo, P. L. (2009), "A semi-automatic system for ground truth generation of soccer video sequences," in *Advanced Video and Signal Based Surveillance, 2009. AVSS'09. Sixth IEEE International Conference on*, pp. 559–564, IEEE.

Felzenszwalb, P. F., Girshick, R. B., McAllester, D., and Ramanan, D. (2010), "Object detection with discriminatively trained part-based models," *IEEE transactions on pattern analysis and machine intelligence*, 32, 1627–1645.

Ferryman, J. and Shahrokni, A. (2009), "Pets2009: Dataset and challenge," in *Performance Evaluation of Tracking and Surveillance (PETS-Winter), 2009 Twelfth IEEE International Workshop on*, pp. 1–6, IEEE.

Fiduccia, C. M. and Mattheyses, R. M. (1988), "A linear-time heuristic for improving network partitions," in *Papers on Twenty-five years of electronic design automation*, pp. 241–247, ACM.

Fleuret, F., Berclaz, J., Lengagne, R., and Fua, P. (2008), "Multi-Camera People Tracking with a Probabilistic Occupancy Map," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 30, 267–282.

Gilbert, A. and Bowden, R. (2006), "Tracking Objects Across Cameras by Incrementally Learning Inter-camera Colour Calibration and Patterns of Activity," in *Computer Vision ECCV 2006*, eds. A. Leonardis, H. Bischof, and A. Pinz, no. 3952 in Lecture Notes in Computer Science, pp. 125–136, Springer Berlin Heidelberg.

Hadsell, R., Chopra, S., and LeCun, Y. (2006), "Dimensionality reduction by learning an invariant mapping," in *Computer vision and pattern recognition, 2006 IEEE computer society conference on*, vol. 2, pp. 1735–1742, IEEE.

Hamid, R., Kumar, R., Grundmann, M., Kim, K., Essa, I., and Hodgins, J. (2010), "Player localization using multiple static cameras for sports visualization," in *2010 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 731–738.

He, K., Zhang, X., Ren, S., and Sun, J. (2016), "Deep residual learning for image recognition," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 770–778.

Henriques, J. F., Caseiro, R., and Batista, J. (2011), "Globally optimal solution to multi-object tracking with merged measurements," in *Computer Vision (ICCV), 2011 IEEE International Conference on*, pp. 2470–2477, IEEE.

Henschel, R., Leal-Taix, L., Cremers, D., and Rosenhahn, B. (2017), "Improvements to Frank-Wolfe optimization for multi-detector multi-object tracking," *arXiv:1705.08314*.

Hermans, A., Beyer, L., and Leibe, B. (2017), "In Defense of the Triplet Loss for Person Re-Identification," *arXiv preprint arXiv:1703.07737*.

Insafutdinov, E., Pishchulin, L., Andres, B., Andriluka, M., and Schiele, B. (2016), "Deepercut: A deeper, stronger, and faster multi-person pose estimation model," in *European Conference on Computer Vision*, pp. 34–50, Springer.

Javed, O., Shafique, K., Rasheed, Z., and Shah, M. (2008), "Modeling inter-camera space–time and appearance relationships for tracking across non-overlapping views," *Computer Vision and Image Understanding*, 109, 146–162.

Jiuqing, W. and Li, L. (2013), "Distributed optimization for global data association in non-overlapping camera networks," in *2013 Seventh International Conference on Distributed Smart Cameras (ICDSC)*, pp. 1–7.

Kalayeh, M. M., Basaran, E., Gokmen, M., Kamasak, M. E., and Shah, M. (2018), "Human Semantic Parsing for Person Re-identification," *arXiv preprint arXiv:1804.00216*.

Kamal, A., Farrell, J., and Roy-Chowdhury, A. (2013), "Information Consensus for Distributed Multi-target Tracking," in *2013 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 2403–2410.

Kasturi, R., Goldgof, D., Soundararajan, P., Manohar, V., Garofolo, J., Bowers, R., Boonstra, M., Korzhova, V., and Zhang, J. (2009), "Framework for performance evaluation of face, text, and vehicle detection and tracking in video: Data, metrics, and protocol," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 31, 319–336.

Kernighan, B. W. and Lin, S. (1970), "An efficient heuristic procedure for partitioning graphs," *The Bell system technical journal*, 49, 291–307.

Kuhn, H. W. (1956), "Variants of the Hungarian method for assignment problems," *Naval Research Logistics (NRL)*, 3, 253–258.

Kumar, R., Charpiat, G., and Thonnat, M. (2014), "Multiple Object Tracking by Efficient Graph Partitioning," in *Computer Vision–ACCV 2014*, pp. 445–460, Springer.

Kuo, C.-H., Huang, C., and Nevatia, R. (2010), "Inter-camera Association of Multi-target Tracks by On-Line Learned Appearance Affinity Models," in *Computer Vision ECCV 2010*, eds. K. Daniilidis, P. Maragos, and N. Paragios, no. 6311 in Lecture Notes in Computer Science, pp. 383–396, Springer Berlin Heidelberg.

Layne, R., Hospedales, T. M., and Gong, S. (2012), "Towards person identification and re-identification with attributes," in *European Conference on Computer Vision*, pp. 402–412, Springer.

Leal-Taixé, L., Milan, A., Reid, I., Roth, S., and Schindler, K. (2015), "MOTChallenge 2015: Towards a Benchmark for Multi-Target Tracking," *arXiv:1504.01942 [cs]*, arXiv: 1504.01942.

Leal-Taixé, L., Milan, A., Schindler, K., Cremers, D., Reid, I., and Roth, S. (2017), "Tracking the Trackers: An Analysis of the State of the Art in Multiple Object Tracking," *arXiv preprint arXiv:1704.02781*.

Leichter, I. and Krupka, E. (2013), "Monotonicity and error type differentiability in performance measures for target detection and tracking in video," *IEEE transactions on pattern analysis and machine intelligence*, 35, 2553–2560.

Li, W., Zhu, X., and Gong, S. (2017), "Person re-identification by deep joint learning of multi-loss classification," *arXiv preprint arXiv:1705.04724*.

Liang, Y. and Zhou, Y. (2017), "Multi-camera Tracking Exploiting Person Re-ID Technique," in *International Conference on Neural Information Processing*, pp. 397–404, Springer.

Liao, S., Hu, Y., Zhu, X., and Li, S. Z. (2015), "Person re-identification by local maximal occurrence representation and metric learning," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 2197–2206.

Lin, Y., Zheng, L., Zheng, Z., Wu, Y., and Yang, Y. (2017), "Improving person re-identification by attribute and identity learning," *arXiv preprint arXiv:1703.07220*.

Liu, W., Anguelov, D., Erhan, D., Szegedy, C., Reed, S., Fu, C.-Y., and Berg, A. C. (2016), "Ssd: Single shot multibox detector," in *European conference on computer vision*, pp. 21–37, Springer.

Luo, W., Xing, J., Milan, A., Zhang, X., Liu, W., Zhao, X., and Kim, T.-K. (2014), "Multiple object tracking: A literature review," *arXiv preprint arXiv:1409.7618*.

Makris, D., Ellis, T., and Black, J. (2004), "Bridging the gaps between cameras," in *Proceedings of the 2004 IEEE Computer Society Conference on Computer Vision and Pattern Recognition, 2004. CVPR 2004*, vol. 2.

Maksai, A., Wang, X., Fleuret, F., and Fua, P. (2017), "Non-Markovian Globally Consistent Multi-Object Tracking," in *Proceedings of the IEEE International Conference on Computer Vision*.

Manen, S., Gygli, M., Dai, D., and Van Gool, L. (2017), "PathTrack: Fast Trajectory Annotation with Path Supervision," .

Martinel, N., Micheloni, C., and Foresti, G. L. (2014), "Saliency weighted features for person re-identification," in *Computer Vision-ECCV 2014 Workshops*, pp. 191–208, Springer International Publishing.

Milan, A., Schindler, K., and Roth, S. (2013), "Challenges of ground truth evaluation of multi-target tracking," in *Computer Vision and Pattern Recognition Workshops (CVPRW), 2013 IEEE Conference on*, pp. 735–742, IEEE.

Milan, A., Leal-Taix, L., Schindler, K., and Reid, I. (2015), "Joint Tracking and Segmentation of Multiple Targets," in *CVPR*.

Milan, A., Leal-Taixé, L., Reid, I., Roth, S., and Schindler, K. (2016), "MOT16: A benchmark for multi-object tracking," *arXiv preprint arXiv:1603.00831*.

Milan, A., Rezatofighi, S. H., Dick, A., Reid, I., and Schindler, K. (2017), "Online Multi-Target Tracking using Recurrent Neural Networks," in *AAAI*.

Mishchuk, A., Mishkin, D., Radenovic, F., and Matas, J. (2017), "Working hard to know your neighbor's margins: Local descriptor learning loss," in *Advances in Neural Information Processing Systems*, pp. 4829–4840.

Padberg, M. and Rinaldi, G. (1991), "A branch-and-cut algorithm for the resolution of large-scale symmetric traveling salesman problems," *SIAM review*, 33, 60–100.

Per, J., Kenk, V. S., Mandeljc, R., Kristan, M., and Kovačič, S. (2012), "Dana36: A multi-camera image dataset for object identification in surveillance scenarios," in *Advanced Video and Signal-Based Surveillance (AVSS), 2012 IEEE Ninth International Conference on*, pp. 64–69, IEEE.

Pirsiavash, H., Ramanan, D., and Fowlkes, C. C. (2011), "Globally-optimal greedy algorithms for tracking a variable number of objects," in *Computer Vision and Pattern Recognition (CVPR), 2011 IEEE Conference on*, pp. 1201–1208, IEEE.

Ranjan, R., Castillo, C. D., and Chellappa, R. (2017), "L2-constrained softmax loss for discriminative face verification," *arXiv preprint arXiv:1703.09507*.

Ren, S., He, K., Girshick, R., and Sun, J. (2015), "Faster R-CNN: Towards real-time object detection with region proposal networks," in *Advances in neural information processing systems*, pp. 91–99.

Ristani, E. and Tomasi, C. (2014), "Tracking Multiple People Online and In Real Time," in *ACCV-12th Asian Conference on Computer Vision*, Springer.

Ristani, E. and Tomasi, C. (2018), "Features for Multi-Target Multi-Camera Tracking and Re-Identification," in *Computer Vision and Pattern Recognition (CVPR), 2012 IEEE Conference on*, IEEE.

Ristani, E., Solera, F., Zou, R., Cucchiara, R., and Tomasi, C. (2016), "Performance measures and a data set for multi-target, multi-camera tracking," in *European Conference on Computer Vision Workshops*, pp. 17–35, Springer.

Schroff, F., Kalenichenko, D., and Philbin, J. (2015), "Facenet: A unified embedding for face recognition and clustering," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 815–823.

Schulter, S., Vernaza, P., Choi, W., and Chandraker, M. (2017), "Deep Network Flow for Multi-Object Tracking," *CVPR*.

Schumann, A. and Stiefelhagen, R. (2017), "Person re-identification by deep learning attribute-complementary information," in *Computer Vision and Pattern Recognition Workshops (CVPRW), 2017 IEEE Conference on*, pp. 1435–1443, IEEE.

Shu, G., Dehghan, A., Oreifej, O., Hand, E., and Shah, M. (2012), "Part-based multiple-person tracking with partial occlusion handling," in *Computer Vision and Pattern Recognition (CVPR), 2012 IEEE Conference on*, pp. 1815–1821, IEEE.

Si, J., Zhang, H., Li, C.-G., Kuen, J., Kong, X., Kot, A. C., and Wang, G. (2018), "Dual Attention Matching Network for Context-Aware Feature Sequence based Person Re-Identification," *arXiv preprint arXiv:1803.09937*.

Smith, K., Gatica-Perez, D., Odobez, J.-M., and Ba, S. (2005), "Evaluating multi-object tracking," in *Computer Vision and Pattern Recognition-Workshops, 2005. CVPR Workshops. IEEE Computer Society Conference on*, pp. 36–36, IEEE.

Solera, F., Calderara, S., and Cucchiara, R. (2015), "Towards the evaluation of reproducible robustness in tracking-by-detection," in *Advanced Video and Signal Based Surveillance (AVSS), 2015 12th IEEE International Conference on*, pp. 1–6, IEEE.

Sun, Y., Zheng, L., Deng, W., and Wang, S. (2017), "SVDNet for Pedestrian Retrieval," .

Swamy, C. (2004), "Correlation clustering: maximizing agreements via semidefinite programming," in *ACM-SIAM symposium on Discrete algorithms*.

Tan, J. (2008), "A note on the inapproximability of correlation clustering," .

Tang, S., Andres, B., Andriluka, M., and Schiele, B. (2015), "Subgraph Decomposition for Multi-Target Tracking," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 5033–5041.

Tang, S., Andres, B., Andriluka, M., and Schiele, B. (2016), "Multi-person tracking by multicut and deep matching," in *European Conference on Computer Vision*, pp. 100–111, Springer.

Tang, S., Andriluka, M., Andres, B., and Schiele, B. (2017), "Multiple People Tracking by Lifted Multicut and Person Re-identification," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 3539–3548.

Tesfaye, Y. T., Zemene, E., Prati, A., Pelillo, M., and Shah, M. (2017), "Multi-Target Tracking in Multiple Non-Overlapping Cameras using Constrained Dominant Sets," *arXiv preprint arXiv:1706.06196*.

Varior, R. R., Haloi, M., and Wang, G. (2016), "Gated siamese convolutional neural network architecture for human re-identification," in *European Conference on Computer Vision*, pp. 791–808, Springer.

Vondrick, C., Patterson, D., and Ramanan, D. (2012), "Efficiently Scaling up Crowdsourced Video Annotation," *International Journal of Computer Vision*, pp. 1–21, 10.1007/s11263-012-0564-1.

Wang, G., Yuan, Y., Chen, X., Li, J., and Zhou, X. (2018), "Learning Discriminative Features with Multiple Granularities for Person Re-Identification," *ArXiv e-prints*.

Wen, L., Li, W., Yan, J., Lei, Z., Yi, D., and Li, S. Z. (2014), "Multiple target tracking based on undirected hierarchical relation hypergraph," in *Computer Vision and Pattern Recognition (CVPR), 2014 IEEE Conference on*, pp. 1282–1289, IEEE.

Wen, L., Du, D., Cai, Z., Lei, Z., Chang, M., Qi, H., Lim, J., Yang, M., and Lyu, S. (2015), "UA-DETRAC: A New Benchmark and Protocol for Multi-Object Detection and Tracking," *arXiv CoRR*, abs/1511.04136.

Wen, Y., Zhang, K., Li, Z., and Qiao, Y. (2016), "A discriminative feature learning approach for deep face recognition," in *European Conference on Computer Vision*, pp. 499–515, Springer.

Wu, B. and Nevatia, R. (2006), "Tracking of multiple, partially occluded humans based on static body part detection," in *Computer Vision and Pattern Recognition, 2006 IEEE Computer Society Conference on*, vol. 1, pp. 951–958, IEEE.

Wu, B. and Nevatia, R. (2007), "Detection and tracking of multiple, partially occluded humans by bayesian combination of edgelet based part detectors," *International Journal of Computer Vision*, 75, 247–266.

Wu, C.-W., Zhong, M.-T., Tsao, Y., Yang, S.-W., Chen, Y.-K., and Chien, S.-Y. (2017), "Track-clustering Error Evaluation for Track-based Multi-Camera Tracking System Employing Human Re-identification," in *Computer Vision and Pattern Recognition Workshops (CVPRW), 2017 IEEE Conference on*, pp. 1416–1424, IEEE.

Wu, Y., Lin, Y., Dong, X., Yan, Y., Ouyang, W., and Yang, Y. (2018), "Exploit the Unknown Gradually: One-Shot Video-Based Person Re-Identification by Stepwise Learning," in *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*.

Xiang, Y., Alahi, A., and Savarese, S. (2015), "Learning to track: Online multi-object tracking by decision making," in *2015 IEEE international conference on computer vision (ICCV)*, no. EPFL-CONF-230283, pp. 4705–4713, IEEE.

Xiao, Q., Luo, H., and Zhang, C. (2017), "Margin Sample Mining Loss: A Deep Learning Based Method for Person Re-identification," *arXiv preprint arXiv:1710.00478*.

Xu, J., Zhao, R., Zhu, F., Wang, H., and Ouyang, W. (2018), "Attention-aware Compositional Network for Person Re-Identification," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*.

Yoon, K., Song, Y.-m., and Jeon, M. (2018), "A Multiple Hypothesis Tracking Algorithm for Multi-Target Multi-Camera Tracking with Disjoint Views," .

Yu, F., Li, W., Li, Q., Liu, Y., Shi, X., and Yan, J. (2016a), "Poi: Multiple object tracking with high performance detection and appearance feature," in *European Conference on Computer Vision*, pp. 36–42, Springer.

Yu, S.-I., Meng, D., Zuo, W., and Hauptmann, A. (2016b), "The Solution Path Algorithm for Identity-Aware Multi-Object Tracking," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 3871–3879.

Zamir, A., Dehghan, A., and Shah, M. (2012), "GMCP-Tracker: Global Multi-object Tracking Using Generalized Minimum Clique Graphs," in *Proceedings of the European Conference on Computer Vision (ECCV)*.

Zhang, L., Li, Y., and Nevatia, R. (2008), "Global data association for multi-object tracking using network flows," in *Computer Vision and Pattern Recognition, 2008. CVPR 2008. IEEE Conference on*, pp. 1–8, IEEE.

Zhang, L., Xiang, T., and Gong, S. (2016), "Learning a discriminative null space for person re-identification," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 1239–1248.

Zhang, S., Staudt, E., Faltemier, T., and Roy-Chowdhury, A. (2015a), "A Camera Network Tracking (CamNeT) Dataset and Performance Baseline," in *2015 IEEE Winter Conference on Applications of Computer Vision (WACV)*, pp. 365–372.

Zhang, S., Zhu, Y., and Roy-Chowdhury, A. (2015b), "Tracking multiple interacting targets in a camera network," *Computer Vision and Image Understanding*, 134, 64–73.

Zhang, X., Luo, H., Fan, X., Xiang, W., Sun, Y., Xiao, Q., Jiang, W., Zhang, C., and Sun, J. (2017a), "AlignedReID: Surpassing Human-Level Performance in Person Re-Identification," *arXiv preprint arXiv:1711.08184*.

Zhang, Z., Wu, J., Zhang, X., and Zhang, C. (2017b), "Multi-Target, Multi-Camera Tracking by Hierarchical Clustering: Recent Progress on DukeMTMC Project," *arXiv preprint arXiv:1712.09531*.

Zhao, R., Ouyang, W., and Wang, X. (2013), "Unsupervised Salience Learning for Person Re-identification," in *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*.

Zheng, L., Shen, L., Tian, L., Wang, S., Wang, J., and Tian, Q. (2015), "Scalable Person Re-identification: A Benchmark," in *Computer Vision, IEEE International Conference on*.

Zheng, L., Bie, Z., Sun, Y., Wang, J., Su, C., Wang, S., and Tian, Q. (2016a), "MARS: A Video Benchmark for Large-Scale Person Re-identification," in *European Conference on Computer Vision*, Springer.

Zheng, L., Yang, Y., and Hauptmann, A. G. (2016b), "Person re-identification: Past, present and future," *arXiv preprint arXiv:1610.02984*.

Zheng, Z., Zheng, L., and Yang, Y. (2017a), "Pedestrian Alignment Network for Large-scale Person Re-identification," *arXiv preprint arXiv:1707.00408*.

Zheng, Z., Zheng, L., and Yang, Y. (2017b), "Unlabeled Samples Generated by GAN Improve the Person Re-identification Baseline in vitro," in *Proceedings of the IEEE International Conference on Computer Vision*.

Zhong, Z., Zheng, L., Kang, G., Li, S., and Yang, Y. (2017), "Random Erasing Data Augmentation," *arXiv preprint arXiv:1708.04896*.

Zhou, Q., Fan, H., Su, H., Yang, H., Zheng, S., and Ling, H. (2018), "Weighted Bilinear Coding over Salient Body Parts for Person Re-identification," *arXiv preprint arXiv:1803.08580*.

Zhou, S., Wang, J., Wang, J., Gong, Y., and Zheng, N. (2017), "Point to set similarity based deep feature learning for person re-identification," in *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*.

Biography

Ergys Ristani was born on 7 January 1990 in Gjirokastër, Albania. After completing high school in his hometown in 2008 he began his undergraduate degree at the American University in Bulgaria. He graduated with a B.A. in Computer Science and Mathematics in May 2012. He matriculated at Duke University in August 2012, obtained a M.S. in Computer Science in December 2016, and will defend his Ph.D. in June 2018.

During his graduate degree Ristani was actively involved in the tracking and re-identification community. He helped organize the CVPR 2017 Workshop on Target Re-Identification and Multi-Target Multi-Camera Tracking, was invited to give a talk at the CVPR 2017 Workshop on Benchmarking Multi-Target Tracking, helped maintain the MOTChallenge server, and actively reviewed for ICCV, CVPR and ECCV conferences.

Ristani will join Facebook Reality Labs in September 2018.