A dissertation submitted in accordance with the requirements for the degree of

Doctor of Philosophy

Generic Multiple Object Tracking

WENHAN LUO

Imperial College of Science, Technology and Medicine

Department of Electrical & Electronic Engineering

May 2016

Declaration

This dissertation is the result of my own work and includes nothing which is the outcome of work done in collaboration except where specifically indicated in the text. The research detailed in this dissertation was conducted under the guidance of Dr. Tae-Kyun Kim.

The research presented in this dissertation resulted in several publications and submissions with joint authorship as follows:

W. Luo and T-K. Kim, Generic Object Crowd Tracking by Multi-Task Learning, Proc. of British Machine Vision Conference (BMVC), Bristol, UK, pp. 73.1-73.13, 2013.

W. Luo, T-K. Kim, B. Stenger, X. Zhao and R. Cipolla, Bi-label Propagation for Generic Multiple Object Tracking, Proc. of IEEE Conf. on Computer Vision and Pattern Recognition (CVPR), Columbus, Ohio, USA, pp. 1290-1297, 2014.

W. Luo, B. Stenger, X. Zhao and T-K. Kim, Automatic Topic Discovery for Multi-object Tracking, Proc. of the Association for the Advancement of Artificial Intelligence (AAAI), Austin, Texas, USA, pp. 3820-3826, 2015.

W. Luo, J. Xing, X. Zhang, X. Zhao and T-K. Kim, Multiple Object Tracking: A Literature Review, submitted to International Journal of Computer Vision (IJCV).

W. Luo, B. Stenger, X. Zhao and T-K. Kim, Multi-object Tracking by Automatic Topic Discovery, submitted to IEEE Transactions on Pattern Analysis and Machine Intelligence (TPAMI).

Wenhan Luo, May 2016

Abstract

Multiple object tracking is an important problem in the computer vision community due to its applications, including but not limited to, visual surveillance, crowd behavior analysis and robotics. The difficulties of this problem lie in several challenges such as frequent occlusion, interaction, high-degree articulation, etc. In recent years, data association based approaches have been successful in tracking multiple pedestrians on top of specific kinds of object detectors. Thus these approaches are type-specific. This may constrain their application in scenario where type-specific object detectors are unavailable. In view of this, I investigate in this thesis tracking multiple objects without ready-to-use and type-specific object detectors. More specifically, the problem of multiple object tracking is generalized to tracking targets of a generic type. Namely, objects to be tracked are no longer constrained to be a specific kind of objects. This problem is termed as *Generic Multiple Object Tracking (GMOT)*, which is handled by three approaches presented in this thesis.

In the first approach, a generic object detector is learned based on manual annotation of only one initial bounding box. Then the detector is employed to regularize the online learning procedure of multiple trackers which are specialized to each object. More specifically, multiple trackers are learned simultaneously with shared features and are guided to keep close to the detector. Experimental results have shown considerable improvement on this problem compared with the state-of-the-art methods. The second approach treats detection and tracking of multiple generic objects as a bi-label propagation procedure, which is consisted of class label propagation (detection) and object label propagation (tracking). In particular, the cluster Multiple Task Learning (cMTL) is employed along with the spatio-temporal consistency to address the online detection problem. The tracking problem is addressed by associating existing trajectories with new detection responses considering appearance, motion and context information. The advantages of this approach is verified by extensive experiments on several public data

sets. The aforementioned two approaches handle GMOT in an online manner. In contrast, a batch method is proposed in the third work. It dynamically clusters given detection hypotheses into groups corresponding to individual objects. Inspired by the success of topic model in tackling textual tasks, Dirichlet Process Mixture Model (DPMM) is utilized to address the tracking problem by cooperating with the so-called must-links and cannot-links, which are proposed to avoid physical collision. Moreover, two kinds of representations, superpixel and Deformable Part Model (DPM), are introduced to track both rigid and non-rigid objects. Effectiveness of the proposed method is demonstrated with experiments on public data sets.

To my family for their continual love, support and encouragement.

Acknowledgments

I thank my supervisor, Dr. Tae-Kyun Kim, for his guidance, support and encouragement. I am also highly grateful to Dr. Björn Stenger, for his constructive suggestion which helps shape me as a researcher.

I thank both the China Scholarship Council and Imperial College London for providing me with financial security through the CSC-Imperial Scholarship, without which this work would not have been possible.

I thank all the members, past and present, of the Imperial College Vision Lab, for their companionship. Discussions with them also inspired some of the findings in this thesis. I also thank the administrative staff of Imperial College London for their help during my study period.

Finally, I thank my family for their support and encouragement throughout my life.

CONTENTS

List of Figures

List of Tables

List of Algorithms

Acronyms

AR Augmented Reality. 6

cMTL clustered Multiple Task Learning. v, 9, 57, 60, 62, 67, 68, 73, 104, 107

CRP Chinese Restaurant Process. 81, 89

DBT Detection Based Tracking. 17, 19, 23

DFT Detection Free Tracking. 17–19, 23, 35

DPM Deformable Part Model. vi, 9, 10, 78, 85, 100

DPMM Dirichlet Process Mixture Model. vi, 9–11, 77, 78, 80–82, 85, 89, 90, 92, 95, 101, 104

FM FragMentation. 92–95, 98, 100, 105

GMOT Generic Multiple Object Tracking. v, vi, xiii, xxi, 1, 3–11, 13, 36, 39, 46, 56, 57, 60, 83, 93, 95, 103, 105, 106

HCI Human Computer Interface. 6

HOG Histogram of Oriented Gradients. 51, 66, 85

IDS ID Switches. 92–95, 98, 100

LBP Local Binary Patterns. 51, 66

LDA Latent Dirichlet Allocation. 79, 80

LSA Latent Semantic Analysis. 78, 79

LSI Latent Semantic Indexing. 78

ML Mostly Lost trajectories. 52, 70–72, 92, 93, 95, 98–100

MOT Multiple Object Tracking. xxi, 2, 3, 5–8, 13, 14, 17, 19, 22–25, 35, 37, 38, 46, 51, 62, 82

MOTA Multiple Object Tracking Accuracy. 52, 70, 72

MOTP Multiple Object Tracking Precision. 52, 70–72

MT Mostly Tracked trajectories. 52, 70, 72, 92, 95, 99, 100

MTL Multiple Task Learning. 9, 38–42, 46, 47, 57, 104, 105

MTT Multiple Target Tracking. 2

pLSA probabilistic Latent Semantic Analysis. 79

SVM Support Vector Machine. 9, 10, 38, 39, 45, 57, 67, 100, 105

VR Virtual Reality. 6

VS Visual Surveillance. 6

CHAPTER 1

INTRODUCTION

Because of the popularity of mobile devices, multimedia data, such as pictures, videos and so on, are much more than before. As a result, automatic processing and understanding of these kinds of multimedia are highly demanded. Computer Vision, which is a field including methods for acquiring, processing, analyzing, and understanding images and, in general, high-dimensional data from the real world. It draws more and more attention for both its academic importance and commercial applications. In computer vision, tracking plays an important role. As a mid-level task, tracking is the basis of tasks like action recognition, event detection in academic research. Meanwhile, tracking also plays a core role in visual surveillance and virtual reality for commercial applications. The focus of this thesis is set on the development of algorithms to address the problem of Generic Multiple Object Tracking (GMOT). This problem is the generalization of the traditional multiple object tracking problem. It is closely related to problems of both visual tracking and multiple object tracking. Thus, before formally representing the problem, these two related problems are introduced as follows.

Visual Tracking

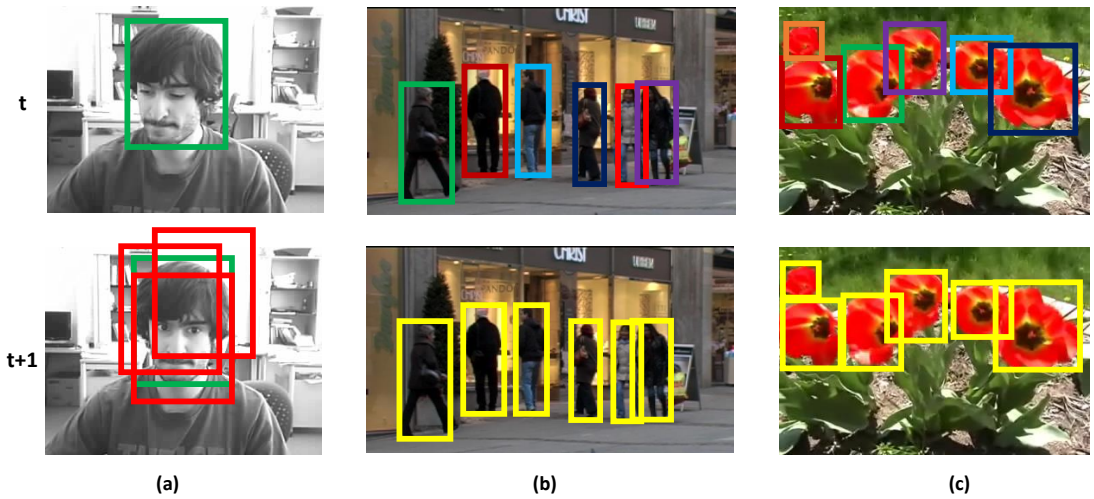

Figure 1.1: *Comparison among visual tracking, multiple object tracking and the concerned generic multiple object tracking from one frame to the next frame in this thesis. Note that the objects in the problem of multiple object tracking are usually of some specific categories, while generic multiple object tracking does not make such an assumption. For example, objects in (c) could be of any other types.*

Visual tracking is an important problem in the community due to its numerous applications in domains such as video surveillance and human computer interface. It aims at estimating the state (size, position, etc.) of an object in a given image sequence. In the first frame, annotation of an interested object is given manually or by detection. To track the interested object, an observation model and a dynamic model are required. The dynamic model is to obtain candidates in the next frame based on the state of current frame. The observation model measures the similarity between observations extracted from the candidates and the up-to-time state of object. As shown in Figure 1.1(a), the green bounding box is the object state in the current frame and the red bounding boxes are candidates according to the dynamic model. Based on the observation model, the candidate sharing the largest similarity with the green bounding box (in the current frame) would be the estimation in the next frame.

Multiple Object Tracking

Multiple Object Tracking (MOT), or Multiple Target Tracking (MTT), plays an important role in computer vision as a result of its importance for other tasks like scene understanding and pose estimation. MOT addresses the problem of locating multiple objects, maintaining their

identities and yielding their individual trajectories given an input video containing a specific type of objects. The specific type of objects here, referring to objects which draw considerable attention in practical applications. Common targets of MOT include pedestrians on the street, sport players in the court, vehicles, human faces and so on.

Due to the recent progress in object detection, especially in pedestrian detection [Felzenszwalb et al., 2010, Dalal and Triggs, 2005], the MOT problem has also achieved great success. Traditional approaches employ the well-developed detectors of specific kinds of objects to obtain detection hypotheses. Based on the detection hypotheses, MOT becomes a data association problem, i.e., detection hypotheses which are considered to belong to one specific object are associated into a tracklet. As shown in Figure 1.1(b), six detection hypotheses (different colors encode different identities) are obtained in the current frame. There are six detection hypotheses in the next frame while they are without identity information. Multiple object tracking is conducted by associating detection hypotheses which are without identity information to detection hypotheses which are of identity information. This is the so-called "data association" approach to this problem.

Generic Multiple Object Tracking

As mentioned above, traditional multiple object tracking focuses on specific kinds of objects. The primary reason is that one can easily adopt a ready-to-use object detector trained for a specific kind of objects offline. However, there are several drawbacks of such standard approaches to MOT:

- object detectors of high accuracy usually require a large amount of labeled data, which costs a lot of labor to collect.
- assuming there is an off-the-shelf object detector to use, the performance on a specific sequence is not guaranteed to be optimal. Because an object detector is trained based on a specific set of data in most cases. It is only guaranteed to be optimal in this set of data while does not generalize well when it is applied in other scenario.

- even if an object detector can be guaranteed to achieve the optimal performance on different sets of data, it could be applied only to sequences of the same type of objects. For different types of objects, various kinds of object detectors are required. This limits application of this solution as it is not practical to access an accurate object detector trained offline already for every specific kind of objects.

In view of this, I propose to generalize the problem of traditional multiple object tracking from specific types of objects to generic types of objects. Accordingly, the problem is called *Generic Multiple Object Tracking (GMOT)* in this thesis. In particular, the type of objects to be tracked is not constrained to be pedestrian or some other specific types of objects, and the type of objects is not known in advance in the concerned problem. As shown in Figure 1.1(c), the task is the same as traditional multiple object tracking, while the objects could be of any type. Therefore, it is not always possible to apply a ready-to-use detector in the problem of GMOT. To address this issue, an object detector should be trained online. To train such an object detector, one instance of the concerned type of objects is required as a starting point. Given an image sequence, based on a given initial bounding box of one instance of multiple objects in the first frame, an object detector is obtained by collecting positive and negative samples and then training according to the collected data. Applying this object detector, objects of the same type as the labeled object are detected such that more and more training samples could be collected. Relying on more training data, the object detector could be refined progressively. Subsequently, objects are tracked to maintain their identities. Finally, a set of trajectories corresponding to individual objects are produced as the estimation. Figure 1.2 shows the general layout of solutions to this problem.

More detailed comparison among these three problems are listed in Table 1.1. Given a video, the problem of visual tracking aims to seize a concerned object in the image sequence based on the initial annotation in the first frame. The output is the trajectory of the interested object. For multiple object tracking problem concerning a specific kind of objects, a detector for this type of objects is demanded to make the algorithm be aware of the occurrence and disappearance of objects. Accordingly, the output is a set of trajectories corresponding to

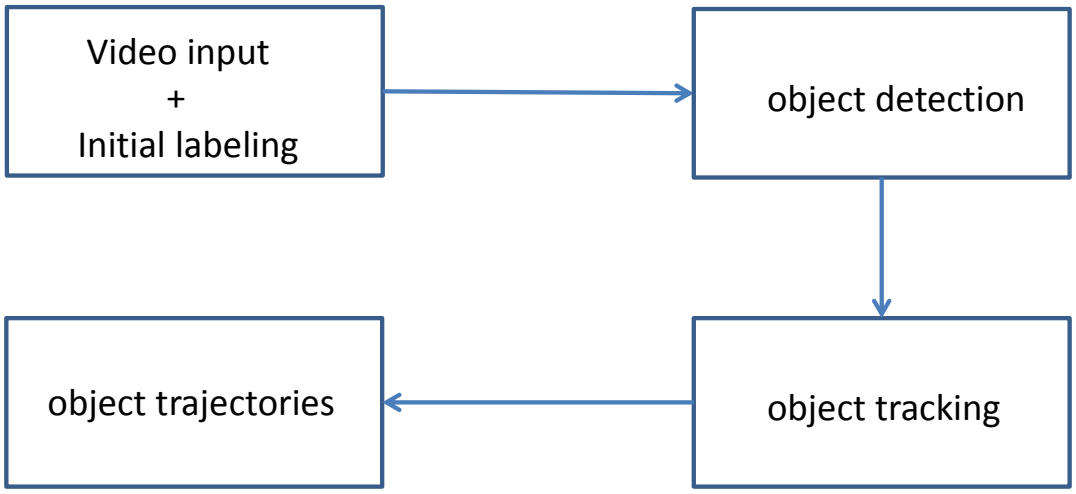

Figure 1.2: *An overview of a typical solution to the problem of generic multiple object tracking. The main aim of my research is the development of effective algorithms for detection and tracking of objects of a generic type.*

Table 1.1: *Comparison among the three problems of visual tracking, MOT and GMOT in terms of input and output.*

Problem	Input	Output
Visual tracking	video and an initial bounding box	object trajectory
MOT	video and a (*type-specific*) object detector	object trajectories
GMOT	video and an initial bounding box	object trajectories

multiple objects. In terms of GMOT, the task is identical to that of MOT, while the concerned objects do not need to be some specific kinds of objects. Thus it is not necessary to rely on type-specific object detectors trained offline. In contrast, it requires an initial bounding box in the first frame as a starting point to train a detector for objects of the same type as the annotated object in an online manner. The output is also a set of trajectories corresponding to different objects.

1.1 Academic Importance and Commercial Applications

In this section, the academic importance and commercial application of MOT are firstly introduced. Then, applications particularly of GMOT are given.

1.1.1 Academic Importance

As a mid-level task, multiple object tracking grounds many tasks in the computer vision community. For instance, it is the foundation of tasks such as segmentation [Yilmaz et al., 2004], pose estimation [Zhang et al., 2014, Zhang et al., 2007, Gammeter et al., 2008], population counting [Liu et al., 2005], object recognition [Lowe, 1999], action recognition [Choi and Savarese, 2012], behavior analysis [Moeslund et al., 2006], abnormal event detection [Cong et al., 2011] and scene understanding [Zhou et al., 2011a]. A robust and effective approach to multiple object tracking could provide great convenience to these kinds of tasks.

1.1.2 Commercial Applications

Apart from the academic importance, MOT has significant potential in many commercial applications. Some of the typical applications are listed as follows.

- Visual Surveillance (VS). The massive amount of videos (especially surveillance videos) requires automatic analysis to detect abnormal behaviors, which is based on analysis of objects' actions, trajectories, etc. To obtain such information, one needs to locate targets and track them, which is exactly the objective of multiple object tracking.
- Human Computer Interface (HCI). Visual information, such as expression and gesture, can be employed to achieve advanced function in HCI. Extraction of visual information

requires visual tracking as the basis. When multiple objects appear in the scene, interactions among them need to be considered. In this case, MOT plays a crucial rule to make HCI more natural and intelligent.

- Virtual Reality (VR) and Augmented Reality (AR). MOT has also been applicable in these areas. For instance, AR systems need to know the accurate position, pose and geometric relations of objects so as to locate virtual objects in the real world. In VR, tracking is indispensable for natural interaction between human and virtual environment.

As mentioned before, the difference between MOT and GMOT is that, MOT is interested in some special kinds of objects while GMOT concerns more general objects. Thus, GMOT is more practical in real world. It can be applied in the scenario where there is not a ready-to-use object detector available. Compared with MOT, GMOT has some additional applications, including but not limited to: 1) medical image processing. Some tasks of medical image processing require laborious manual labeling. For instance, one task could be labeling or tracking multiple cells in images. As it is usually not easy to have a reliable and customized detectors of cells like pedestrian, this is a representative application of GMOT. It can help to save a large amount of labeling cost, 2) automatic video editing/tagging. In some commercial videos, there are multiple similar objects. Users may be interested in searching other similar objects given a query object, 3) wildlife conservation. There is a typical application in the field of wildlife conservation. Sometimes people need to know the status of some kinds of wild animals in a specific area. However, traditional sensors like GPS device are not allowed or too expensive to set. In this case, a footage recorded from a helicopter is analyzed by detecting, tracking and counting them to investigate their behavior.

1.2 Challenges

The academic importance and enormous potential of applications above have sparkled enormous interest in this topic. However, issues shared by both MOT and GMOT render the tracking problem challenging:

Frequent occlusions

Occlusion appears frequently in the scenario of multiple object tracking. To put it simply, occlusion refers to that observation of an object is blocked partly or completely. Generally, there are two cases of occlusion. The first case is that one object is occluded by other object(s). The second case is that one object is occluded by stuff in the scene. Occlusion leads to false positive hypotheses in the detection stage and also confuses the tracker due to the loss of observation.

Tracking management

Tracking management is known as the strategy of determining initialization and termination of tracks. Such strategy is even more important for online multiple object tracking. As the detector of objects are not perfect, false positive and false negative detection hypotheses are usually inevitable in reality. In most cases, object detector would produce false positive and false negative detection hypotheses. In case of false positive hypotheses, it is not appropriate to initialize a track at once as this may generate trajectories of background objects. Similarly, terminating a track in case of false negative hypotheses is not suitable since it may result in fragmentation of trajectories.

Small size of objects

In some cases of multiple object tracking, size of the interested objects is considerably small [Betke et al., 2007]. For example, it can be 20 pixels by 20 pixels in aerial videos [Reilly et al., 2010]. In such a case, the commonly used appearance information is not reliable.

Group behavior of multiple objects

Objects are usually not separated from each other. On the contrary, there exists interaction among objects, such as moving together, crossing each other and so on. Appropriate modeling of group behavior could alleviate the problem to some extent. However, the group behavior is complex to define, and there is not any general guidance of modeling it.

Compared with standard MOT, GMOT cannot rely on the detectors trained offline, thus it is even more challenging. The difficulties mainly raise from the following perspectives.

Online object detection

As mentioned before, object detection in GMOT starts from one single bounding box in the first frame. This is challenging as an accurate object detector in general requires a large amount of data with label, which is not available in this case. Thus, how to conduct object detection online is a primary issue.

Similar appearance among objects

This issue is more challenging in case of generic multiple object tracking as objects are more similar to each other compared with objects in case of traditional multiple object tracking. Therefore, besides from distinguishing objects from background, object needs to be additionally discriminated from other objects of the same category. Namely, the problem poses more requirements in the tracker.

1.3 Approaches and Contributions

In this thesis, the following approaches are proposed:

1. A framework of two primary components to tackle the problem of GMOT is proposed. One is online detection and the other one is online tracking. A linear Laplacian SVM classifier considering smoothness is developed for online detection with adaptivity. Inspired by Multi-Task Learning (MTL), multiple objects are tracked simultaneously with sharable features among different trackers. Furthermore, the detection and tracking components are connected using the detector as the mean to regularize the multiple trackers.

2. GMOT is addressed by propagating class and object labels jointly in spatial and temporal domains. Moreover, the clustered Multi-Task Learning (cMTL) is introduced for generic object detection and improved by considering the spatio-temporal consistency.

3. A topic model based on Dirichlet Processing Mixture Model (DPMM) is proposed to discover objects automatically and effectively as a batch solution. To be more specific,

detection hypotheses are dynamically clustered into groups based on the assumption that detection instances belonging to an identical object share the similar co-occurrence of visual words with each other. More importantly, this dynamic clustering algorithm could serve as a basic framework to integrate other appearance or motion models for multi-object tracking.

4. Dirichlet Processing Mixture Model (DPMM) is combined with Deformable Part Model (DPM) [Felzenszwalb et al., 2010] to deal with the problem of tracking multiple pedestrians. The deformable part model describes the latent semantic characters of non-rigid pedestrians. By applying DPMM along with DPM, instances of an identical object are clustered together as a trajectory.

I present the following contributions to the computer vision community:

1. It is the first time to introduce the generic multiple object tracking to the community. Objects of general types rather than of a specific kind are considered in the thesis.

2. Detection and tracking are handled in a unified framework by multi-task learning. It is the first time that detector is elegantly used to regularize the learning of trackers.

3. The concept of label propagation is utilized to tackle both tracking and detection of objects. To adapt it to the concerned problem, bi-label propagation is proposed.

4. Rather than data association, dynamic clustering is employed to track multiple objects offline. A novel topic model is developed to cluster instances sharing similar patterns into clusters, corresponding to individual objects.

5. In experiments, the proposed methods achieve the state-of-the-art performance, verifying their ability of solving the GMOT problem.

1.4 Outlines

This thesis consists of 6 chapters in total. The content of the remaining chapters are listed and summarized as follows.

1.4.1 Chapter 2. Background

Chapter 2 presents the related research in literature. To begin with, I briefly introduce preliminaries of generic multiple object tracking. Then a general formulation of generic multiple object tracking is presented. This is followed by the introduction of categorization. In the end, some of the most relevant works are discussed.

1.4.2 Chapter 3. Generic MOT by Multi-task Learning

A novel approach named as GMOT-MTL [Luo and Kim, 2013] is presented to address GMOT in this chapter. In this work, the problem of GMOT is decomposed into two major tasks, i.e., detection and tracking. For the detection task, Laplacian SVM is introduced to learn an object detector. For the tracking task, tracking each object is treated as a sub task within the major tracking task. Inspired by the concept of multi-task learning, i.e., learning of multiple related tasks is better than learning them independently, trackers of multiple objects are learned by sharing features among them. At the same time, the detector learned in the detection phase is utilized to regularize the learning of multiple trackers in the tracking stage. By employing the detector to regularize the learning of multiple trackers, the trackers are prevented from drifting to the background. The proposed method is evaluated on several public data sets, and proven to be effective in solving GMOT.

Related Publication

W. Luo, T-K. Kim, Generic Object Crowd Tracking by Multi-Task Learning, Proc. of British Machine Vision Conference (BMVC), Bristol, UK, pages 73.1-73.13, 2013.

1.4.3 Chapter 4. Bi-label Propagation for Generic MOT

This chapter describes an approach which casts the GMOT problem as a bi-label propagation problem [Luo et al., 2014a]. The concept of bi-labels refers to the combination of binary class label and individual object labels. To propagate the binary class label, the clustered multiple task learning is employed to train an object detector, distinguishing objects from background. To further improve the performance, the so-called spatio-temporal consistency is incorporated into the energy function. Tracking is treated as object label propagation. When computing the affinity between detections and trajectories, the appearance, motion and context are considered jointly, which are proved to be robust. I show in the experimental section that the bi-label propagation for GMOT (referred as GMOT-BLP) outperforms several approaches including the previously proposed GMOT-MTL method.

Related Publication

W. Luo, T-K. Kim, B. Stenger, X. Zhao, R. Cipolla, Bi-label Propagation for Generic Multiple Object Tracking, Proc. of IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Columbus, Ohio, USA, pages 1290-1297, 2014.

1.4.4 Chapter 5. Automatic Topic Discovery for Generic MOT

In Chapter 5, I propose an offline solution to GMOT relying on provided detection hypotheses. The tracking phase is taken as the major concern in this chapter. This approach [Luo et al., 2015] therefore focuses on tracking. Particularly, this method associates detection responses into trajectories by dynamic clustering. Inspired by the success of topic model in textual process tasks, the Dirichlet Process Mixture Model (DPMM) is employed to discover the appearance coherence between tracklets, i.e., co-occurrence of visual words in the tracklets. By doing this, the detections of a specific object are clustered as a unique topic. Namely, the proposed approach tracks objects as a procedure of automatic topic discovery. This batch method (called GMOT-ATD) improves the performance compared with the previous sequential ones [Luo and

Kim, 2013, Luo et al., 2014a]. Additionally, the proposed method is applied to the problem of pedestrian tracking, and shows promising results, which further proves its potential value in real-world applications.

Related Publication

W. Luo, B. Stenger, X. Zhao, T-K. Kim, Automatic Topic Discovery for Multi-object Tracking, Proc. of the Association for the Advancement of Artificial Intelligence (AAAI), Austin, Texas, USA, pages 3820-3826, 2015.

1.4.5 Chapter 6. Conclusion and Future Work

Chapter 6 draws a conclusion of this thesis. Contents of the previous chapters are summarized and feasible directions which could further improve the performance are provided.

CHAPTER 2

BACKGROUND

Before diving into the technical chapters, I will firstly provide some preliminary knowledge relevant to the GMOT problem. Then a general formulation of multiple object tracking is given, followed by detailed discussion on categorization of MOT. Some key components in addressing MOT, such as appearance, motion, interaction, exclusion and occlusion are discussed in detail. In the end, some of the most relevant works are discussed.

2.1 Preliminaries

Object is considered as a continuous closed area in an image which is distinct from its surroundings.

Detection is a computer vision task which localizes objects in images. A detector is usually trained from a large amount of labeled samples to distinguish the foreground objects from background. In most situations, detection does not involve temporal information.

Tracking is to localize an identical object in continuous frames. Thus tracking is usually applied in a video or an image sequence with temporal information. In MOT, tracking means simultaneous localization and identification of multiple objects of interest.

Detection responses are also known as detection observations or detection hypotheses, which are the outputs of an object detector trained for a specific kind of objects, such as human, vehicle, face and animal. They are configurations of objects such as positions, sizes, etc.

Trajectory is the output of a MOT system. One trajectory corresponds to only one target, thus a trajectory is unique. In particular, one trajectory is composed of multiple object responses of an identical target in an image sequence. Each response represents the location, size and some other information in one frame.

Tracklet is an intermediate representation of output between detection responses and trajectories. It is composed of several detection responses which are from an identical target. As a fact, a detection response can be viewed as a tracklet composed of only one detection response. Tracklet is usually obtained by linking *confident* detection responses, thus it is shorter than trajectory in terms of time span. In some approaches [Huang et al., 2008], the final trajectories are obtained by progressively linking detection responses into longer and longer tracklets and eventually forming trajectories. Figure 2.1 shows the concepts of detection responses, tracklets and trajectories.

Data association is a typical solution to multiple object tracking when the problem is cast as a paradigm of matching detection responses across frames based on object detection. The technique of data association figures out correspondences between detection hypotheses.

2.2 Problem Formulation

Generic multiple object tracking can be generally formulated as a multi-variable estimation problem. Given an image sequence $\{\mathbf{I}_1, \mathbf{I}_2, ..., \mathbf{I}_t, ...\}$ as input, $\mathbf{s}_t^i$ is employed to denote the state of the i-th object in the t-th frame. This chapter uses $\mathbf{S}_t = (\mathbf{s}_t^1, \mathbf{s}_t^2, ..., \mathbf{s}_t^{M_t})$ to denote

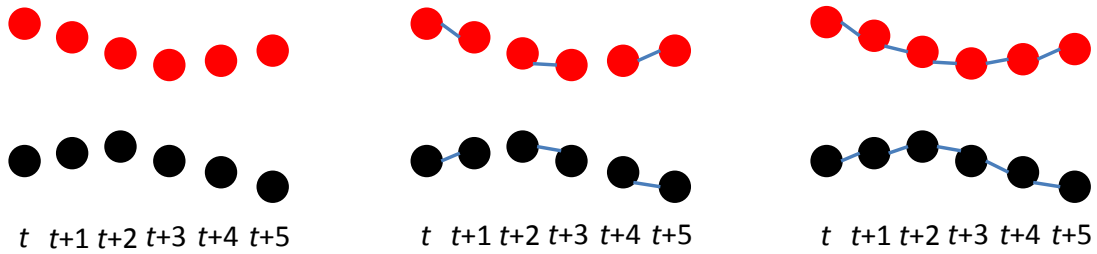

Figure 2.1: *Detection responses (left), tracklets (center), and trajectories (right) are shown in continuous 6 frames. Different colors encode different targets. Best viewed in color.*

states of all the M_t objects in the t-th frame, $\mathbf{s}_{1:t}^i = \{\mathbf{s}_1^i, \mathbf{s}_2^i, ..., \mathbf{s}_t^i\}$ to denote the sequential states of the i-th object from the first frame to the t-th frame, and $\mathbf{S}_{1:t} = \{\mathbf{S}_1, \mathbf{S}_2, ..., \mathbf{S}_t\}$ to denote all the sequential states of all the objects from the first frame to the t-th frame. Note that the number of objects may vary from frame to frame.

To estimate the states of objects, some observations should be firstly collected from the image sequence. Correspondingly, $\mathbf{o}_t^i$ is utilized to denote the collected observations for the i-th object in the t-th frame. $\mathbf{O}_t = (\mathbf{o}_t^1, \mathbf{o}_t^2, ..., \mathbf{o}_t^{M_t})$ denotes the collected observations for all the M_t objects in the t-th frame. $\mathbf{o}_{1:t}^i = \{\mathbf{o}_1^i, \mathbf{o}_2^i, ..., \mathbf{o}_t^i\}$ denotes the sequential observations collected from the first frame to the t-th frame and $\mathbf{O}_{1:t} = \{\mathbf{O}_1, \mathbf{O}_2, ..., \mathbf{O}_t\}$ denotes all the collected sequential observations of all the objects from the first frame to the t-th frame.

The objective of multiple object tracking is to find the "optimal" sequential states of all the objects, which can be generally modeled by performing MAP (maximal a posterior) estimation from the conditional distribution of the sequential states of all the objects given all the observations:

$$\widehat{\mathbf{S}}_{1:t} = \arg\max_{\mathbf{S}_{1:t}} P\left(\mathbf{S}_{1:t} | \mathbf{O}_{1:t}\right). \tag{2.1}$$

The estimation can be performed using *probabilistic inference* algorithms based on a two-step iterative procedure [Liu et al., 2012, Breitenstein et al., 2009, Yang et al., 2009, Mitzel and Leibe, 2011, Rodriguez et al., 2011, Kratz and Nishino, 2010, Reid, 1979]:

Predict: $P(\mathbf{S}_t | \mathbf{O}_{1:t-1}) = \int P(\mathbf{S}_t | \mathbf{S}_{t-1}) P(\mathbf{S}_{t-1} | \mathbf{O}_{1:t-1}) d\mathbf{S}_{t-1}$

Update: $P(\mathbf{S}_t|\mathbf{O}_{1:t}) \propto P(\mathbf{O}_t|\mathbf{S}_t)P(\mathbf{S}_t|\mathbf{O}_{1:t-1})$

In the formula above, $P(\mathbf{S}_t|\mathbf{S}_{t-1})$ and $P(\mathbf{O}_t|\mathbf{S}_t)$ are the *Dynamic Model* and the *Observation Model*, respectively. These two models play a very important role in a tracking algorithm. Since the distributions of these two models are usually unknown, sampling methods like Particle Filter [Jin and Mokhtarian, 2007, Yang et al., 2005, Hess and Fern, 2009, Han et al., 2007, Hu et al., 2012, Liu et al., 2012, Breitenstein et al., 2009, Yang et al., 2009], MCMC [Khan et al., 2004, Khan et al., 2005, Khan et al., 2006], RJMCMC [Choi et al., 2013], etc., are employed to perform the estimation.

The estimation problem can also be coped with *deterministic optimization* approaches, e.g., directly maximizing the likelihood function $P(\mathbf{O}_{1:t}|\mathbf{S}_{1:t})$ as a delegate of $P(\mathbf{S}_{1:t}|\mathbf{O}_{1:t})$:

$$\widehat{\mathbf{S}}_{1:t} = \arg\max_{\mathbf{S}_{1:t}} P(\mathbf{S}_{1:t}|\mathbf{O}_{1:t}) = \arg\max_{\mathbf{S}_{1:t}} P(\mathbf{O}_{1:t}|\mathbf{S}_{1:t}), \tag{2.2}$$

or conversely minimizing an energy function

$$\begin{aligned} \widehat{\mathbf{S}}_{1:t} &= \arg\max_{\mathbf{S}_{1:t}} P(\mathbf{S}_{1:t}|\mathbf{O}_{1:t}) \\ &= \arg\max_{\mathbf{S}_{1:t}} \frac{1}{Z}\exp(-C(\mathbf{S}_{1:t}|\mathbf{O}_{1:t})) \\ &= \arg\min_{\mathbf{S}_{1:t}} C(\mathbf{S}_{1:t}|\mathbf{O}_{1:t}), \end{aligned} \tag{2.3}$$

where Z is a normalization factor to make $P(\mathbf{S}_{1:t}|\mathbf{O}_{1:t})$ a probability distribution and $C(\bullet)$ is a cost function.

The specific optimization approaches include bipartite graph matching [Shu et al., 2012, Breitenstein et al., 2009, Wu and Nevatia, 2007b, Qin and Shelton, 2012, Reilly et al., 2010, Perera et al., 2006, Xing et al., 2009, Huang et al., 2008], dynamic programming [Wolf et al., 1989, Jiang et al., 2007, Berclaz et al., 2009, Andriyenko and Schindler, 2010], min-cost max-flow network flow [Zhang et al., 2008, Choi and Savarese, 2012, Wu et al., 2012, Butt and Collins, 2013a, Pirsiavash et al., 2011] and max weight independent set [Shafique et al., 2008, Brendel et al., 2011].

2.3 Categorization

In general, it is difficult to get a universal classification of MOT. In this chapter, MOT is categorized by different criteria to obtain a more comprehensive understanding of the problem. Existing works of visual tracking [Yilmaz et al., 2006, Cannons, 1991] have provided some views for categorization. For example, object shape representation is adopted in [Yilmaz et al., 2006] to group existing work of visual tracking into subsets. To be more specific, different types of object shape representations, such as point, primitive geometric shapes, object contours and region shapes, are described individually. For the concerned MOT problem, as object is represented by region shapes (bounding box or ellipse [Kuo and Nevatia, 2011]) in most works, it is not necessary to discuss it by object representations. Alternatively, the categorization of MOT is discussed from the following perspectives.

2.3.1 Initialization Method

The first criterion is that how objects are initialized. According to this criterion, most of existing MOT work could be grouped into two sets [Yang and Nevatia, 2012c]: Detection Based Tracking (DBT) and Detection Free Tracking (DFT). DBT relies on object detection while DFT does not.

DBT. In DBT, objects are at first localized in each frame and then object hypotheses are linked into trajectories. Figure 2.2(a) shows the flow of DBT. Given a sequence, type-specific object detection or motion detection (based on background modeling) [Bose et al., 2007, Song et al., 2010] is applied to each frame to obtain object hypotheses, then (sequential or batch) tracking is conducted to link detection hypotheses into trajectories. There are three issues worthy noting. First, in most cases object detection procedure is not the focus of DBT methods. The majority of DBT approaches builds upon a pre-trained object detector which produces object hypotheses as observations. Second, as mentioned above, since object detector is trained in advance, the majority of DBT focuses on specific kinds of targets, such as pedestrians,

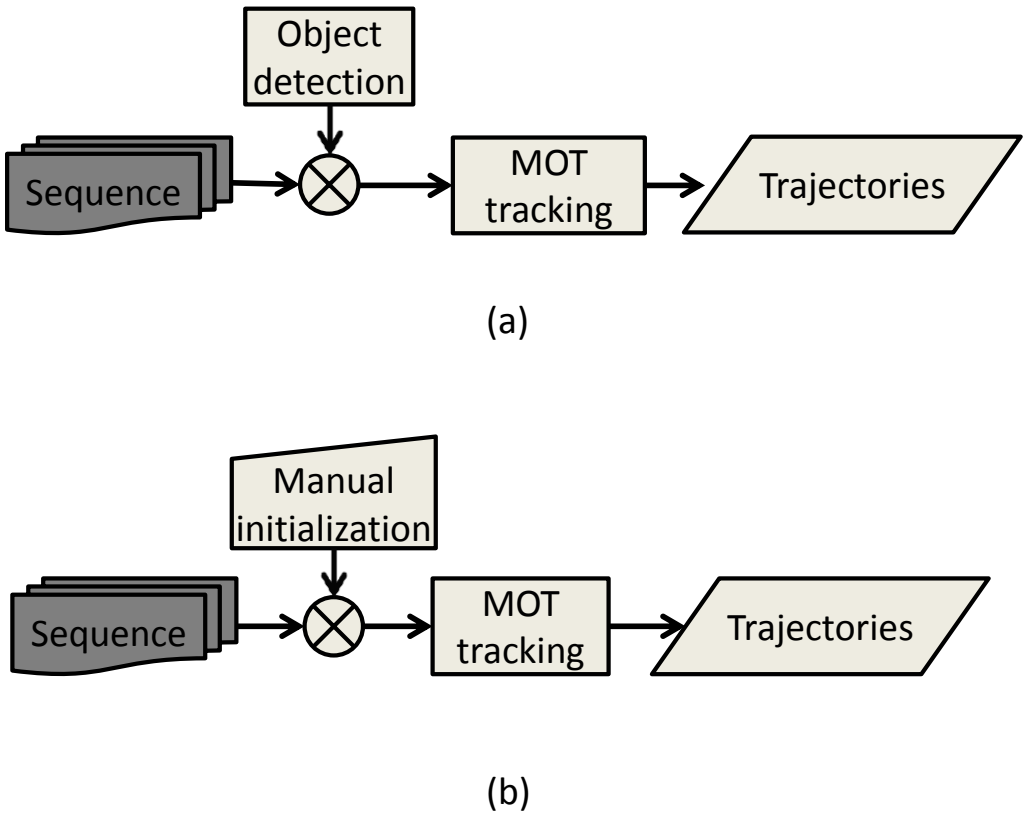

Figure 2.2: *Procedure flow of DBT (a) and DFT (b).*

vehicles or faces. The underlying reason is that detection of these types of objects has gained great progress in recent years [Dalal and Triggs, 2005, Felzenszwalb et al., 2010, Sun et al., 2006]. Third, the performance of DBT depends on the performance of the employed object detectors to a certain extent.

DFT. As shown in Figure 2.2(b), DFT [Hu et al., 2012, Zhang and van der Maaten, 2013b, Zhang and van der Maaten, 2013a, Yang et al., 2007] requires manual initialization of a fixed number of objects in the first frame (in the form of bounding boxes or other shape configurations), then localizes these fixed number of objects in the subsequent frames. Therefore it does not rely on object detector to provide object hypotheses. Noting that, when the

Table 2.1: *Comparison between DBT and DFT. Part of this table is from [Yang and Nevatia, 2012c].*

Item	DBT	DFT
Initialization	automatic, imperfect	manual, perfect
# of objects	varying	fixed
Applications	specific type of objects (in most cases)	any type of objects
Advantages	ability to handle varying number of objects	free of object detector
Drawbacks	performance depends on object detection	requires manual initialization

number of objects is one, DFT degrades into the classical visual tracking problem.

Generally speaking, DBT is more popular since it can handle the occurrence of new objects and disappearance of existing objects naturally. DFT requires manual initialization of each object, thus it cannot deal with the case when new objects appear. However, it is model-free, i.e., free of pre-trained object detectors. So it can deal with sequences of any type of objects. However, the requirements on the fixed number of objects limits its applications in practical systems. Table 2.1 lists the major differences between DBT and DFT.

2.3.2 Processing Mode

According to the way of data processing, MOT could be categorized into online tracking and offline tracking. The difference is whether or not the observations in future frames are utilized when handling the current frame. Online tracking utilizes observations up to the current time to conduct the estimation, while offline tracking employs observations both in the past and in the future.

Online tracking. In online tracking, the image sequence is handled in a step-by-step way, thus online tracking is also named as sequential tracking. Based on the up-to-time observations, trajectories are outputted on the fly.

Sequential approaches derive a cost function considering multiple types of information up to the current frame and estimate the lowest cost state [Kratz and Nishino, 2010, Sugimura et al., 2009, Duan et al., 2012, Breitenstein et al., 2009, Yang et al., 2009] based on sophisti-

cated appearance models [Hu et al., 2012, Shu et al., 2012], delicate motion models [Kratz and Nishino, 2010] and interaction models [Yamaguchi et al., 2011, Pellegrini et al., 2009].

For example, in order to maintain discrimination of individual objects, Yang et al. fuse multiple components: bags of local features, a head model and a color model of torso regions [Yang et al., 2009]. In [Breitenstein et al., 2009], generic object category and instance-specific information are integrated to track multiple objects in a particle filter framework. Inspired by crowd simulation models, a dynamic model considering social motion patterns is introduced in [Pellegrini et al., 2009]. Similarly, Yamaguchi et al. develop an agent-based behavior model taking social and environmental factors into account to predict destinations of pedestrians [Yamaguchi et al., 2011]. The work in [Kratz and Nishino, 2010] estimates object motion based on structured crowd patterns and learns spatio-temporal variations using a set of hidden Markov models.

Offline tracking. Offline tracking [Song et al., 2010, Qin and Shelton, 2012, Yang and Nevatia, 2012a, Yang and Nevatia, 2012b, Brendel et al., 2011, Yang et al., 2011, Kuo et al., 2010, Henriques et al., 2011, Sugimura et al., 2009, Choi and Savarese, 2010] utilizes a batch way to process data, therefore it is also called batch tracking. Observations from all frames are required to be obtained in advance and are investigated together to estimate the final output. Note that, due to computation ability, sometimes it is not possible to handle all the frames at one time. An alternative solution is to divide the whole video into a set of segments or clips, handle these clips respectively, and fuse the results hierarchically.

More specifically, approaches [Izadinia et al., 2012a, Andriyenko and Schindler, 2011, Benfold and Reid, 2011, Brostow and Cipolla, 2006] considering both the past and the future information typically require low-level observations such as foreground, tracklet, trajectory and etc. These types of low level observations can be obtained by background modeling [Song et al., 2010], by associating confident responses of a human detector, head detector or part based detector into tracklets [Qin and Shelton, 2012, Yang and Nevatia, 2012a, Yang and Nevatia, 2012b, Brendel et al., 2011, Yang et al., 2011, Kuo et al., 2010, Henriques et al., 2011, Choi and Savarese, 2010] or by estimating trajectories based on the KLT tracker [Sugimura et al.,

Table 2.2: *Comparison between online and offline tracking.*

Item	Online tracking	Offline tracking
Required input	up-to-time observations	all observations
Methodology	gradually extend existing trajectories with current observations	link observations into trajectories
Advantages	suitable for online tasks	can obtain global optimal solution theoretically
Drawbacks	suffer from shortage of observation	Delay in outputting final results

2009] or Kalman Filter [Choi and Savarese, 2010]. Then, these types of low level observations are associated by optimization methods, such as Markov Chain Monte Carlo (MCMC) [Song et al., 2010], Dynamic Programming, Hungarian algorithm [Qin and Shelton, 2012, Yang and Nevatia, 2012a, Song et al., 2010], greedy bipartite algorithm [Shu et al., 2012], min-cost network flow [Wu et al., 2012, Butt and Collins, 2013b, Zhang et al., 2008], K-Shortest Paths (KSP) algorithm [Berclaz et al., 2011], Conditional Random Fields (CRF) [Yang and Nevatia, 2012b, Milan et al., 2013b] or Maximum Weight Independent Set [Brendel et al., 2011]. Please refer to [Luo et al., 2014b] for a more extensive review.

In general, online tracking is more suitable for cases when video stream is obtained sequentially. Offline tracking typically deals with the data globally when all the frames are obtained, thus it exhibits delay in outputting results. As offline tracking could access all observations simultaneously, theoretically offline tracking could obtain globally optimal solution though it is not as practical as online tracking. Table 2.2 gives a clear comparison between online and offline tracking.

2.3.3 Mathematical Methodology

MOT could be classified into probabilistic tracking and deterministic tracking with regard to the adopted mathematical methodology. The differences are in two folds. First, the methods of estimating object states are different. In probabilistic tracking, the estimation is based on probabilistic inference, while in deterministic tracking the estimation is based on deterministic optimization. Second, the outputs are different. Output of probabilistic tracking may be

different in different running trials, while deterministic tracking gives constant outputs.

2.3.4 Discussion

The insights behind "online vs. offline" and "DBT vs. DFT" are related. The difference between DBT and DFT is whether a detection model is adopted (DBT) or not (DFT). The key to differentiate online and offline tracking is the way they process observations. One may question whether DFT is identical to online tracking because DFT always processes observations sequentially. That is true because DFT is free of (type-specific) object detection. It cannot attain future observations, thus it can only follow the sequential way. Another vagueness may rise between DBT and offline tracking, as in DBT tracklets or detection responses are usually associated in a batch way. Note that there are also sequential DBT which conducts association between previously obtained trajectories and new detection responses [Luo et al., 2014a, Luo and Kim, 2013, Xing et al., 2009].

"Online vs. offline" and "probabilistic vs. deterministic" are also related. In practice, online tracking usually adopts probabilistic inference for estimation. There indeed exists deterministic optimization based online tracking, such as online tracking by linking up-to-time trajectories and detections in the next frame based on Hungarian algorithm. On the other hand, offline tracking always employs deterministic optimization in the derivation of object states.

2.4 Key Components

To track multiple objects, multi-type information can be utilized when calculating the likelihood. Previous works have investigated various information including appearance cues, motion models, interaction, exclusion and occlusion among objects. In the following, some of the typical clues are introduced.

2.4.1 Appearance Model

Appearance is an important cue for affinity computation in MOT. Generally, appearance model includes two components – *visual representation* and *statistical measuring*. Visual representation is closely related to features, but more than just features. It tackles the problem of precisely describing visual characteristics of objects based on features, and can be usually grouped into two sets – visual representations based on single cue and multiple cues. Statistical measuring is the computation of similarity or dissimilarity between different observations when visual representation is ready. In the following, the features/cues employed in MOT are discussed, and then appearance models based on single cue and multiple cues are described respectively.

2.4.1.1 Feature

Different kinds of features have been employed in MOT. They are categorized into the following subsets.

Point features. Point features have been successfully applied in single object tracking [Shi and Tomasi, 1994]. For MOT, point features can also be helpful. For instance, KLT tracker is employed to track feature points and generate a set of trajectories or short tracklets [Sugimura et al., 2009, Zhao et al., 2012]. KLT tracking is utilized in [Benfold and Reid, 2011] to estimate motion. Local feature points [Lowe, 2004] are adopted along with the bag-of-word model in [Yang et al., 2009] to capture the texture characteristics of a region. Point features are also employed in [Brostow and Cipolla, 2006] for motion clustering.

Color/intensity features. This is the commonest feature for MOT. Usually the color or intensity features are employed to calculate the affinity between two counterparts (detection hypotheses, tracklets or short trajectories). The simple raw pixel template is employed in [Yamaguchi et al., 2011] to compute the appearance affinity. Color histogram is used in [Sugimura et al., 2009, Song et al., 2010, Mitzel et al., 2010, Izadinia et al., 2012b, Okuma et al., 2004, Mitzel and Leibe, 2011].

Optical flow. The optical flow feature can be utilized to conduct short-term visual tracking. Thus many solutions to MOT employ optical flow to link detection responses from continuous frames into short tracklets for further data association processing [Rodriguez et al., 2009] or directly use it for data association [Izadinia et al., 2012b]. Besides, optical flow is also employed to complement HOG for observation model [Andriyenko and Schindler, 2011]. Additionally, optical flow is popular in extremely crowded scenarios for discovering crowd motion pattens [Ali and Shah, 2008, Rodriguez et al., 2011].

Gradient/pixel-comparison features. There are some features based on gradient or pixel comparison. Mitzel et al. utilize a variation of the level-set formula to track objects in continuous frames [Mitzel et al., 2010]. Apart from the success in human detection, HOG [Dalal and Triggs, 2005] plays an important role in the multiple pedestrian tracking problem as well. For instance, HOG is employed in [Izadinia et al., 2012b, Kuo et al., 2010, Breitenstein et al., 2009, Choi and Savarese, 2012, Yu et al., 2008] to detect objects and/or compute similarity between detections for data association.

Region covariance matrix features. Region covariance matrix [Porikli et al., 2006, Tuzel et al., 2006] features are robust to issues such as illumination changes, scale variations, etc. The region covariance matrix based dissimilarity is used to compare appearance in [Henriques et al., 2011]. Covariance matrices along with other features constitute the feature pool for appearance learning in [Kuo et al., 2010]. Hu et al. utilize the covariance matrix to represent object for both single and multiple object tracking [Hu et al., 2012].

Depth. Mitzel et al. utilize depth information to correct bounding box of detection response and re-initialize the bounding box for level-set tracking in [Mitzel et al., 2010]. Depth information is used in [Ess et al., 2009, Ess et al., 2007] to refine detection responses. Similarly, Ess et al. employ depth information to obtain more accurate object detections in a mobile vision system [Ess et al., 2008]. The stereo depth is taken into account by Giebel et al. to estimate weight of a particle in the proposed Bayesian framework for multiple 3D object tracking [Giebel et al., 2004].

Others. Some other features are utilized to conduct multiple object tracking as well. For

instance, gait features in the frequency domain are employed in [Sugimura et al., 2009] to maximize the discrimination between the tracked individuals. The Probabilistic Occupancy Map (POM) [Fleuret et al., 2008, Berclaz et al., 2011] is employed to estimate how probable an object would occur in a specific grid under the multi-camera settings for MOT.

2.4.1.2 Single Cue based Appearance Model

Raw pixel template representation. The raw pixel template representation is the raw pixel intensity or color of a region. It encodes the spatial information since the comparison is element wise when matching two templates. Normalized Cross Correlation (NCC) is used to evaluate the predicted position of object in [Yamaguchi et al., 2011]. The appearance affinity is calculated as the NCC between the target template and a candidate bounding box in [Ali and Shah, 2008]. Wu et al. build a network-flow approach to handle multiple target tracking [Wu et al., 2012]. When they compute the transitional cost on the arcs of the network as flows, the normalized cross correlation between the upper one-fourth bounding boxes of the corresponding two detection observations is used.

Color histogram representation. Color histogram is effective to capture the statistical information of target region. For example, distance between color histograms are used to calculate likelihood in [Kratz and Nishino, 2010]. Similarly, to capture the dissimilarity, Sugimura et al. use the Bhattacharyya distance between hue-saturation color histograms when constructing a graph [Sugimura et al., 2009].

Covariance matrix representation. The covariance matrix descriptor is employed to represent the appearance of an object in [Henriques et al., 2011, Hu et al., 2012].

Bag of words representation. Fast dense SIFT-like features [Lowe, 2004] are computed in [Yang et al., 2009] and encoded based on the bag-of-word model. To incorporate spatial information, the Spatial Pyramid Matching (SPM) method [Lazebnik et al., 2006] is adopted. It is used as an observation model for appearance modeling.

2.4.1.3 Multi-cue based Appearance Model

Different kinds of cues can compensate each other, making appearance model more robust. However, there arises an issue that how to fuse the information from multiple cues. Regarding this, the following discusses over five popular strategies for multiple cues.

Boosting. The strategy of Boosting usually selects a portion of features from a feature pool sequentially via a Boosting based algorithm (e.g. Adaboost by [Kuo et al., 2010] and RealBoost by [Yang and Nevatia, 2012c]). Features are selected according to their discrimination power. A discriminative appearance model is proposed in [Kuo et al., 2010] to assign high similarity to tracklets which are of the same object, but low affinity to tracklets of different objects. Similarly, Yang et al. employ the standard RealBoost algorithm to learn the feature weights from training sample set [Yang and Nevatia, 2012c]. A HybridBoost algorithm is proposed in [Li et al., 2009] to automatically select features with maximum discrimination.

Concatenating. A SVM model classifier is trained to distinguish a specific target from targets in its temporal window. Color, HOG and optical flow are concatenated and further processed with PCA projection for dimension reduction to describe detection responses [Brendel et al., 2011].

Summation. Mitzel et al. integrate color information with depth information to simultaneously segment and track multiple objects [Mitzel et al., 2010]. The probabilities computed from color and depth are weighted by a parameter. The similar weighting strategy is adopted in [Liu et al., 2012] to balance two cues of raw pixel intensity and silhouette.

Product. A formula of production is adopted in [Song et al., 2010]. The likelihood considering color histogram is multiplied with the likelihood regarding foreground response to compute the final likelihood in the observation model. Likelihoods in terms of shape, texture and depth are multiplied to be the weight of a particle in the Bayesian framework [Giebel et al., 2004]. Dividing the scene under multiple cameras into multiple grids, appearance model is constructed based on color model and ground plane occupancy estimation [Berclaz et al., 2006]. Similarity concerning these two cues are multiplied in the MAP formula.

Cascading. Cues of depth, shape and texture are utilized in a cascade manner to narrow the search space for multiple object detection and tracking in [Gavrila and Munder, 2007]. Real-time performance is achieved by doing so. The similar idea is also used in [Izadinia et al., 2012b].

In this thesis, color histogram, HOG and LBP are employed as features to build appearance model in Chapter 3 and Chapter 4 for their efficiency. In modeling appearance, they are jointly used in a concatenating way. Super-pixel and Deformable Part Model are employed in Chapter 5 for appearance modeling. These two kinds of representation are effective and kind of robust to occlusion.

2.4.2 Motion Model

Object motion model is important for multiple object tracking since it can predict the potential position of objects in future frames, to reduce search space. In general, objects are assumed to move smoothly (*cf.* the abrupt motion is a special case). Most of existing motion models can be divided into the following two classes.

2.4.2.1 Constant Velocity Motion Models/Linear Motion Models

Objects following this kind of models are assumed to move with constant velocity [Shafique et al., 2008, Yu et al., 2007]. The velocity of object in the next frame is the same as the current velocity (added by process noise independently drawn from some types of distributions). For example, Breitenstein et al. employ a constant velocity motion model to propagate particles [Breitenstein et al., 2009]. In their model, the more the number of successfully tracked frames is, the smaller the variance will be. Method in [Andriyenko and Schindler, 2011, Milan et al., 2014] also adopts such constant velocity models. To be specific, the model considers differences between the velocities of one object in different time instants. Intuitively, it penalizes the difference between velocities and forces trajectories to be smooth. A constant velocity model considering both the forward velocity and the backward velocity is proposed

in [Xing et al., 2009] to compute the affinity between two tracklets in terms of motion. The forward-direction motion (the backward-direction motion vice visa) is described by a Gaussian distribution centered in the position of the head response of one tracklet. Then it estimates the probability of the position plus forward displacement of tail response of the other tracklet. Different from traditional graph based MOT approaches which treat each node as an individual observation (e.g., one detection response), node is treated as a pair of tracklets in [Yang and Nevatia, 2012b]. The affinity in terms of motion is calculated based on the displacement between the estimated positions via a linear motion model and the observed positions. This motion model is essentially the same as the one in [Xing et al., 2009] and widely applied in many works [Kuo et al., 2010, Kuo and Nevatia, 2011, Yang et al., 2011, Qin and Shelton, 2012, Nillius et al., 2006]. However, it only considers the pair of tracklets itself. A motion model concerning two pairs of tracklets is proposed in [Yang and Nevatia, 2012b]. Apart from considering position and velocity, Kuo and Nevatia also take the accelerate rate into consideration in [Kuo and Nevatia, 2011].

2.4.2.2 Non-linear Motion Model

A non-linear motion model is proposed in [Yang and Nevatia, 2012a] to produce more accurate motion affinity between tracklets. Given two tracklets $\mathbf{T}_1$ and $\mathbf{T}_2$ which belong to the same target in Figure 2.3(a), a linear motion model would produce low probability to link them. Employing the nonlinear motion model, the gap between tail of tracklet $\mathbf{T}_1$ and head of tracklet $\mathbf{T}_2$ could be reasonably explained by a support tracklet $\mathbf{T}_0$. As shown in Figure 2.3(b), elements of $\mathbf{T}_0$ are matched with the tail of $\mathbf{T}_1$ and the head of $\mathbf{T}_2$. Then the real path to bridge $\mathbf{T}_1$ and $\mathbf{T}_2$ is estimated based on $\mathbf{T}_0$.

In this thesis, motion smoothness is assumed in all chapters. Meanwhile, in Chapter 4 a linear motion model based on cosine similarity is developed for its efficiency.

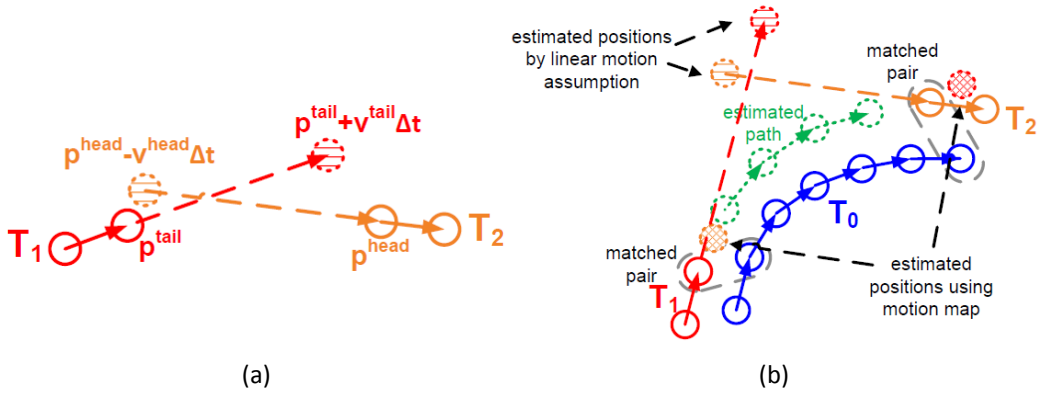

Figure 2.3: *An image comparing the linear motion model (a) with the non-linear motion model (b) [Yang and Nevatia, 2012a]. Best viewed in color.*

2.4.3 Interaction Model

Interaction model captures the influence of an object to other objects. In the crowd scenery, an object would be affected by "force" from others. For instance, when a pedestrian is walking on the street, he would consider his speed, direction and destination, in order to avoid collision with others. Another example is that when a group of people walks across a street, each of them follows others and guides others at the same time. They form a motion pattern and every one follows this pattern. In fact, these are examples of two typical interaction models known as the *social force models* [Helbing and Molnar, 1995] and the *crowd motion pattern models* [Hu et al., 2008].

2.4.3.1 Social Force Models

In social force models, objects are considered as agencies which determine their speed, velocity, destination based on observations of other objects and the around environment. More specifically, target behavior is modeled based on *individual force* and *group force*.

Individual force. For each individual in the scenario of multiple objects, two types of force are considered: 1) *fidelity*, which means one should not change his desired destination and

2) *constancy*, which means one should not suddenly change his velocity, including speed and direction.

Group force. For a whole group, three types of force are considered: 1) *attraction*, which means individuals moving together as a group should stay close, 2) *repulsion*, which means that individuals moving together as a group should keep some distance away from others to make all participants comfortable and 3) *coherence*, which means individuals moving together as a group should be with similar velocity.

The majority of existing publications with social force model follows the concepts of these two types of force. For instance, by minimizing an energy function considering sudden speed change and drift from destination for an object, the search space of its destination is largely reduced in [Pellegrini et al., 2009]. Consequently, the data association procedure is further simplified . In [Yamaguchi et al., 2011] the destination of an object is determined by considering the so-called personal, social and environmental factors, which are formulated as extra terms in a cost function. Social grouping behavior is considered to improve the performance of data association for MOT [Qin and Shelton, 2012]. To be specific, people are assumed to form K groups and every tracklet assigned to the same group should be consistent with the group mean trajectory. Two factors, repulsion and group motion, are considered in [Choi and Savarese, 2010]. The repulsion factor tries to separate objects if they are too close to each other. Group motion factor assumes the relative distance between two objects in continuous two frames should keep unchanged. A social force model of four components is proposed in [Scovanner and Tappen, 2009] to learn dynamics of pedestrians in the real world. These four components contribute four energy terms which consider avoidance of jump in the space grid, constant velocity, destination and collisions respectively. These four energy terms are weighted to form an energy objective which is then minimized to predict the movement of a target.

2.4.3.2 Crowd Motion Pattern Models

Inspired by the crowd simulation literature [Zhan et al., 2008], motion patterns are introduced to alleviate the difficulty of tracking an individual object in crowd. In general, this type of models is applied in the highly-crowded scenario where objects are usually quite small. In this case, cues such as appearance and individual motion are ambiguous, thus motion from the crowd is a comparably reliable cue.

There have been some works in this direction. For example, an assumption is made that the behavior of an individual is determined by the scene layout and the surrounding objects in [Ali and Shah, 2008] and three kinds of force from the floor fields are proposed. These fields are Static Floor Fields (SFF), Boundary Floor Field (BFF) and Dynamic Floor Field (DFF). SEF considers the scene structure. BFF takes the barriers in the scene into consideration. DFF captures the motion of a crowd around to determine the future positions of objects in the crowd. Observing that a group of pedestrians exhibits collective spatio-temporal structure, movement of an object within any local space-time location of a video are learned by training a set of Hidden Markov Models (HMM) in [Kratz and Nishino, 2010, Kratz and Nishino, 2012]. The entire video is divided into local spatio-temporal cubes. The motion pattern of a specific spatio-temporal cube is represented as a 3D Gaussian distribution considering the 3D gradients of all pixels in the cube. This motion pattern is assumed to vary through time and exhibits the Markov property. Thus the future motion pattern could be predicted based on the previous states, and the predicted motion pattern can constrain tracking of objects in the concerned spatio-temporal location. Correlated Topic Model (CTM) is adopted in [Rodriguez et al., 2009] to learn various motion behaviors in the scene. A tracker which can predict a rough displacement based on scene codebook from all the moving pixel in the unstructured scene, along with the learned high-level behavior, are weighted to track objects. Similar to image retrieval, motion pattern could also be retrieved in [Rodriguez et al., 2011]. Motion patterns are firstly learned in an unsupervised and offline manner from a database composed of a large number of videos. Then given a test video, a set of space-time patches are matched to obtain motion prior, which assists object tracking.

2.4.4 Exclusion Model

Exclusion is a constraint due to physical collisions. Given multiple detection responses and multiple trajectory hypotheses, there are generally two constraints to be considered. The first one is the so-called *detection-level exclusion* [Milan et al., 2013a], i.e., two different detection responses in the same frame cannot be assigned to an identical trajectory hypothesis. The second one is the so-called *trajectory-level exclusion*, i.e., two trajectories cannot occupy an identical detection response.

2.4.4.1 Detection-level Exclusion Modeling

The detection-level exclusion is explicitly modeled by defining a cost term to penalize the case if two simultaneous and distant detection responses are assigned the same label of trajectory with a cost in [Milan et al., 2013a]. Label propagation is employed in [KC and De Vleeschouwer, 2013] for multiple object tracking. To model exclusion, a special exclusion graph is constructed to capture the constraint that detection responses with the same time stamp (occurring at the same time) should have different labels.

2.4.4.2 Trajectory-level Exclusion Modeling

To model the trajectory level exclusion, Milan et al. penalize the case that two close trajectories $\mathbf{T}_i$ and $\mathbf{T}_j$ have different labels [Milan et al., 2013a]. The penalty is proportional to the spatial-temporal overlap between $\mathbf{T}_i$ and $\mathbf{T}_j$. The closer the two trajectories, the higher penalty it is. Similarly, mutual exclusion is modeled as an additional cost term to penalize the case that two trajectories are very close to each other. The cost is reversely proportional to the minimum distance between the trajectories in their temporal overlap [Andriyenko et al., 2012]. By doing so, one of the trajectory would be abandoned to avoid the collision. Exclusion is also modeled as a constraint in the objective function of network flow in [Butt and Collins, 2013a].

In this thesis, non-maximum suppression is employed to tackle the trajectory-level exclusion in Chapter 3 and Chapter 4. The detection-level exclusion is naturally handled by the

exclusion of membership assignment of clustering, and the trajectory-level exclusion is addressed by the cannot-link in Chapter 5.

2.4.5 Occlusion Handling

Occlusion can lead to ID switch or fragmentation of trajectories. In order to handle occlusion, various kinds of strategies have been proposed.

2.4.5.1 Part-to-whole

This strategy is built on the assumption that, part of the object is still visible when occlusion happens. Based on this assumption, this strategy observe and utilize the visible part to infer state of the whole object. Hu et al. propose a block-division model to deal with occlusion [Hu et al., 2012]. In this model, object is divided into multiple blocks without overlap. This model brings benefits of two folds to the tracking problem under occlusion. Firstly, spatial information is considered as likelihood of an observation is the product of likelihood of all its blocks. Secondly, an occlusion map could be obtained according to reconstruction errors of all blocks which could be further utilized to selectively update appearance model. Part based appearance model is learned to distinguish an object from other objects around and the background in [Yang and Nevatia, 2012c]. To explicitly deal with occlusion, object is represented with 15 parts. Once a part is found occluded, all the features from that part are assumed to be invalid. The appearance model is learned via a boosting algorithm. Part based model is also applied in [Izadinia et al., 2012b] as a multi-person multi-part tracker. The whole body and individual body parts are tracked and the final trajectory estimation is obtained by jointly considering association between the whole human body and the individual human body parts. In case of occlusion, the visible parts are detected and trajectories of visible parts are estimated. Along with the trajectory of the whole body, the complete trajectory is recovered. A similar part based model for occlusion handling is also proposed in [Shu et al., 2012].

2.4.5.2 Hypothesize-and-test

This strategy sidesteps challenges from occlusion by hypothesizing proposals and testing the proposals according to observations at hand. For example, an Explicit Occlusion Model (EOM) is proposed in [Zhang et al., 2008] and integrated into the cost-flow framework to handle long-term occlusion. Occlusion hypotheses are generated based on occlusion constraints. If the distance and scale difference between two observations are small enough, then they are occludable. The generated hypotheses along with the original observations (tracklets) are given as input to the cost-flow framework and MAP is conducted to obtain the optimal solution. The model adopted in [Tang et al., 2013, Tang et al., 2014] also follows a hypothesize-and-test fashion to handle occlusion. Specifically, a double-person detector is built to be aware of different levels of occlusion between two people. They train a double-person detector based on instances generated by synthetically combining two objects with different levels of occlusion. Along with the traditional single person detector, this multi-person detector is employed as the basis of multiple object tracking.

2.4.5.3 Buffer-and-recover

This strategy buffers observations when occlusion happens and remembers states of objects before occlusion. When occlusion ends, object states are recovered based on the buffered observations and the stored states before occlusion. Mitzel et al. combine a level-set tracker and a high-level tracker based on detection in [Mitzel et al., 2010]. The high-level tracker is employed to initialize new tracks from detection response and the level-set tracker is used to tackle the frame-to-frame data association. To tackle occlusion, the high-level tracker keeps a trajectory alive for up to 15 frames when occlusion happens, and extrapolates the position to grow the dormant trajectory through occlusion. In case the object reappears, the track is triggered again and the identity is maintained. Ryoo and Aggarwal propose an “observe-and-explain" strategy to handle the inter-object occlusion and scene-object occlusion [Ryoo and Aggarwal, 2008]. Their strategy saves computation cost as an observation mode is activated

when the state of tracking is not clear due to occlusion. When they get enough observations, explanations are generated. It could also be treated as a “buffer-and-recover" strategy.

2.4.5.4 Others

The strategies described above do not cover all the tactics in the community. On one hand, in practice there exists a method which addresses occlusion based on overlap between detection bounding boxes. It is simple but works in some cases. On the other hand, the three strategies above are not mutually exclusive. Sometimes they are combined and used simultaneously.

2.5 Evaluation Metrics & Data Sets

Evaluation of MOT approaches is based on metrics from [Keni and Rainer, 2008, Li et al., 2009] as follows:

MOTA combines the false positive rate, false negative rate and mismatch rate for MOT.

MOTP simply calculates the average overlap between the ground truth and the estimated objects.

MT is the percentage of the ground-truth trajectories which are covered temporally for over 80% in time.

ML is the percentage of the ground-truth trajectories which are recovered for less than 20% in length.

FM metric counts the number of interruptions of the ground-truth trajectories.

IDS counts the number of times that the ground-truth trajectories change their matched IDs.

To evaluate the proposed methods in this thesis, data sets shown in 2.3 are employed.

Table 2.3: *Details of data sets employed in this thesis.*

Name	# of frames	Resolution
Zebra	28	480×272
Crab	210	640×368
Antelope	173	480×272
Hockey	101	360×240
Flower	500	360×240
Goose	152	640×360
Sailing	243	480×368
Airshow	201	640×368
TUD-Stadtmitte	179	640×480
ETHMS-Sunnyday	354	640×480
ETHMS-Bahnhof	999	640×480
ParkingLot	748	1920×1080

As the problem of GMOT is relatively new in the computer vision community, the state-of-the-art performance of existing methods is difficult to define and given. However, in experimental sections of the following chapters, performance of existing popular methods applied to this problem would be reported.

2.6 The Most Relevant Work

In recent years, some researchers have attempted to investigate generalization of the MOT problem to any kind of objects. The ideal method should involve minimal manual efforts on labeling and should not require any offline detectors. In [Zhao et al., 2012], Zhao et al. propose to track multiple similar objects by requiring one instance in the first frame to be labeled. They firstly track this object, collect training samples, and train an object detector for this kind of objects in the first few frames. Then they start from the first frame again, detect the top M (specified by the user) similar objects and track them in the subsequent frames. Compared with DFT, this work saves much labeling labor. However, the number of objects to track is still fixed. Multiple similar objects are tracked in [Dicle et al., 2013]. However, the detection responses are given as inputs to the algorithm rather than by detection. Brostow and Cipolla deal with GMOT without any supervision [Brostow and Cipolla, 2006]. Assuming that a pair

of points that appears to move together is likely to be part of the same individual, feature points are tracked and motion clustering is conducted to discover and maintain identical entities.

CHAPTER 3

GENERIC MOT BY MULTI-TASK LEARNING

For the problem of generic multiple object tracking, it is relatively easy when objects are isolated or can be clearly distinguished from background and other objects. However, in crowd scenarios, there are frequent occlusions and interactions among objects and many objects have similar appearance. All these issues lead to confusion. A large volume of studies have tackled these challenges. Owing to the great success in object detection (especially human or pedestrian detection), most current approaches take the tracking-by-detection strategy for MOT problems, and good results are reported on some public data sets. However, existing methods for MOT mainly rely on a pedestrian detector and thus have been applied to sequences of pedestrians only, rather than sequences of general type objects.

In this chapter, a method for tracking multiple objects of a general type by the tracking-by-detection strategy is proposed. Similar to multiple pedestrian tracking, a detector which is aware of objects of a generic type is required, and multiple trackers can track these discovered objects individually. From the methodological perspective, this is a problem composed of two

stages. In the first stage, it is treated as a binary classification problem, which has a goal of distinguishing objects from background. In the second stage, each object is discriminated from other objects via tracking, thus it can be considered as a multi-class classification problem.

In the aforementioned two-stage problem, each sample has two kinds of labels. For detection, the label is "object" or "background". For tracking, its label is "object $\boldsymbol{i}$" or not. This problem differs from the traditional multi-object tracking problem, where target objects (pedestrians), despite being of the same type, have quite different appearances due to e.g. clothes. In the concerned problem, target objects are of the same type and visually more alike (see Figure 3.1). Similar objects can be jointly modeled effectively, and this motivates me to formulate the problem as a Multiple Task Learning (MTL) problem.

In the MTL literature, it has been proven helpful to learn related tasks jointly rather than individually. The relevance among the tasks is typically encoded by sharing a common part of features or embedding the learners in a low rank subspace. As mentioned above, there are two main tasks – detection and tracking, and the main tracking task is further partitioned into multiple sub-tasks. Correspondingly, there are one detector and multiple trackers in the classifier space. Taking the learning of them as multiple tasks, the relevance among them is modeled in two folds. Globally, an algorithm called the Mean-Regularized Joint Feature Learning is proposed to associate the two main tasks in the manner that the trackers are learned not to deviate much from the mean i.e. the detector. Locally, the multiple trackers are associated by sharing common features.

Moreover, most previous methods for multi-object tracking train a detector off-line and then classify each testing sample i.e. a scan-window independently (thus locally). In contrast, contextual information such as similarities among samples can help learn a better detector. The Laplacian SVM [Belkin et al., 2006] which includes a smoothness constraint among all labeled and unlabeled samples at present (thus globally) is employed for object detection. The smoothness term is also incorporated into tracking, yielding better trackers.

The main contribution of the proposed method is threefold:

- Objects of a general type rather than pedestrians are considered for MOT in crowds.

To the best of my knowledge, this is the first attempt to tackle the detection and tracking of multiple generic objects in crowds.

• This problem (GMOT) is formulated into MTL. I propose a novel Mean Regularized Joint Feature Learning method. In the method, the detection and tracking of a general type of multiple objects are linked using the detector as the mean to regularize the multiple trackers. Additionally, sharable features are selected among different trackers to better relate one tracking task to the others.

• Formulations of a linear Laplacian SVM classifier are derived for detection. The smoothness term in the modified linear Laplacian SVM enables to view the candidates globally. The linear classifier is easy to incorporate into the MTL framework. A smoothness term is also introduced into the learning of multiple trackers.

With regard to the generality of objects' types, this work is related to multi-class object detection [Wu and Nevatia, 2007a, Mikolajczyk et al., 2006, Torralba et al., 2007] to some extent. However, for multi-class object detection, the classes of objects are known in advance and there are sufficient training samples available to train good classifiers. In the concerned problem, object type is not known and training data can only be collected online.

3.1 Multiple Task Learning

Generally, there are multiple tasks in Multiple Task Learning (MTL) [Caruana, 1997], whereas each task is related to other tasks. The motivation of multi-task learning is that learning multiple related tasks simultaneously outperforms learning them independently. The benefits are: (1) sharing information among multiple tasks; (2) joint feature learning; (3) capability of training without sufficient training data; (4) and so on. Let me take an example to give an intuitive illustration of multi-task learning. Assuming there are a few schools and student properties such as student ID, student age, student height as training samples associated with each school. For each school, there is a set of training samples. There are multiple tasks here, i.e., for each

school, the task is to predict the score of a student in this school. Obviously, these multiple tasks are related to each other as they are dealing with the same problem. One issue is that there are scarce training samples for each task. Thus training a model for each task with the limited training data probably leads to poor generalization ability. Multi-task learning can help here to train these multiple tasks at the same time in order to share information among them.

The most important issue of multi-task learning is how to model the relevance among multiple tasks. Appropriate modeling of the relevance among multiple tasks would lead to performance boost, which is the motivation and benefit of learning multiple tasks simultaneously rather than independently. However, if the relationship among multiple tasks is not modeled appropriately, decrease in the performance would probably happen.

Assuming there are m tasks and m learners, let these multiple learners be $\mathbf{W} = [\mathbf{w}_1, \mathbf{w}_2, ..., \mathbf{w}_m]$, where $\mathbf{W} \in \mathbb{R}^{d \times m}$ and d is the dimension of feature space. For the i-th task, there are training samples as $\mathbf{X}_i \in \mathbb{R}^{d \times N_i}$ and labels or groundtruth $\mathbf{y}_i \in \mathbb{R}^{N_i}$. In most cases, to learn multiple learners at the same time, one should minimize a cost function composed of two sub-parts. One is a cost term $f(\bullet)$ from the training data, and the other one is a regularization term $g(\bullet)$ which models the relationship among multiple learners. It can be written as

$$\mathcal{L} = f(\mathbf{W}, \mathbf{X}, \mathbf{y}) + g(\mathbf{W}) \ . \tag{3.1}$$

The cost term is the same as an ordinary term that usually adopts the least square form, or the Hamming distance to measure the difference between the model prediction and the groundtruth of training data. For instance, the least square form of the cost term is:

$$f(\mathbf{W}, \mathbf{X}, \mathbf{y}) = \sum_{i=1}^{m} \frac{1}{N_i} \parallel \mathbf{X}_i^T \mathbf{W}_i - \mathbf{y}_i \parallel^2 \ . \tag{3.2}$$

In recent years, typical ways of associating multiple tasks include:

- Mean regularized MTL [Evgeniou and Pontil, 2004] which assumes that all tasks are related to each other, and all tasks are regularized to not drift away from the mean of all

tasks. Intuitively, the regularization term penalizes the deviation of each task from the mean, which would try to make them as close as possible.

- Embedded feature selection [Liu et al., 2009, Obozinski et al., 2010] which aims to learn/select some features more expressive for multiple tasks, so it is also called joint feature learning. Usually these features are selected by assuming that all the models share a set of common features. In formulation, this constraint is modeled as group sparsity of model vectors $\mathbf{W}$. Obviously, as the sparsity works, some dimensions of model vectors $\mathbf{W}$ would be zero. This procedure chooses the features corresponding to the non-zero dimensions of $\mathbf{W}$.

- Low-rank subspace learning [Ji and Ye, 2009] which captures the relatedness among multiple tasks. Assuming all the model vectors share a subspace, the regularization term is usually represented by the rank of the model vectors $\mathbf{W}$ as $Rank\,(\mathbf{W})$. However, as the rank minimization is NP hard in practice, it is usually relaxed to the trace norm which is theoretically shown to be a good approximation for the rank function.

- Clustered MTL [Zhou et al., 2011b] which supposes that tasks have a clustered structure, and tasks in the same cluster are closer to each other compared with the ones in another cluster. Based on this, the clustered MTL captures the relevance among multiple tasks similar to the K-means clustering.

- Tree regularized MTL [Kim and Xing, 2010] which employs the tree structure to model the relevance among multiple tasks. Within the tree structure, tasks corresponding to the nodes with the same parent node are close to each other, and the similarity between nodes/tasks are determined by the common depth that these nodes share in the tree structure.

- Graph regularized MTL [Chen et al., 2010] which utilizes the graph structure to represent the relationship among task models. In the graph structure, each vertex indicates a task model, and the edge connecting two vertexes measures the similarity between the two

tasks by the weight associated with it. One way to regularize the multiple tasks is to penalize the difference between two tasks.

In terms of applications in the computer vision community, MTL is combined with the boosting framework to learn the features shared by multiple classes to conduct multi-class detection. By doing so, it can avoid construction of a specialized classifier for each class in [Torralba et al., 2007]. MTL is also utilized to handle single object tracking in [Zhang et al., 2012] by treating representation of multiple particles based on the collected templates as multiple tasks.

3.2 Methodology

In this approach, the tracking-by-detection strategy is employed for multiple object tracking in crowds. Given the initial bounding box of an arbitrary target object in the first frame, a classifier is trained to discriminate all target objects from background. For each of the detected objects, an individual tracker is trained by taking the corresponding object as a positive sample, and other objects around it and random background patches as negative samples. Then the trackers are used to follow those objects in the subsequent frames respectively. After processing every few frames, objects which are tracked confidently are selected to retrain the detector. When the detector is aware of new objects or disappearance of existing objects, new trackers are generated or the existing trackers are deleted. Figure 3.1 illustrates the overview of the approach.

3.2.1 Generic Detector

For detection, candidates are generated using the sliding window strategy [Viola and Jones, 2004]. Like the previous work, most of the candidates can be rejected confidently. However, unlike pedestrian detection, the type of objects is unknown. To tackle this problem, some

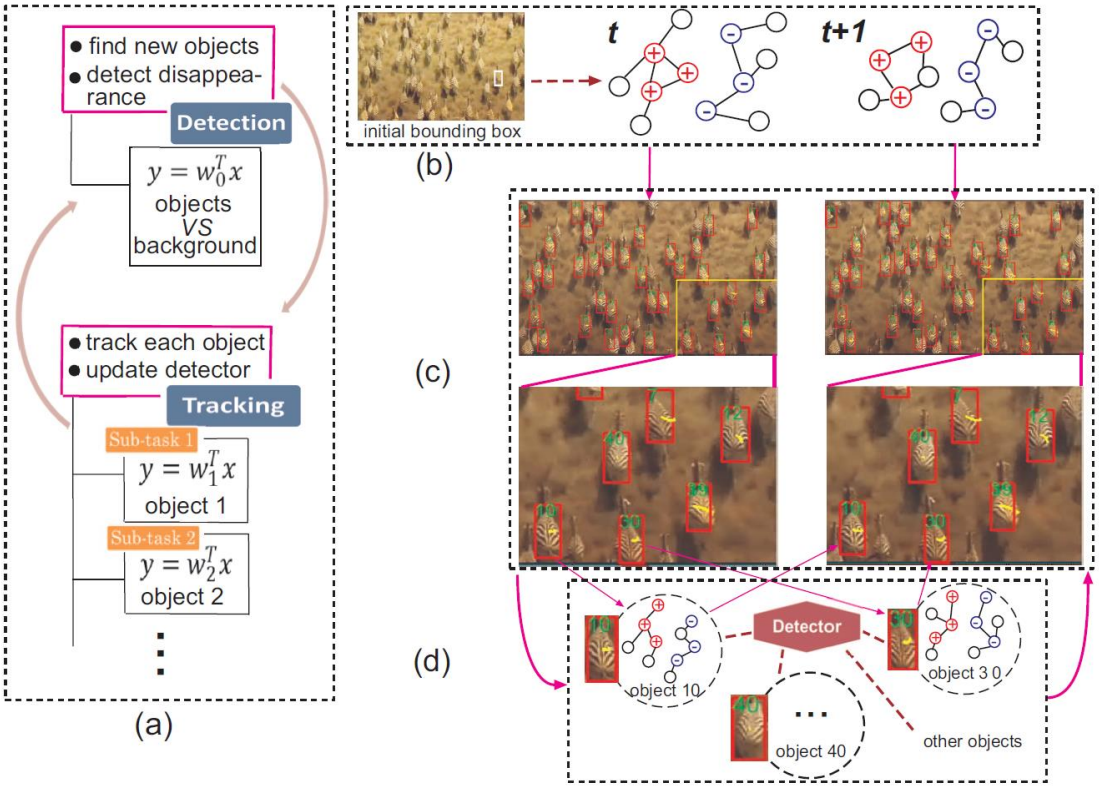

Figure 3.1: *An overview of the proposed approach. (a) Problem decomposition within the MTL framework. (b) Detection by the linear Laplacian SVM (given one initial bounding box). Graphs of two continuous frames are shown here. (c) Tracking results of continuous two frames. I zoom in a part of the image to give a clear view. (d) Tracking by the detector regularized multiple trackers (see Section 3.2.2). Each object is associated with a graph. The dotted line between each tracker and the detector indicates their association. This figure is best viewed in color.*

efficient criteria including Region Variance, Edge Density and Color Contrast are employed in the following to measure the objectness [Alexe et al., 2010] of candidate windows and then reject candidates which are not likely to be objects.

Region Variance. This criterion computes the variance of the pixels within a candidate region as $RV = \frac{1}{N_R - 1} \sum_i (g_i - \bar{g})$, where N_R is the number of pixels in the region, g_i is the gray intensity of pixel i and $\bar{g}$ is the mean intensity of all the pixels in this region. This criterion can reject some candidates from the background such as grass or sky.

Edge Density. This criterion calculates the density of edge pixels within a region as $ED = \frac{1}{N_R} \sum_i 1\{i \in \mathcal{E}_\mathcal{R}\}$, where $1\{\cdot\}$ is the indicator function, $\mathcal{E}_\mathcal{R}$ is the set of pixels which belong

to edge. This criterion helps to reject candidates which are too smooth. Note that in [Alexe et al., 2010] the Edge Density is also a cue to measure the objectness, but here I use different methods to calculate the edge density.

Color Contrast. The Color Contrast cue is borrowed as $CC(\theta_{CC}) = \chi(\mathbf{h}_{Region}, \mathbf{h}_{Surr(\theta_{CC})})$ from [Alexe et al., 2010] to measure the objectness of a window. $\mathbf{h}_{Region}$ is the color histogram of the region and $\mathbf{h}_{Surr(\theta_{CC})}$ is the color histogram of the surrounding of the region (θ_{CC} measures how large the surrounding is). $\chi(\cdot,\cdot)$ is the chi-square distance function. Although this criterion is used in [Alexe et al., 2010] where only one object is in the image scene, it is also helpful to reject some windows in my case.

Typically the number of sliding windows is greater than $30,000$, and the number of windows survived from these three rejecters is about 1000. This enables me to adopt an elaborate detector. The survived windows are treated as unlabeled samples and written as $\mathbf{X}_u = [\mathbf{x}_1, ..., \mathbf{x}_{n_u}]$, where $\mathbf{x}_i \in \mathbb{R}^d$, d is the dimension of the feature space. As an initial bounding box has been given as a target object, the positive sample set is augmented by adding some slight disturbance to it. At the same time, instances are sampled in a further distance (between r_1 and r_2) as negative data. The corresponding labels of them are $y_i \in \{1, -1\}, i = 1, \cdots, n_l$, where $y_i = 1$ means x_i is object and $y_i = -1$ corresponds to non-object (background). Along with the unlabeled candidates, all the n samples are written as $\mathbf{X} = [\mathbf{X}_l,\ \mathbf{X}_u] \in \mathbb{R}^{d \times n}$.

The detector is defined as $f(\mathbf{x}) = \mathbf{w}_0^T \mathbf{x}$, where $\mathbf{w}_0 \in \mathbb{R}^d$. To tackle the detection problem, the following objective function is minimized:

$$\mathcal{L}_p = \gamma_1 \|\mathbf{w}_0\|^2 + \gamma_2 \mathbf{w}_0^T \mathbf{X}\mathbf{L}\mathbf{X}^T \mathbf{w}_0 + \gamma_3 \sum_{i=1}^{n_l} [1 - y_i f(\mathbf{x}_i)]\,. \tag{3.3}$$

In Equation 3.3, the first term is the regularization of the classifier to improve the generalization ability, the second term is the smoothness among all the samples and the third term is the fitting error of the labeled samples. $\mathbf{L}$ is the Laplacian matrix calculated from the graph constructed based on all the samples. It is notable that this objective function has the same form as Laplacian SVM [Belkin et al., 2006]. However, here I modify the original Laplacian SVM to the linear case.

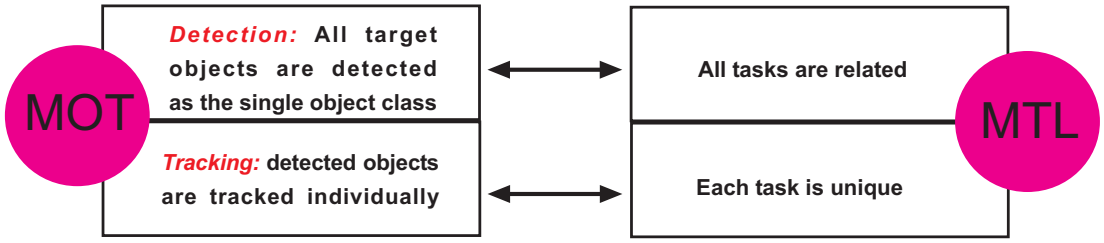

Figure 3.2: *Formulation of the MOT problem into MTL. This figure is best viewed in color.*

Introducing the slack variables ε_i, the primal problem is:

$$\begin{aligned} \min_{\mathbf{w}_0,\varepsilon_i} \quad & \gamma_1\|\mathbf{w}_0\|^2 + \gamma_2\mathbf{w}_0^T\mathbf{X}\mathbf{L}\mathbf{X}^T\mathbf{w}_0 + \gamma_3\sum_{i=1}^{n_l}\varepsilon_i\,, \\ s.t. \quad & y_i\mathbf{w}_0^T\mathbf{x} \geq 1-\varepsilon_i,\ i=1,2,...,n_l\,, \\ & \varepsilon_i \geq 0,\ i=1,2,...,n_l\,. \end{aligned} \tag{3.4}$$

Following the primal-dual formulation, I have:

$$\begin{aligned} \max_{\alpha\in\mathbb{R}^{n_l}} \quad & \sum_{i=1}^{n_l}\alpha_i - \frac{1}{2}\alpha^T\mathbf{Q}\alpha\,, \\ s.t. \quad & 0 \leq \alpha_i \leq \gamma_3,\ i=1,2,...,n_l\,, \end{aligned} \tag{3.5}$$

where $\mathbf{Q} = \mathbf{Y}^T\mathbf{J}^T\mathbf{X}^T(2\gamma_1\mathbf{I}+2\gamma_2\mathbf{X}\mathbf{L}\mathbf{X}^T)^{-1}\mathbf{X}\mathbf{J}\mathbf{Y}$, $\mathbf{J} = [\mathbf{I}\ \mathbf{0}]^T$ is a $n\times n_l$ matrix with $\mathbf{I}$ as the $n_l\times n_l$ identity matrix, $\mathbf{Y} = diag(y_1,...,y_{n_l}) \in \mathbb{R}^{n_l\times n_l}$ and $\alpha = [\alpha_1,...,\alpha_{n_l}]^T \in \mathbb{R}^{n_l}$ are Lagrangian multipliers.

This problem is a typical quadratic optimization problem which can be solved by standard optimization software. After α is obtained, $\mathbf{w}_0$ can be obtained by Equation 3.6. For more details, please refer to [Belkin et al., 2006].

$$\mathbf{w}_0 = (2\gamma_1\mathbf{I}+2\gamma_2\mathbf{X}\mathbf{L}\mathbf{X}^T)^{-1}\mathbf{X}\mathbf{J}\mathbf{Y}\alpha\,. \tag{3.6}$$

3.2.2 Detector Regularized Trackers

As mentioned before, for each object, an individual tracker is maintained. However, they are objects of the same type, which is confirmed by the detector. From this perspective, the GMOT problem can be naturally formulated within the MTL framework. All the tasks in MTL are related, while all the objects are treated as the same type in the detection stage. All the tasks in MTL are different from each other, while objects are treated differently when they are being tracked. Figure 3.2 illustrates how the MOT and MTL problems are inherently linked.

Based on the above inspiration, detection and tracking of multiple objects are considered as two main tasks, and tracking of each object as a sub-task within the MTL framework. The tracker for object t is denoted as $f_t(\mathbf{x}) = \mathbf{w}_t^T\mathbf{x}$. To relate the two main tasks, the deviation of each tracker from the detector $\mathbf{w}_0$ is penalized using the cost function as the following,

$$\sum_{t=1}^{T} \|\mathbf{w}_t - \mathbf{w}_0\|^2 . \tag{3.7}$$

This regularization term benefits the trackers in two aspects. Firstly, as $\|\mathbf{w}_t\|^2 = \|\mathbf{w}_t - \mathbf{w}_0 + \mathbf{w}_0\|^2 \leq \|\mathbf{w}_t - \mathbf{w}_0\|^2 + \|\mathbf{w}_0\|^2$, and $\|\mathbf{w}_0\|^2$ has been minimized in the detection stage, thus minimizing $\|\mathbf{w}_t - \mathbf{w}_0\|^2$ equals minimizing $\|\mathbf{w}_t\|^2$, further improving the generalization ability of each tracker. Secondly, this term can prevent trackers from drifting to the background as each tracker is forced to be close to the detector.

Furthermore, relatedness of multiple sub-tasks is encoded by learning the features jointly shared by all the trackers via a regularization term $\|\mathbf{W}\|_{2,1}$, where $\mathbf{W} \in \mathbb{R}^{d\times T}$ is the matrix composed of all the trackers as $[\mathbf{w}_1, \cdots, \mathbf{w}_T]$. $\|\mathbf{W}\|_{2,1}$ is the $\ell_{2,1}$ norm of $\mathbf{W}$ which first computes the ℓ_2 norm of each row to obtain a column vector, then computes the ℓ_1 norm of the column vector. This regularization term can result in that only some rows of $\mathbf{W}$ are non-zero, which correspond to the features shared by all sub-tasks. Figure 3.3 shows this clearly.

In addition, a smoothness term is introduced for each tracker. The smoothness term enables the tracker to view the labeled and unlabeled samples (candidates) together. It has been applied

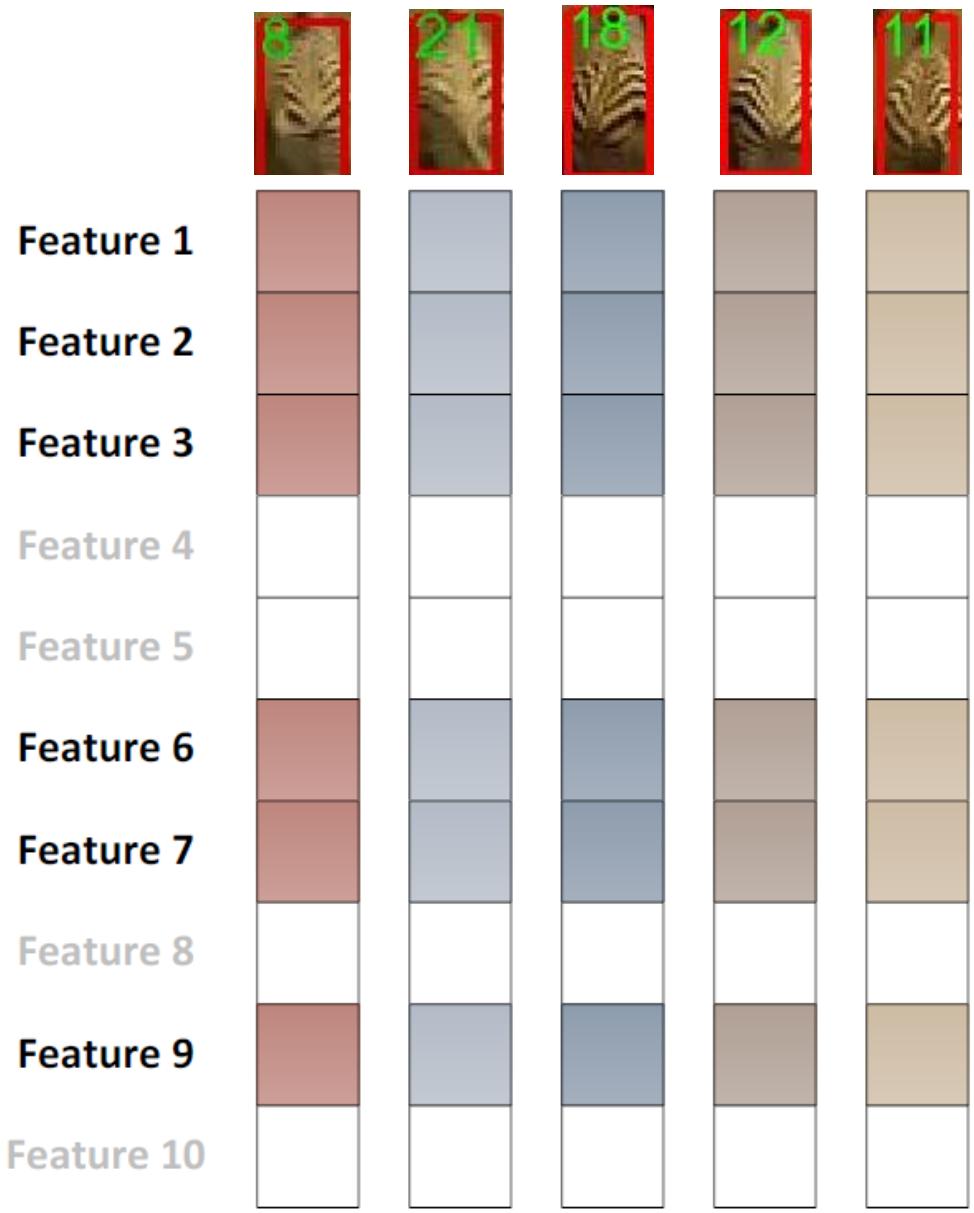

Figure 3.3: *Features are learned jointly.*

to the MTL framework by Luo et al. [Luo et al., 2013] to handle semi-supervised learning. Here it is introduced to gain the smoothness property of all the trackers.

The nearby instances around the current estimation (in the current frame) are sampled as

positive data, and the farther instances away from the current estimation (in the current frame) are sampled as negative data. The surrounding samples in the next frame are taken as unlabeled data. Having obtained training data, the Mean Regularized Joint Feature Learning algorithm which minimizes the following objective function is proposed:

$$\min_{\mathbf{W}\in\mathbb{R}^{d\times T}} \frac{1}{2}\sum_{t=1}^{T}\|\mathbf{J}_t^T\mathbf{X}_t^T\mathbf{w}_t-\mathbf{Y}_t\|^2+\rho_1\|\mathbf{W}\|_{2,1}+\frac{\rho_2}{2}\sum_{t=1}^{T}\|\mathbf{w}_t-\mathbf{w}_0\|^2+\frac{\rho_3}{2}\sum_{t=1}^{T}\mathbf{w}_t^T\mathbf{X}_t\mathbf{L}_t\mathbf{X}_t^T\mathbf{w}_t\,, \tag{3.8}$$

where $\mathbf{X}_t \in \mathbb{R}^{d\times(n_t^l+n_t^u)}$ is the combination of n_t^l labeled samples and n_t^u unlabeled samples for a sub-task t, $\mathbf{J}_t = \begin{bmatrix} \mathbf{I} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} \end{bmatrix}$ is a $(n_t^l+n_t^u)\times(n_t^l+n_t^u)$ matrix with $\mathbf{I}$ as the $n_t^l \times n_t^l$ identity matrix. $\mathbf{Y}_t \in \mathbb{R}^{(n_t^l+n_t^u)}$ is the label vector of task t (0 is given to the unlabeled data as neutral label). $\mathbf{L}_t$ is the Laplacian matrix associated with the graph of task t, and ρ_1, ρ_2, ρ_3 are the trade-off parameters. The above objective function captures the relatedness of the multiple tasks from two perspectives. One lies in the feature level, which makes the tasks share a common set of features. The other one lies in the classifier level, which encodes that all of the trackers should not be too different from the detector.

Solving Equation 3.8. The Accelerated Gradient Method (AGM) [Nesterov, 2007] is adopted to solve this composite optimization problem. Compared to the traditional gradient method, the AGM has the convergence speed of $\mathcal{O}(\frac{1}{k^2})$ (i.e. it achieves the solution with $\mathcal{O}(\frac{1}{k^2})$ residual from the optimal solution after k iterations), which is the optimal among the first order methods. For the sake of convenience, Equation 3.8 is written as a combination of a smooth component $\mathcal{L}(\mathbf{W}) = \frac{1}{2}\sum_{t=1}^{T}\|\mathbf{J}_t^T\mathbf{X}_t^T\mathbf{w}_t-\mathbf{Y}_t\|^2+\frac{\rho_2}{2}\sum_{t=1}^{T}\|\mathbf{w}_t-\mathbf{w}_0\|^2+\frac{\rho_3}{2}\sum_{t=1}^{T}\mathbf{w}_t^T\mathbf{X}_t\mathbf{L}_t\mathbf{X}_t^T\mathbf{w}_t$ and a non-smooth component $\Omega(\mathbf{W}) = \rho_1\|\mathbf{W}\|_{2,1}$. The AGM here iterates by using a linear combination of previous two points as the search point, rather than the latest point in the traditional gradient method. Each AGM iteration is composed of two steps: (1) Generalized Gradient Mapping which updates $\mathbf{W}^{(k+1)}$ given the search point $\mathbf{W}_S^{(k)}$, (2) Updating the current search point $\mathbf{W}_S^{(k)}$ by combining the previous two points.

(1) Generalized Gradient Mapping: given the current search point $\mathbf{W}_S^{(k)}$, the estimation

Algorithm 1: Mean Regularized Joint Feature Learning for MOT

Data: $\mathbf{X}_t, \mathbf{Y}_t, \mathbf{w}_0, t = 1, \cdots, T$.
Result: $\mathbf{W}$.
Initialization: each column of $\mathbf{W}^{(0)}$ and $\mathbf{W}^{(1)}$ is $\mathbf{X}_t * \mathbf{Y}_t$, $t^{(0)} = 0$, $t^{(1)} = 1$, $k = 1$,$\alpha = (t^{(0)} - 1)/t^{(1)}$, $\mathbf{W}_S^{(1)} = (1+\alpha)\mathbf{W}^{(1)} - \alpha\mathbf{W}^{(0)}$.
while *not converged* **do**
  Obtain $\mathbf{U} = \mathbf{W}_S^{(k)} - \frac{1}{\gamma}\nabla\mathcal{L}(\mathbf{W}_S^{(k)})$,
  Solve Equation 3.11 via Equation 3.12 to acquire $\mathbf{W}_i^{(k+1)}$, $i = 1, \cdots, d$,
  Update the search point as $\mathbf{W}_S^{(k+1)} = (1+\alpha)\mathbf{W}^{(k+1)} - \alpha\mathbf{W}^{(k)}$,
  $k \leftarrow k+1$, $t^{(k+1)} \leftarrow \frac{1}{2}(1 + \sqrt{1 + 4(t^{(k)})^2})$, $\alpha \leftarrow (t^{(k)} - 1)/t^{(k+1)}$.

$\mathbf{W}^{(k+1)}$ can be obtained by solving Equation 3.9

$$\mathbf{W}^{(k+1)} = \arg\min_{\mathbf{W}} \frac{\gamma}{2}\|\mathbf{W} - (\mathbf{W}_S^{(k)} - \frac{1}{\gamma}\nabla\mathcal{L}(\mathbf{W}_S^{(k)}))\|_F^2 + \Omega(\mathbf{W}), \tag{3.9}$$

where γ is a step parameter and $\nabla\mathcal{L}(\mathbf{W})$ is the gradient of $\mathcal{L}(\mathbf{W})$. Each column of $\nabla\mathcal{L}(\mathbf{W})$ is:

$$\mathbf{X}_t\mathbf{J}_t(\mathbf{J}_t^T\mathbf{X}_t^T\mathbf{w}_t - \mathbf{Y}_t) + \rho_2(\mathbf{w}_t - \mathbf{w}_0) + \rho_3\mathbf{X}_t\mathbf{L}_t\mathbf{X}_t^T\mathbf{w}_t, \quad t = 1, \cdots, T. \tag{3.10}$$

Considering the computation procedure of $\ell_{2,1}$ norm, Equation 3.9 can be decoupled as d disjoint sub-problems in Equation 3.11 (one for each row vector $\mathbf{W}_i$),

$$\mathbf{W}_i^{(k+1)} = \arg\min_{\mathbf{W}_i} \frac{1}{2}\|\mathbf{W}_i - \mathbf{U}_i\|_2^2 + \lambda\|\mathbf{W}_i\|_2, \quad i = 1, \cdots, d, \tag{3.11}$$

where $\mathbf{U} = \mathbf{W}_S^{(k)} - \frac{1}{\gamma}\nabla\mathcal{L}(\mathbf{W}_S^{(k)})$, $\mathbf{U}_i$ is the i-th row of $\mathbf{U}$ and $\lambda = \rho_1/\gamma$. Following [Chen et al., 2009, Zhang et al., 2012], the solution to Equation 3.11 is:

$$\mathbf{W}_i^{(k+1)} = \max(1 - \frac{\lambda}{\|\mathbf{U}_i\|_2}, 0)\mathbf{U}_i, \quad i = 1, \cdots, d. \tag{3.12}$$

(2) Updating the current search point as a linear combination of the previous two points:

Table 3.1: *Quantitative results compared with the extended TLD (eTLD) and modified MST (mMST). In the metrics with the upward arrow, the greater number indicates the better performance (and vice versa for the downward arrow). For each data sequence, the last row shows results of the proposed method. The best performance values are in bold.*

Sequence	MOTA ↑	MOTP ↑	Rec. ↑	Prec. ↑
Zebra (eTLD)	52.93±6.46	66.75±1.96	54.45±6.61	94.58±2.64
Zebra (mMST)	73.10±4.11	63.55±2.35	**79.16±3.01**	91.87±1.93
Zebra (GMOT-MTL)	**75.30±2.84**	**66.80±0.10**	77.81±3.50	**97.20±0.84**
Antelope (eTLD)	8.89±3.10	**68.03±4.08**	24.11±5.35	61.72±9.51
Antelope (mMST)	18.92±0.45	59.90±0.07	61.37±1.12	59.22±0.26
Antelope (GMOT-MTL)	**30.10±12.23**	62.49±4.29	**69.78±6.45**	**63.84±5.63**

$$\mathbf{W}_S^{(k+1)} = (1+\alpha)\mathbf{W}^{(k+1)} - \alpha\mathbf{W}^{(k)}\,, \tag{3.13}$$

where $\alpha = (t^{(k)}-1)/t^{(k+1)}$ and $t^{(k+1)} = \frac{1}{2}(1+\sqrt{1+4(t^{(k)})^2})$. The algorithm terminates when the change of the function is lower than a threshold or the number of iterations has achieved the maximum. The Mean Regularized Joint Feature Learning algorithm is summarized in Algorithm 1. Note that this algorithm is implemented based on the code from the MALSAR package [Zhou et al., 2011c].

After the solution $\mathbf{W}$ is obtained, each column $\mathbf{w}_t$ is the tracker for each sub-task (each object). For tracking, the most confident candidate is selected as the estimation of each object, i.e.,

$$\mathbf{x}_t^* = \arg\max_{\mathbf{x}\in\mathbf{X}_t^u} \mathbf{w}_t^T\mathbf{x}\,, \tag{3.14}$$

where $\mathbf{X}_t^u$ is the unlabeled part of $\mathbf{X}_t$.

Table 3.2: *Quantitative results compared with the extended TLD (eTLD) and modified MST (mMST). In the metrics with the upward arrow, the greater number indicates the better performance (and vice versa for the downward arrow). For each data sequence, the last row shows results of the proposed method. The best performance values are in bold.*

Sequence	MT ↑	ML ↓	FM ↓	IDS ↓
Zebra (eTLD)	18.84±3.84	43.48±2.51	110.00±9.54	40.27±3.65
Zebra (mMST)	38.04±9.56	35.87± 4.64	59.25±16.68	22.45±4.79
Zebra (GMOT-MTL)	**42.03±5.23**	**34.30±4.43**	**28.33±0.58**	**5.94±0.88**
Antelope (eTLD)	1.47±2.55	77.45±5.94	117.67±42.52	70.15±30.43
Antelope (mMST)	19.12±6.24	51.47±2.08	93.00±2.83	60.07±2.15
Antelope (GMOT-MTL)	**32.84±5.57**	**40.68±8.10**	**91.00±28.83**	**52.81±10.87**

Table 3.3: *Quantitative results compared with two baselines (BL1, BL2). In the metrics with the upward arrow, the greater number indicates the better performance (and vice versa for the downward arrow). For each data sequence, the last row shows results of the proposed method. The best performance values are in bold.*

Sequence	MOTA ↑	MOTP ↑	Rec. ↑	Prec. ↑	MT ↑	ML ↓	FM ↓	IDS ↓
Zebra (BL1)	72.31	58.23	78.58	93.10	30.43	34.78	36	11
Zebra (BL2)	69.40	**67.12**	75.30	93.08	33.33	34.78	34	10
Zebra (GMOT-MTL)	**77.69**	66.78	**80.30**	**97.02**	**43.48**	**30.43**	**29**	**7**
Crab (BL1)	26.95	59.39	47.63	69.24	8.74	76.70	**275**	**78**
Crab (BL2)	32.70	59.40	45.25	77.89	**9.71**	74.76	284	84
Crab (GMOT-MTL)	**39.06**	**60.00**	**51.50**	**80.65**	**9.71**	**70.87**	306	92
Antelope (BL1)	24.46	**67.29**	65.31	61.75	35.29	39.71	**59**	**23**
Antelope (BL2)	23.62	66.73	65.55	61.27	**38.24**	38.24	64	27
Antelope (GMOT-MTL)	**35.58**	63.31	**73.97**	**65.81**	36.76	**36.76**	96	30

3.3 Experiments

3.3.1 Feature

The HOG [Dalal and Triggs, 2005], LBP [Wang et al., 2009b] and the Color Histogram are computed as features and these feature vectors are concatenated to represent a window. The joint feature learning in the proposed algorithm will select the useful features for MOT.

Table 3.4: *Quantitative results compared with other MOT approaches. In the metrics with the upward arrow, the greater number indicates the better performance (and vice versa for the downward arrow). The last row shows results of the proposed method. The best performance values are in bold. Note that [Brendel et al., 2011, Breitenstein et al., 2009, Okuma et al., 2004] do not supply the MT and ML results.*

Sequence	MOTA ↑	MOTP ↑	Rec. ↑	Prec. ↑	MT ↑	ML ↓
UBC Hockey (eTLD)	54.66	64.66	65.04	84.25	17.86	25.00
UBC Hockey [Brendel et al., 2011]	79.7	60.0	80.5	98.9	-	-
UBC Hockey [Breitenstein et al., 2009]	76.5	57.0	77.7	98.8	-	-
UBC Hockey [Okuma et al., 2004]	67.8	51.0	68.7	**100**	-	-
UBC Hockey (GMOT-MTL)	**80.30**	**69.09**	**92.37**	89.20	**67.86**	**10.71**

3.3.2 Tracking Management

At runtime, a list is maintained to save the objects. If the detector discovers a new object, it will be assigned a weight and it will be buffered. Then its weight increases when it is detected again. In contrast, the weight decrease if it is not detected. If the weight is greater than a threshold τ, a tracker is initialized for it. For the objects existed in the list, an opposite process is operated to delete objects when they disappear from the image scene.

3.3.3 Parameters

Here the setting of some parameters is noted. For the Color Contrast cue, the default parameters is used as in [Alexe et al., 2010] except θ. This parameter is empirically set as 60 as the result of shortage of training examples. It works well on the experimental data sets. When constructing graph, the *10-NN* and the *rbf* kernel are employed to calculate the adjacency matrix.

3.3.4 Data Sets & Evaluation Metrics

The proposed approach (termed as GMOT-MTL) is tested on four challenging data sets named Zebra, Crab, Antelope and UBC Hockey [Okuma et al., 2004] respectively. There are scale

changes in the Zebra sequence and there are background clutter, scale variation and rotation in the Crab sequence. For the Antelope and UBC Hockey sequences, there exist severe occlusions and out-of-plane rotation.

To evaluate the tracking performance quantitatively, the Multiple Object Tracking Accuracy (MOTA), Multiple Object Tracking Precision (MOTP), Recall, Precision, the number of Mostly Tracked (MT) and Mostly Lost (ML) trajectories [Li et al., 2009], FM and IDS metrics are computed.

3.3.5 Results

The experimental results are reported as four parts. The first part compares the proposed method with two third-party methods. The second part reports the comparison between the proposed method and two based line methods originated from the proposed one. The third part tests the effect from different numbers of initial bounding boxes. The fourth part tests the proposed approach on a data set of human (UBC Hockey).

Part 1. In the first part, the proposed approach is compared with two different methods on data sets of Zebra and Antelope. The first one is called extended TLD (eTLD). The TLD framework [Kalal et al., 2012] is extended for generic MOT. The detector of the TLD framework is based on random ferns, which can detect non-specific type objects. The extended TLD (eTLD) selects the detected similar objects to track. The second one is modified MST (mMST) [Zhao et al., 2012]. I borrow the problem setting from this paper. To be specific, an object indicated by a given bounding box is tracked through 10 frames and training data is collected based on the tracking results. According to the training data, an initial detector is trained and applied at the first frame to detect objects. Then the following procedure is the same as the proposed approach. The comparison is based on 15 times of running each method (initialized from different bounding boxes). Quantitative results are shown in Table 3.1 and 3.2.

The figures in Table 3.1 and 3.2 reveal that the extended TLD performs slightly worse than the proposed approach on the Zebra sequence, but on the other sequence its results are much

Figure 3.4: *Images excerpted from Zebra sequence. The number attached to each bounding box is the object's ID and the yellow line is its estimated trajectory. This figure is best viewed in color.*

Figure 3.5: *Images excerpted from the Crab sequence. The number attached to each bounding box is the object's ID and the yellow line is its estimated trajectory. This figure is best viewed in color.*

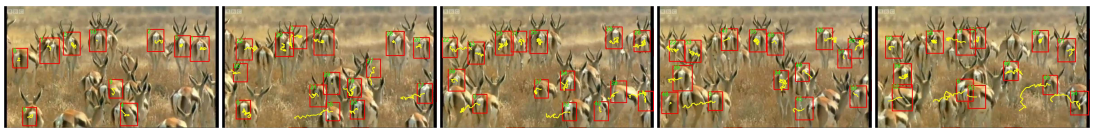

Figure 3.6: *Images excerpted from the Antelope sequence. The number attached to each bounding box is the object's ID and the yellow line is its estimated trajectory. This figure is best viewed in color.*

worse than the proposed one. That is because the antelopes do not have evident patterns like the zebras and the backgrounds of this sequence are cluttered (see Figure 3.4 and Figure 3.6). The mMST approach generally works slightly worse than GMOT-MTL on these two sequences.

Part 2. To verify the improvement from the joint feature learning term and the smoothness term, another two baselines are developed to be compared with. The first one (BL1) is formed by only keeping the fitting error term and the mean-regularized term. The second one (BL2) is formed by that the jointly feature learning term is incrementally added to BL1 (still without the smoothness term). Table 3.3 shows the quantitative results. Note that, the experiment is conducted based on the same initial bounding box. Figures from 3.4 to 3.6 show the qualitative results of these three data sets. It is easy to observe the improvement from the jointly feature learning term and the smoothness term if results of GMOT-MTL are compared with those of the two baselines BL1 and BL2.

Part 3. This part tests the effect to the final results from different numbers of initial bounding boxes. Specifically, the proposed method is initialized by different numbers of bounding boxes (ranging from 1 to 7 in the experiment) on the Zebra sequence, and the results from

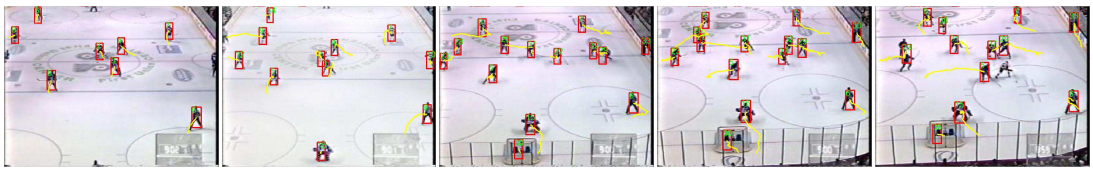

Figure 3.7: *Images excerpted from the UBC Hockey sequence. The number attached to each bounding box is the object's ID and the yellow line is its estimated trajectory. This figure is best viewed in color.*

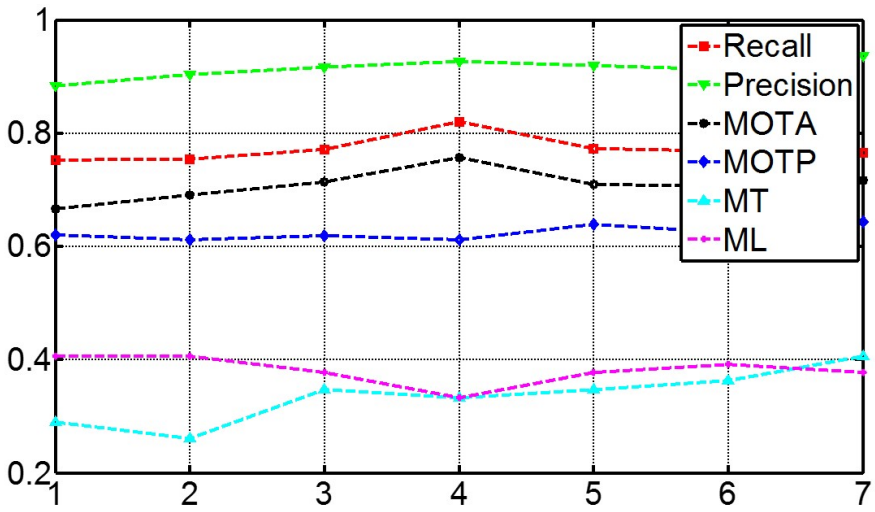

Figure 3.8: *Performance of the proposed method initialized by different numbers of initial bounding boxes on the Zebra sequence.*

different initial settings are compared. The results are shown in Figure 3.8 and 3.9. Generally, the results suggests that more initial bounding boxes would lead to slightly better performance.

Part 4. To further illustrate that the proposed approach is effective, I also test it on a public data set named UBC Hockey [Okuma et al., 2004]. The results (shown in Table 3.4) are compared with some MOT approaches [Brendel et al., 2011, Breitenstein et al., 2009, Okuma et al., 2004]. The purpose of comparison with other MOT approaches is to certify that the proposed approach can also work well on some human data sets even if it is not given an elaborate human detector.

Note that, the computation speed is about 1 frame per second on a desktop with Intel i7-965

CPU, 8G ram and unoptimized Matlab code. The optimization stage costs most of the time.

3.4 Remarks

In this chapter I have shown how generic object crowd tracking is formulated into the multiple task learning framework and have proposed the novel methods. I have decomposed the problem into two main tasks and represented their relation by the proposed Mean Regularized Joint Feature Learning algorithm. The optimization functions of these two main tasks include the terms for the generalization ability, the smoothness, the fitting errors and feature learning. Solving the optimization problems yields the desired list of detected and tracked objects in frames. Experimental results on the challenging data sequences have confirmed the efficacy of the proposed approach over the state-of-the-art ones.

It is observed that, tracking performance relies on detection performance to some extent. Better detection results mean smaller false positive and false negative values, which consequently result in better tracking performance such as greater MOTA values. Thus, seeking a

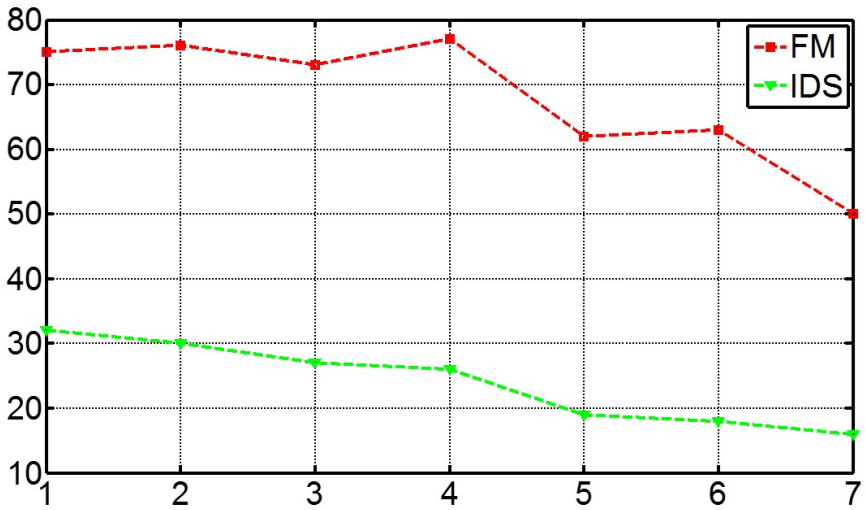

Figure 3.9: *Performance of the proposed method initialized by different numbers of initial bounding boxes on the Zebra sequence.*

better solution to online generic object detection is demanded in next chapter.

CHAPTER 4

BI-LABEL PROPAGATION FOR GENERIC MOT

As discussed in previous chapters, the generic multiple object tracking problem is difficult due to the lack of a reliable object detector for objects of a generic type (as a basis, this is especially important), frequent occlusions and appearance similarity among objects. Owing to advances in object detection (especially in pedestrian detection [Dalal and Triggs, 2005, Felzenszwalb et al., 2010]), the task of traditional multiple object tracking can be solved efficiently using a tracking-as-detection approach. However, generalizing the task to other objects (see the data sets in Section 4.4) would require training a detector for each new object type, which is impractical. In this chapter the same problem is dealt, i.e. tracking multiple objects of the same generic type given only one initial bounding box [Luo and Kim, 2013], and the task remains the same, i.e. recovering multiple trajectories from image observations.

Treating a concerned video as a spatio-temporal cuboid, sliding windows as unlabeled points and the initial bounding box as a single labeled point in this cuboid, I aim to discover and track new objects by propagating labels to similar candidates. From this perspective, it

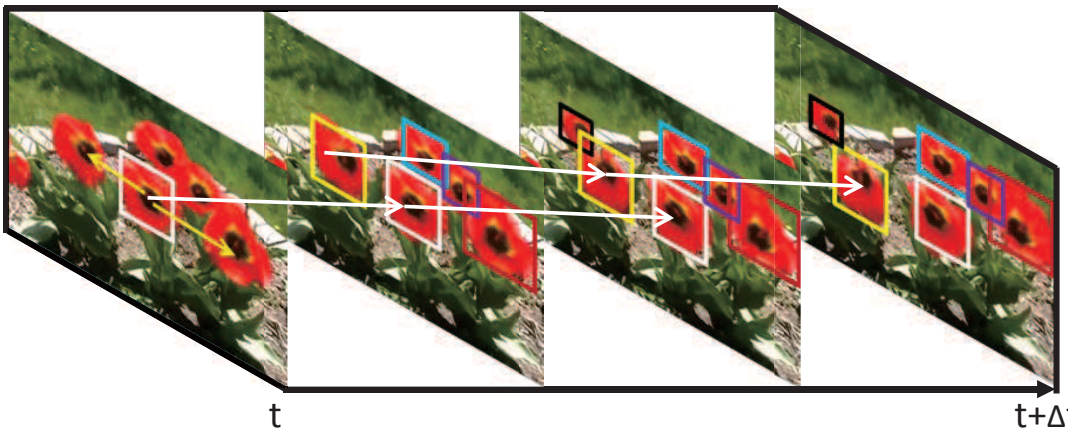

Figure 4.1: *The proposed framework. Yellow arrows indicate the propagation of class labels within the same frame and white arrows indicate object label propagation over time (best viewed in color).*

shares great similarity with semantic video segmentation [Badrinarayanan et al., 2010] which aims to label all the pixels in a video given pixel labels in the first frame. However, these two problems have significant differences: labels in video segmentation involve only a fixed number of pre-defined classes, whereas labels in GMOT involve both binary classes (object vs. background) and multiple classes (specific object identities). Thus the number of classes in GMOT problem varies as objects appear or disappear. Also, in video segmentation more than one pixel can share the same label while in GMOT object labels are exclusive.

For each sample, the labels are treated as a combination of *binary class labels* and *object labels* (identities), and detection responses are referred as an intermediate layer between image observations and trajectory estimations. Furthermore, a sequential label propagation framework (Figure 4.1) is developed to propagate class labels and object labels in both spatial and temporal domains. This so-called *bi-label propagation* framework coincides with a tracking-by-detection strategy: through spatially propagating the class labels (yellow arrows in Figure 4.1), the detection problem is solved, discovering the appearance and disappearance of objects; by temporally propagating object labels (white arrows in Figure 4.1), the multi-object tracking problem is tackled.

Learning a robust detector from a single training instance is challenging and standard meth-

ods tend to either overfit (e.g. using a kernel SVM) or underfit (e.g. using a linear SVM). To address the generalization issue, multiple detectors inspired by ensemble learning are trained. Multiple detectors are inherently related to each other since they are dealing with the same type of objects. The motivation of Multiple Task Learning (MTL) [Evgeniou and Pontil, 2004] is to learn multiple related tasks simultaneously rather than independently. Thus, training each of the detector is considered as one task and clustered MTL (cMTL) [Zhou et al., 2011b] is adopted to regularize the training process of multiple detectors. In addition, images and hence detection results are assumed to do not change drastically from frame to frame. This spatio-temporal consistency is modeled by integrating it into the cMTL formula during the class label propagation.

The key contributions in this chapter are (1) a probabilistic framework is proposed for jointly propagating class and object labels in spatial and temporal domains for GMOT and (2) cMTL is introduced for generic object detection and it is improved by considering the spatio-temporal consistency.

Methods for generic object detection in video data require either pre-trained detectors [Yang et al., 2013] or off-line training [Ali et al., 2011]. Models are adapted to a given input video in order to improve the detection accuracy, e.g. by iterative boosting [Ali et al., 2011]. The closest work is coupled detection and tracking [Leibe et al., 2008, Wu et al., 2012]. However, most work assumes that a detector trained offline is available. For example, in [Wu et al., 2012], Wu et al. use a dictionary of foreground images for pedestrian detection together with background subtraction. The work in [Leibe et al., 2008] employs off-line trained pedestrian and car detectors. In terms of problem setting, I follow the model-free approaches in [Luo and Kim, 2013, Zhang and van der Maaten, 2013b]. Zhang and van der Maaten require initialization with bounding boxes of all objects. In contrast to the proposed method it does not discover new similar objects [Zhang and van der Maaten, 2013b]. Luo and Kim first train a generic object detector, and subsequently employ the detector to regularize the training of multiple trackers [Luo and Kim, 2013]. In contrast to this approach, I learn detection with the help of tracking, i.e. the spatio-temporal consistency, as well as tracking based on detection, in a joint optimization framework.

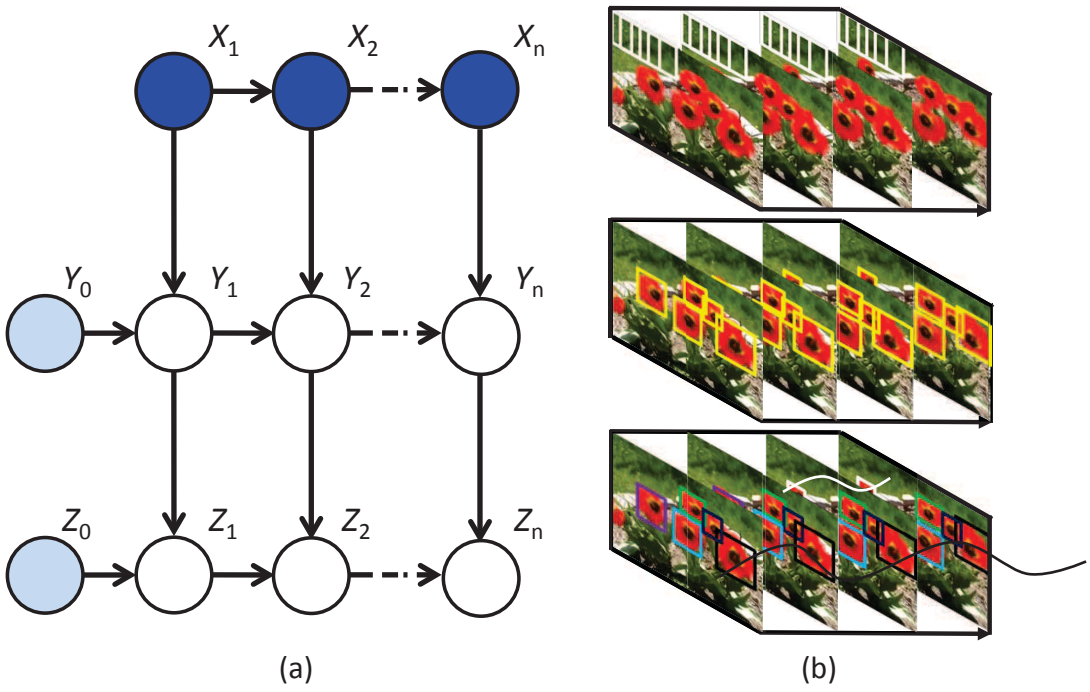

Figure 4.2: *(a) Graphical model of the proposed approach. (b) Top to bottom: sliding windows X, detection responses Y, and trajectories Z. For sake of display, only two trajectories are shown (best viewed in color).*

4.1 Bayesian Perspective

Let X, Y and Z represent sliding windows (image observations), detection responses and trajectories, respectively. Figure 4.2(a) shows graphical model which has three layers: image observation, detection, and trajectory layer, respectively. The darkly shaded nodes are observed nodes, the transparent nodes are hidden (or latent) nodes, and the lightly shaded nodes (Y_0 and Z_0) are partly observed as only a single initial bounding box is given in the first frame. From the image layer to the detection response layer class labels are propagated. From the detection response layer to the trajectory layer object labels are propagated.

Solving the problem corresponds to maximizing $P(Z|X)$. Introducing variable Y could

Figure 4.3: *Illustration of intra-class variance. Shown are cropped regions from (a) the* Airshow *sequence, (b) the* Goose *sequence and (c) the* Hockey *sequence.*

obtain

$$\begin{aligned} \max_{Z} P(Z|X) &\propto \max_{Z,Y} P(Z|X,Y)P(Y|X) \\ &= \max_{Z,Y} \prod_{t} P(Z_t|X_t, Y_t, Z_{0:t-1})P(Y_t|X_t, Y_{t-1}) \,, \end{aligned} \tag{4.1}$$

where $P(Y|X)$ models class label propagation (detection) and $P(Z|X,Y)$ models object label propagation (tracking). It can be expanded sequentially as

$$\max_{Z_t,Y_t} P(Z_t|X_t, Y_t, Z_{0:t-1})P(Y_t|X_t, Y_{t-1}) \,, \tag{4.2}$$

and this estimation problem can be solved by decomposition. Taking the negative logarithm of Equation 4.2, it can be rewritten as:

$$\min_{\mathbf{W}_t,\Theta_t} \mathcal{L}_C(\mathbf{W}_t) + \mathcal{L}_O(\Theta_t) \,, \tag{4.3}$$

where $\mathcal{L}_C(\mathbf{W}_t)$ models class label propagation, $\mathcal{L}_O(\Theta_t)$ models object label propagation and $\mathbf{W}_t, \Theta_t$ are parameters representing the detector and propagation configuration at time t. To minimize the function, the following procedure is operated.

(1) fix Θ_{t-1} to minimize $\mathcal{L}_C$ via $\mathbf{W}_t$;

(2) fix $\mathbf{W}_t$, minimize $\mathcal{L}_O$ via Θ_t;

(3) $t \leftarrow t+1$ (go to the next frame).

4.2 Class Label Propagation

Let us review the Bayesian formula of class label propagation $P(Y_t|X_t, Y_{t-1})$ in Equation 4.2. The objective is to maximize the likelihood of Y_t conditioned on observations X_t (spatial domain) and the previous estimation Y_{t-1} (temporal domain).

The detection problem in GMOT differs from the traditional detection problem as there are not sufficient data to handle large intra-class variation. Figure 4.3 illustrates the extent of intra-class variation in three test videos. This figure reveals that, even samples belong to the same type of objects, the variance among them is considerably large.

As training a single classifier leads to underfitting or overfitting, multiple detectors are trained and a decision is made based on all of them. Moreover, by treating training each detector as one task, the relationship among multiple detectors is investigated and cMTL is adopted to train these detectors simultaneously, improving the generalization ability.

In the first frame, small perturbations are added to the initial bounding box (slight shift, rotation, scale changes) to augment the positive data. To be specific, sliding windows whose overlap (intersection/union) with the initial bounding box greater than 0.7 are selected as positive data. Sliding windows whose overlap with the initial bounding box between 0.2 and 0.3 are negative samples. In the following frames, the training data is augmented in the same way while the only difference is that the training data are augmented based on the detected bounding boxes.

By randomly sampling a subset of instances from the whole training data without placement m times, m sets of training data $\mathbf{X}_{l,t,i} \in \mathbb{R}^{d \times N_{t,i}}, i = 1, ..., m$ are obtained and their labels are $\mathbf{Y}_{t,i} \in \{1, -1\}^{N_{t,i}}$, where the subscript "l" means "labeled", d is the feature space dimension and $N_{t,i}$ is the number of instances. Let the multiple detectors be $\mathbf{W} = [\mathbf{w}_1, ..., \mathbf{w}_m] \in \mathbb{R}^{d \times m}$. Using the least square error the data cost term is $\sum_{i=1}^{m} \|\mathbf{X}_{l,t,i}^T \mathbf{w}_{t,i} - \mathbf{Y}_{t,i}\|^2$. The detectors are related as they are dealing with objects of the same type. Meanwhile, as a result of data distribution, a cluster of instances are more similar to each other compared with others, e.g. some

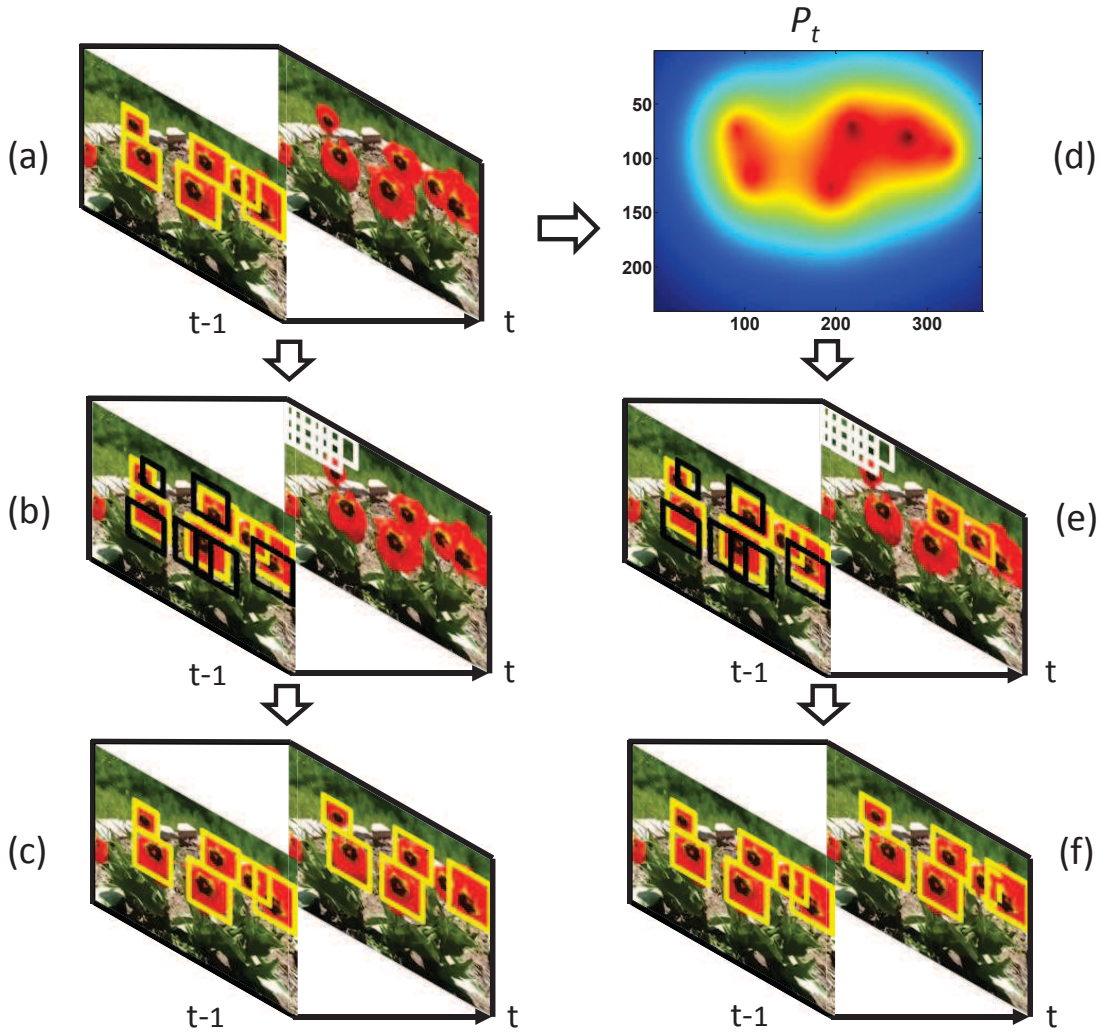

Figure 4.4: *Illustration of how the spatio-temporal consistency guides the detection procedure (best viewed in color).*

instances exhibit a similar viewpoint while some do not. Consequently, some detectors will be closer to each other in the model parameter space. Therefore the detectors are assumed to form k clusters as $C_j, j = 1, ..., k$, and the coupling among all detectors is modeled following [Zhou et al., 2011b]:

$$\sum_{j=1}^{k} \sum_{v \in C_j} \| \mathbf{w}_v - \bar{\mathbf{w}}_j \|^2 = tr(\mathbf{W}^T \mathbf{W}) - tr(\mathbf{F}^T \mathbf{W}^T \mathbf{W} \mathbf{F}) , \tag{4.4}$$

where $\bar{\mathbf{w}}_j$ is the mean of the detectors within the same cluster, $tr(\bullet)$ is the trace norm, and $\mathbf{F} \in$

$\mathbb{R}^{m\times k}$ is an orthogonal cluster indicator matrix with $\mathbf{F}_{i,j} = \frac{1}{\sqrt{n_j}}$ if $i \in \mathcal{C}_j$ and $\mathbf{F}_{i,j} = 0$ otherwise. Along with regularization of each detector $\sum_{i=1}^{m} \parallel \mathbf{w}_i \parallel^2 = tr(\mathbf{W}^T\mathbf{W})$, a regularization term is $tr(\mathbf{W}((1+\eta)\mathbf{I} - \mathbf{F}\mathbf{F}^T)\mathbf{W}^T)$, where η is a weight parameter. Following the convex relaxation of cMTL [Zhou et al., 2011b], this regularization term is relaxed to $tr(\mathbf{W}(\eta\mathbf{I}+\mathbf{M})^{-1}\mathbf{W}^T)$, subject to $tr(\mathbf{M}) = k, \mathbf{M} \preceq \mathbf{I}, \mathbf{M} \in \mathbf{S}_+^m$, where $\mathbf{S}_+^m$ is the set of positive semi-definite (PSD) matrices and $\mathbf{M} \preceq \mathbf{I}$ means $\mathbf{I} - \mathbf{M}$ is PSD.

Traditional MOT applies a detector to every frame independently. By contrast, detection responses in two subsequent frames should not change drastically. To utilize such information, confident instances are tracked via a weak tracker (KLT in the implementation) from frame $t-1$ to frame t, and produce a density map P_t (see an example in Figure 4.4(d)) by smoothing the confidence scores with a Gaussian ($\sigma = 5$). Based on P_t, sliding windows $\mathbf{X}_{u,t} \in \mathbb{R}^{d\times N}$ (here the subscript "u" means "unlabeled") can be weakly labeled as $\Psi(P_t)$ which is the summation of the density of pixels close to their centers (within a circle of radius 4). The cost term $\| \frac{1}{m}\sum_{i=1}^{m} \mathbf{X}_{u,t}^T\mathbf{w}_{t,i} - \Psi(P_t)\|^2$ can be considered as a weakly supervised term which propagates labels in the temporal domain. Intuitively, it assists the detector to recall more instances. Figure 4.4 shows this concept. Yellow boxes indicate the detection results (also positive instances), black boxes are negative instances, and white boxes are unlabeled samples. With the help of spatio-temporal consistency, some candidates have weak labels indicated by the orange boxes in frame t shown in Figure 4.4(e), and the weak labels help to recover missed detections (see the dashed yellow box in frame t in Figure 4.4(f) which is a missed detection caused by occlusion in Figure 4.4(c)). Based on the terms described above, there is

$$\begin{aligned}\mathcal{L}_C(\mathbf{W}_t) = &\underbrace{\alpha\, tr(\mathbf{W}_t(\eta\mathbf{I}+\mathbf{M}_t)^{-1}\mathbf{W}_t^T)}_{regularization} + \\ &\underbrace{\frac{\lambda}{2}\|\frac{1}{m}\sum_{i=1}^{m}\mathbf{X}_{u,t}^T\mathbf{w}_{t,i} - \Psi(P_t)\|^2}_{spatio-temporal\ consistency} + \underbrace{\sum_{i=1}^{m}\frac{1}{2N_{t,i}}\|\mathbf{X}_{l,t,i}^T\mathbf{w}_{t,i} - \mathbf{Y}_{t,i}\|^2}_{loss}, \\ &s.t.\ tr(\mathbf{M}_t) = k, \mathbf{M}_t \preceq \mathbf{I}, \mathbf{M}_t \in \mathbf{S}_+^m .\end{aligned} \tag{4.5}$$

This is a joint convex problem with regard to $\mathbf{W}$ and $\mathbf{M}$ [Argyriou et al., 2007]. Following [Zhou et al., 2011b], the Accelerated Project Gradient method is adopted to optimize this

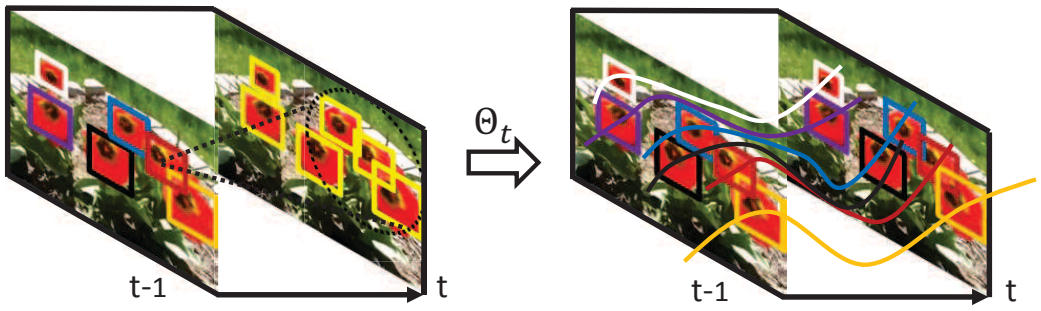

Figure 4.5: *Object labels are propagated from trajectories (different colors mean different objects) in frame $t-1$ to detection responses in frame t. Note the proximity of a flower indicated by the black dashed circle (best viewed in color).*

function. Labels of $\mathbf{X}_{u,t}$ are obtained by averaging the scores of all detectors as:

$$\mathbf{Y}_{u,t} = \frac{1}{m}\sum_{i=1}^{m}\mathbf{X}_{u,t}^{T}\mathbf{w}_{t,i}\,. \tag{4.6}$$

Candidates with a score greater than zero are chosen and non-maximum suppression is applied to output final class labels $\mathbf{Y}_{u,t} \in \{-1,1\}^{N}$.

4.3 Object Label Propagation

In the Bayesian formula Equation 4.2, object label propagation is $P(Z_t|X_t,Y_t,Z_{0:t-1})$, where the estimation of Z_t is conditioned on detection responses Y_t and the history of estimations $Z_{0:t-1}$. Let the n trajectories at time $t-1$ be

$$T = \{T_i | T_i =< T_i^A, T_i^M, T_i^C >, i = 1,...,n\}\,, \tag{4.7}$$

where T_i^A, T_i^M and T_i^C indicate appearance, motion and context information, and let the m detection responses at time t be

$$D = \{D_j | D_j =< D_j^A, D_j^L, D_j^C >, j = 1,...,m\}\,, \tag{4.8}$$

where D_j^A, D_j^L and D_j^C represent the appearance, location and context information. Tracking is carried out by propagating object labels from trajectories to detection responses via a configuration variable $\Theta_t \in \mathbb{R}^{n\times m}$. Initially, all the elements of Θ_t are 0. If an element Θ_{tij} is

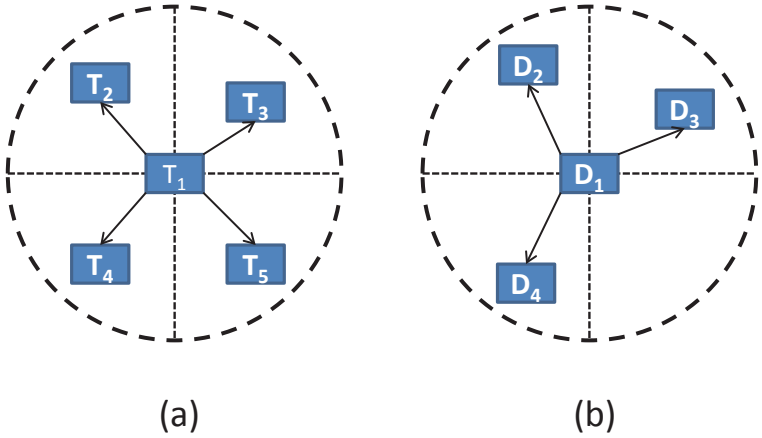

Figure 4.6: *Context model. Contexts of (a) trajectories and (b) detection responses are modeled by histograms, counting objects within an object's proximity.*

switched to 1, then the label of trajectory T_i is propagated to detection response D_j, and the propagated quantity depends on the affinity $S(T_i \rightarrow D_j)$ between T_i and D_j (here "$\rightarrow$" means considering D_j as a component of T_i at time t), which is determined by appearance, motion and context. Figure 4.5 shows this process. Objects are assumed to move smoothly, so only the detection responses within T_i's spatio-temporal proximity Ω_i (a circle with radius d_{Th}) are considered and the following energy function is minimized:

$$\mathcal{L}_O(\Theta_t) = -\sum_i \sum_{j \in \Omega_i} S(T_i \rightarrow D_j)\Theta_{tij}\,. \tag{4.9}$$

Appearance Model. Following [Adam et al., 2006], I save a set of templates to store historical information for appearance modeling. This is different from the solution in the previous chapter as it is online association in this chapter. The intensity cue is considered for appearance affinity computation. The appearance model T_i^A of trajectory T_i consists of the last 15 templates of this object, and the appearance similarity between D_j and T_i is

$$S_A(T_i \rightarrow D_j) = med(\text{NCC}(T_i^A, D_j^A))\,, \tag{4.10}$$

where $\text{NCC}(\bullet, \bullet)$ is the normalized cross-correlation (NCC) similarity measure and $med(\bullet)$ is the median.

Motion Model. The past three displacements are maintained and weighted by $[\frac{4}{7}, \frac{2}{7}, \frac{1}{7}]$ to predict a displacement vec_i, where older values are weighted higher. Given D_j, the actual displacement vec_j is the difference between D_j^L and the most recent location of the object corresponding to T_i. The motion affinity is

$$S_M(T_i \rightarrow D_j) = cos(vec_i, vec_j)\,. \tag{4.11}$$

Context Model. In modeling context information, I follow the work in [Reilly et al., 2010] and employ 2D histograms of nearby objects to improve the robustness. As shown in Figure 4.6, there are (a) five trajectories and (b) four detection responses. To compute a histogram for T_i, the neighborhood of T_i is divided into M partitions (here $M = 4$ for sake of display). For each object located in this neighborhood a distance vector is computed relative to T_i. According to the distance vector, the distance values are accumulated for each partition. By normalization, an M-bin histogram $\mathbf{h}_i$ is obtained. The context affinity is

$$S_C(T_i \rightarrow D_j) = exp(-Bhatt(\mathbf{h}_i, \mathbf{h}_j)). \tag{4.12}$$

Having obtained affinities based on three cues, they are combined as follows:

$$S(T_i \rightarrow D_j) = S_A(T_i \rightarrow D_j) * S_M(T_i \rightarrow D_j) * S_C(T_i \rightarrow D_j)\,. \tag{4.13}$$

The energy (Equation 4.9) is minimized by greedy search in an iterative way. First all propagation switches are turned off. Then the affinities of all propagation pairs are computed and the propagation switch (say T_i and D_j) which most decreases the energy is turned on. At the same time, D_j is labeled as the extension of T_i. This pair of trajectory and detection response is removed from the search space consequently. This procedure is repeated until there is no further energy decrease. Finally, trajectories outside the search space are updated considering the extended component. The remaining trajectories in the search space are terminated, and new trajectories are initialized based on detection responses in the search space. For clarity, the algorithm is summarized in Algorithm 2.

Algorithm 2: Object Label Propagation for MOT

Data: T, D, proximity set Ω.
Result: Θ_t, labels of detection responses.
Initialization: $\Theta_t = \mathbf{0}$.
while $\mathcal{L}_O$ *decrease,* **do**
  foreach $T_i \in T$ *and* $D_j \in \Omega_i$, **do**
    compute the energy decrease of T_i and D_j.
  find T_i and D_j with the greatest decrease via Equation 4.9
  set $\Theta_{tij} = 1$, propagate the label of T_i to D_j.
  remove T_i and D_j, update the proximity set Ω.
Terminate trajectories in T, initialize trajectories according to detection responses in D.

4.4 Experiments

4.4.1 Data Sets & Setup

The proposed algorithm is tested on eight data sets, Airshow, Goose, Sailing, Zebra, Crab, Antelope, Flower[1] and Hockey. The first three are new sequences obtained from YouTube videos, and the last five are public sequences [Luo and Kim, 2013, Okuma et al., 2004, Zhang and van der Maaten, 2013b]. These data sets are challenging as they contain (1) crowd scenarios with similar objects, (2) partial or complete occlusions, (3) background clutter and (4) out-of-plane rotations. Parameters λ, α and η in Equation 4.5 are set to be 0.1, 0.001 and 0.001 respectively. The proximity parameter d_{Th} is 20. The number of detectors is 12. For each task, $\frac{2}{3}$ instances are sampled from the whole training data. HOG [Dalal and Triggs, 2005], LBP and colors are extracted as features for object detection. The threshold to determine the confident instances is 0.5. Note that for the public data sets, results are referred from those reported in [Luo and Kim, 2013]. For data sets which are not public, results are obtained by running the authors' code ([Kalal et al., 2012, Zhang and van der Maaten, 2013b]) or by re-implementing the method ([Luo and Kim, 2013] and **K-SVM**).

[1] This sequence is part of the original sequence in [Zhang and van der Maaten, 2013b] (500 frames of the original 2249 frames)

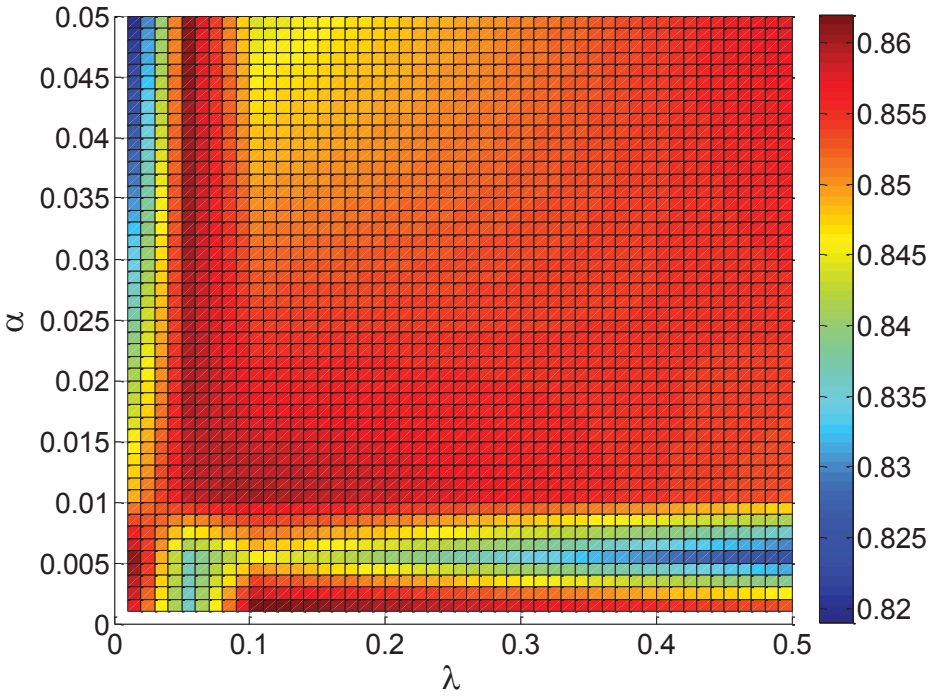

Figure 4.7: *Parameter analysis.*

4.4.2 Parameter Analysis

There are two important parameters, λ and α in Equation 4.5, which weight the spatio-temporal term and the regularization term respectively. I firstly conduct experiments to choose the best value of these two parameters. The metric I employ to choose the parameters is the F score. I vary the value of λ in the range of $[0, 0.5]$ and α in the range of $[0, 0.05]$, apply the detector to the Zebra sequence and calculate the F score based on the recall and precision rates. As shown in Figure 4.7, according to the F score, I choose $\lambda = 0.1$ and $\alpha = 0.001$ and fix them in all the following experiments.

Table 4.1: *Generic object detection results in terms of recall and precision values. The best results are shown in bold, the second best are underlined.*

Sequence	*Recall*					*Precision*				
	eTLD	GMOT-MTL	K-SVM	BL	GMOT-BLP	eTLD	GMOT-MTL	K-SVM	BL	GMOT-BLP
Antelope	0.29	0.74	0.88	0.77	**0.89**	0.57	0.66	0.71	0.76	**0.77**
Goose	0.66	0.80	0.92	0.85	**0.94**	0.94	0.85	0.97	0.98	**0.99**
Zebra	0.60	0.80	0.66	0.74	**0.82**	0.92	**0.97**	0.88	0.91	0.91
Crab	0.22	0.52	0.55	0.56	**0.58**	0.58	0.81	0.70	0.85	**0.88**
Flower	0.21	0.47	0.30	0.50	**0.63**	0.58	0.62	**0.95**	0.94	0.91
Airshow	0.16	0.13	0.38	0.43	**0.63**	0.52	0.56	0.76	**0.77**	0.75
Sailing	0.60	0.63	0.56	0.67	**0.84**	**1.00**	0.93	**1.00**	**1.00**	0.99
Hockey	0.65	**0.84**	0.43	0.65	0.82	0.92	0.89	0.75	0.88	**0.94**
Overall	0.56	0.56	0.56	0.61	**0.70**	0.67	0.79	0.79	0.88	**0.89**

4.4.3 Generic Object Detection

At first, experiments are conducted on generic object detection to verify the effectiveness of the proposed cMTL based detection method. Five methods are compared: (1) **eTLD** [Kalal et al., 2012] which uses a detector based on Random Ferns; (2) **K-SVM**. K independent SVMs are trained on clustered training data from K-means clustering and detect objects by classification. This is a typical way to handle intra-class variance. The number of SVMs is four; the same number of clusters is used in the proposed algorithm; (3) **GMOT-MTL** [Luo and Kim, 2013] is a framework which handles the same problem with a detector based on a Laplacian SVM; (4) a baseline method **BL** which uses cMTL without the spatio-temporal consistency; (5) the full method (termed as **GMOT-BLP**). Table 4.1 shows the results. A detection response is defined as true positive if its overlap with the ground truth bounding box is at least 0.5.

The results indicate that: (1) **eTLD** only discovers a small portion of objects on some sequences. I suspect that this is due to limitations of the **eTLD** detector which uses two-pixel comparisons and therefore cannot handle large intra-class variance; (2) **K-SVM** and **GMOT-**

Table 4.2: *Generic object detection results in terms of recall values. The best results are shown in bold, the second best are underlined.*

Sequence	*Recall*					
	eTLD	mMST	GMOT-MTL	K-SVM	BL	GMOT-BLP
Antelope	0.24±0.05	0.61±0.01	0.70±0.06	**0.85±0.19**	0.76±0.12	0.77±0.14
Zebra	0.54±0.07	0.79±0.03	0.78±0.04	0.57±0.05	0.76±0.02	**0.79±0.02**

Table 4.3: *Generic object detection results in terms of precision values. The best results are shown in bold, the second best are underlined.*

Sequence	*Precision*					
	eTLD	mMST	GMOT-MTL	K-SVM	BL	GMOT-BLP
Antelope	0.62±0.10	0.59±0.00	0.64±0.06	0.67±0.01	0.72±0.07	**0.76±0.10**
Zebra	0.95±0.03	0.92±0.02	**0.97±0.01**	0.70±0.09	0.93±0.04	0.95±0.03

Table 4.4: *Generic object detection results in terms of F values. The best results are shown in bold.*

Sequence	*F Score*					
	eTLD	mMST	GMOT-MTL	K-SVM	BL	GMOT-BLP
Antelope	0.34±0.05	0.60±0.01	0.67±0.06	0.74±0.08	0.74±0.08	**0.76±0.09**
Zebra	0.69±0.05	0.84±0.01	**0.86±0.02**	0.63±0.07	0.84±0.03	**0.86±0.02**

MTL show good performance, and **BL** generally outperforms these, showing the effectiveness of cMTL to handle intra-class variance; (3) the full method **GMOT-BLP** outperforms all other methods; in comparison with **BL** the recall rate is increased due to the spatio-temporal consistency.

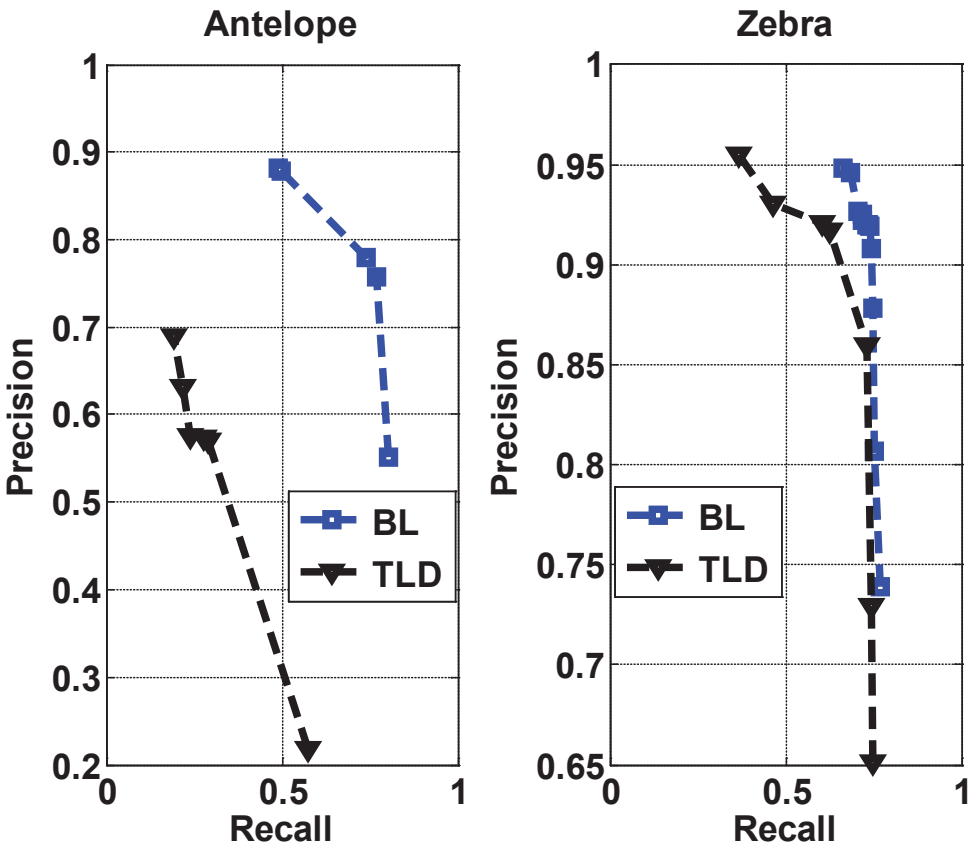

Figure 4.8: *Precision-Recall performance of* ***eTLD*** *and* ***BL*** *on the* Antelope *and* Zebra *sequences.*

To remove the effect from a specific bounding box, I initialize the proposed method with different bounding box and report the recall and precision rates. The experiment is conducted by running 15 times from a randomly chosen bounding box on the Zebra and Antelope sequences. The results are shown in Table 4.2, 4.3 and 4.4 and compared with other methods. In terms of recall and precision metrics, GMOT-BLP achieves either the best or the second best performance on these two data sets. For the F score, which is a trade-off metric considering both recall and precision, GMOT-BLP achieves the best performance on both of the sequences.

In a separate experiment I vary the number K in **K-SVM** as well as the corresponding number of clusters in the proposed algorithm. Two representative public sequences (Antelope and Zebra) are used in this experiment. Table 4.5 shows the results, which demonstrate that the

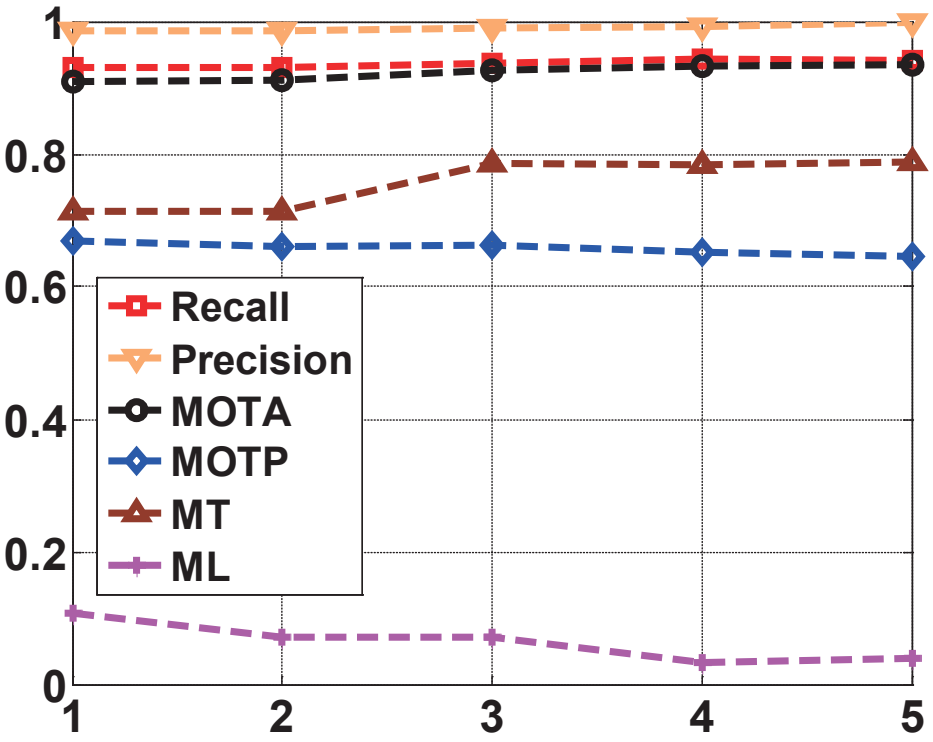

Figure 4.9: *Performance variation of five different initializations on the* Goose *sequence.*

proposed algorithm outperforms **K-SVM** for most K in terms of recall rate, which is important in the setting. I also computed the F score which is a trade-off metric considering both recall and precision. The values are shown in Table 4.6, which reveals that the proposed method is the best considering recall and precision simultaneously on both data sets.

Note that I keep K fixed for the other experiments. A suitable choice of K is beyond the scope of this chapter.

In a more extensive comparison of the baseline method with **eTLD**, I obtain the precision-recall curves for the Antelope and Zebra sequences which are shown in Figure 4.8. **BL** uses a threshold on the score value to determine whether a candidate is an object, and **eTLD** [Kalal et al., 2012] uses the percentage of ferns voting for a positive decision. The results show that the baseline method outperforms **eTLD** consistently.

To test the variation of performance resulting from different initial bounding boxes, I run

Table 4.5: *Comparative results in terms of recall and precision for different values of K (number of SVMs in **K-SVM** and, correspondingly, number of clusters in the proposed method).*

Sequence	**Method**	*Recall*					*Precision*				
	K=	2	4	6	8	*Avg.*	2	4	6	8	*Avg.*
Antelope	K-SVM	0.90	0.88	0.86	0.84	**0.87**	0.66	0.71	0.72	0.73	0.70
	Ours	0.83	0.89	0.80	0.80	0.82	0.81	0.77	0.81	0.80	**0.80**
Zebra	K-SVM	0.66	0.66	0.70	0.70	0.68	0.88	0.88	0.89	0.87	**0.88**
	Ours	0.73	0.82	0.72	0.72	**0.75**	0.85	0.91	0.84	0.84	0.86

Table 4.6: *Comparative results in terms of F score for different values of K (number of SVMs in **K-SVM** and, correspondingly, number of clusters in the proposed method).*

Sequence	**Method**	*F Score*				
	K=	2	4	6	8	*Avg.*
Antelope	K-SVM	.75	.79	.78	.78	.78
	Ours	.80	.83	.80	.80	**.81**
Zebra	K-SVM	.75	.75	.78	.78	.77
	Ours	.79	.86	.78	.78	**.80**

the proposed algorithm five times on the Goose sequence, each time labeling a different initial object. The recall rates are 0.935 ± 0.006 and the precision rates are 0.990 ± 0.004, indicating low dependence on the initialization (see Figure 4.9).

4.4.4 Generic MOT

Experiments are carried out to compare the proposed framework with several state-of-the-art methods on the task of detecting and tracking multiple objects. The experiments are presented in five parts:

(1) For each sequence it is compared with **eTLD** [Kalal et al., 2012] and **GMOT-MTL** [Luo and Kim, 2013]. **eTLD** is originally developed for single object tracking, and I extend it to multiple objects by decreasing the threshold to let it detect some similar objects and track them. It is initialized with the same bounding box as other methods.

Figure 4.10: *Multiple object tracking results shown on frames excerpted from the sequence of Zebra. Different colors correspond to different objects (only 8 colors are adopted so some boxes are of the same color), the yellow lines represent trajectories.*

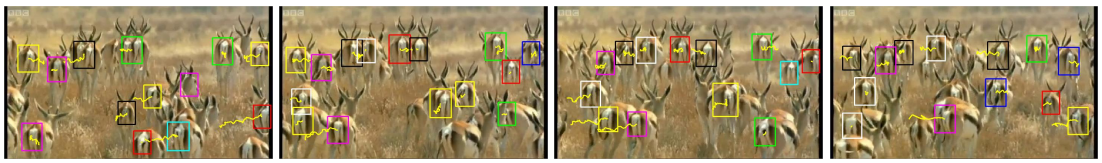

Figure 4.11: *Multiple object tracking results shown on frames excerpted from the sequence of Antelope. Different colors correspond to different objects (only 8 colors are adopted so some boxes are of the same color), the yellow lines represent trajectories.*

(2) For the Zebra, Crab, Flower, Airshow and Sailing sequences, I apply **SPOT** [Zhang and van der Maaten, 2013b] to track multiple objects (four in the experiments) in each sequence. To compare the performance, I excerpt results corresponding to these four objects from the whole result in each sequence and evaluate them. It is worth noting that the proposed algorithm starts with a single bounding box while **SPOT** [Zhang and van der Maaten, 2013b] starts with all four bounding boxes for each sequence.

(3) For the Hockey sequence, I additionally compare the proposed method with [Brendel et al., 2011, Breitenstein et al., 2009, Okuma et al., 2004] using the results from [Luo and Kim, 2013].

Example images are shown from Figure 4.10 to Figure 4.17. I adopt the criteria of MOTA, MOTP proposed in [Keni and Rainer, 2008], as well as MT trajectories and ML trajectories [Li et al., 2009] to give quantitative results. As shown in Table 4.7, the arrows following the criteria indicate the trend of better performance.

Results in Table 4.7 show that: (1) compared with **TLD** and **GMOT-MTL**, the proposed method outperforms other methods on most sequences; (2) compared with **SPOT**, the proposed

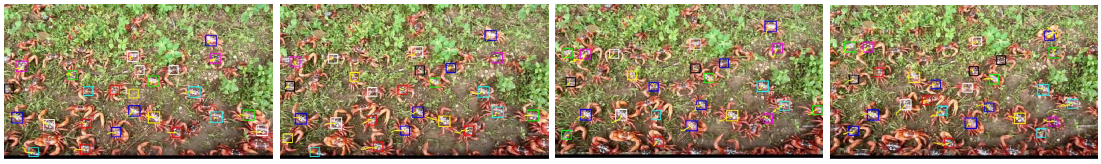

Figure 4.12: *Multiple object tracking results shown on frames excerpted from the sequence of Crab. Different colors correspond to different objects (only 8 colors are adopted so some boxes are of the same color), the yellow lines represent trajectories.*

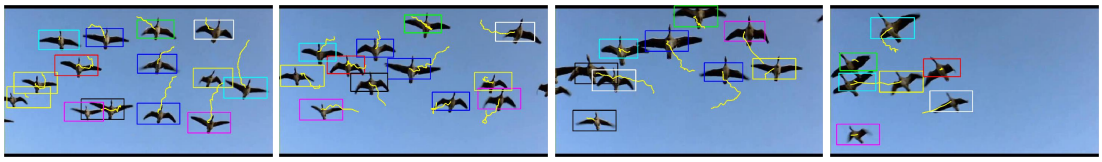

Figure 4.13: *Multiple object tracking results shown on frames excerpted from the sequence of Goose. Different colors correspond to different objects (only 8 colors are adopted so some boxes are of the same color), the yellow lines represent trajectories.*

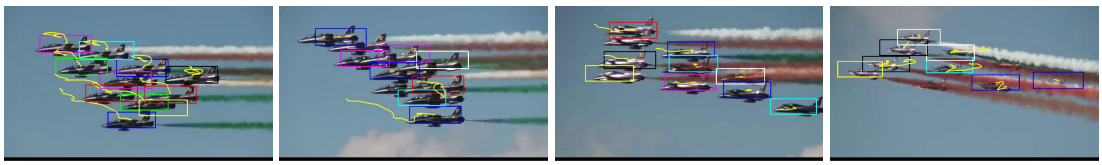

Figure 4.14: *Multiple object tracking results shown on frames excerpted from the sequence of Airshow. Different colors correspond to different objects (only 8 colors are adopted so some boxes are of the same color), the yellow lines represent trajectories.*

approach achieves better results except on the MOTP metric. It is suspected that this is due to **SPOT** trackers being object-specific, thereby obtaining greater overlap scores, i.e. larger MOTP values; (3) for the Hockey sequence, the proposed method obtains results comparable with methods that use a specific off-line trained human detector.

In order to test the sensitivity on different initializations, I run the proposed algorithm on the Goose sequence five times with different initial bounding boxes. The MOTA, MOTP, MT and ML are 0.935 ± 0.012, 0.660 ± 0.009, 0.750 ± 0.042 and 0.071 ± 0.029 respectively (see Figure 4.9), indicating low sensitivity to the initial labeling.

(4) To remove the effect from a specific bounding box, I initialize the proposed method

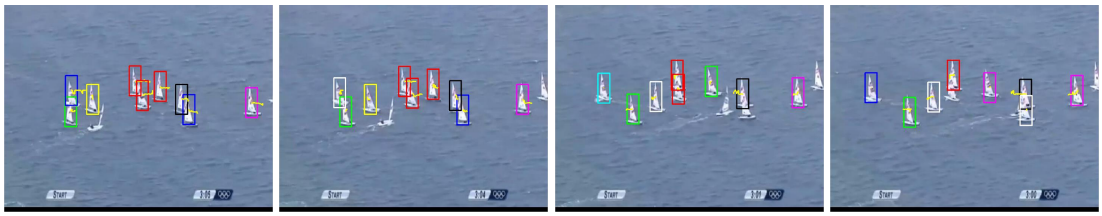

Figure 4.15: *Multiple object tracking results shown on frames excerpted from the sequence of Sailing. Different colors correspond to different objects (only 8 colors are adopted so some boxes are of the same color), the yellow lines represent trajectories.*

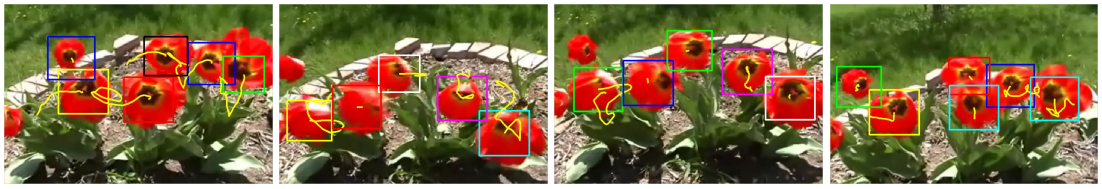

Figure 4.16: *Multiple object tracking results shown on frames excerpted from the sequence of Flower. Different colors correspond to different objects (only 8 colors are adopted so some boxes are of the same color), the yellow lines represent trajectories.*

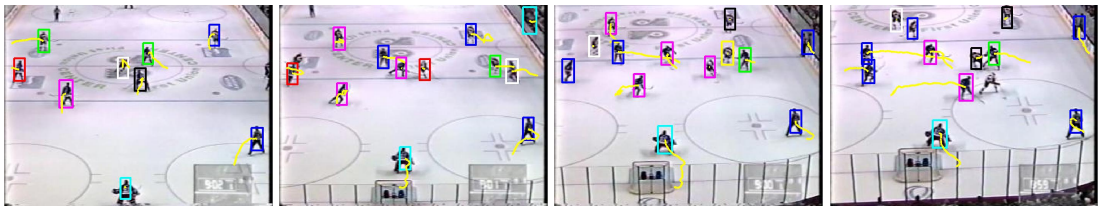

Figure 4.17: *Multiple object tracking results shown on frames excerpted from the sequence of Hockey. Different colors correspond to different objects (only 8 colors are adopted so some boxes are of the same color), the yellow lines represent trajectories.*

with different bounding box and report the tracking performance. The experiment is conducted by running 15 times from randomly chosen bounding boxes on Zebra and Antelope sequences. The results are shown in Table 4.8 and 4.9 and compared with other methods.

The results show that, the proposed GMOT-BLP method achieves the best performance in terms of MOTA and MT metrics on both sequences while performs slightly worse than GMOT-MTL in terms of other metrics. It is supposed to be the result of the tracking management (such as "buffer") in GMOT-MTL. In the case of false negative (missed detections), the "buffer"

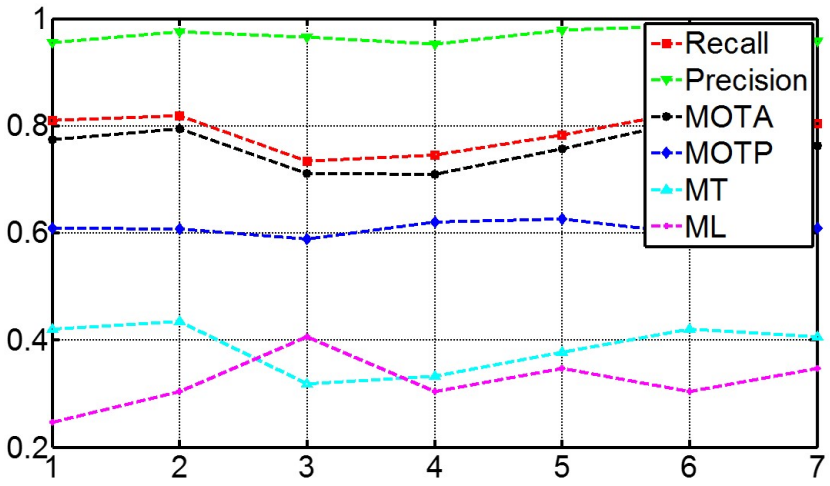

Figure 4.18: *Performance variation of the proposed method initialized by different numbers of initial bounding boxes on the* Zebra *sequence.*

could help to recover more instances of objects, thus results in greater values of FM and IDS. In the proposed method in this chapter, there is not such a mechanism, as determining the length of such a kind of "buffer" is heuristic.

(5) Additionally, to test the sensitivity on different numbers of initial bounding boxes, the proposed method is initialized with different numbers of initial bounding boxes (ranging from 1 to 7 in the experiment), and the results from different initial settings are compared, which are shown in Figure 4.18 and 4.19. The results generally suggest that more initial bounding boxes lead to better performance.

The computation speed is about 0.5 FPS on a desktop of Intel i7-965 CPU, 8G ram and unoptimized Matlab code.

4.5 Remarks

This chapter has presented a framework for tracking multiple objects of the same general type, where class and object labels are propagated in the spatio-temporal domain. I have introduced

cMTL for generic object detection and have shown the benefit of considering spatio-temporal consistency. The proposed method takes a sequential approach, entailing the limitation that object trajectories may be more fragmented than when taking a more global view of the data. Comparative experiments on eight sequences (five public and three new data sets) confirmed the effectiveness of the proposed method.

From a practical viewpoint, an advantage of the proposed method over most other work in the area is the requirement of labeling just a single initial bounding box, thereby providing a multi-object tracker without resorting to an off-line trained detector. However, tracking management is still challenging and there are heuristics in the proposed online-data-association solution. These also deteriorate the tracking performance. I therefore resort to a batch solution in the next chapter.

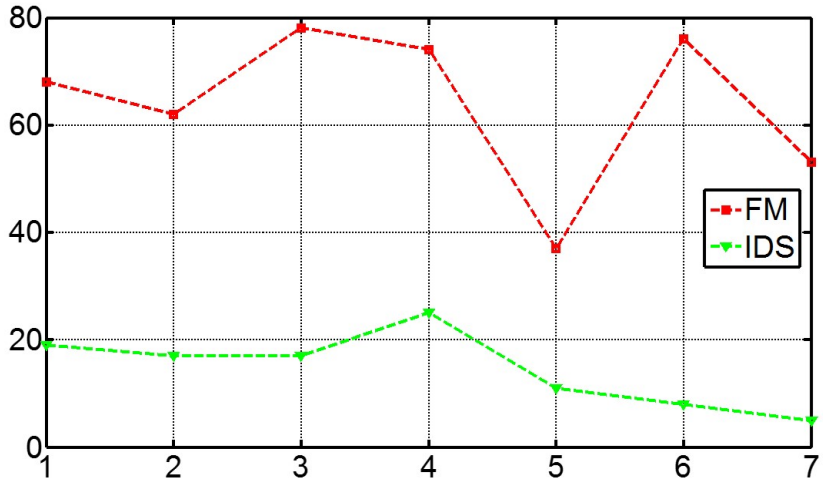

Figure 4.19: *Performance variation of the proposed method initialized by different numbers of initial bounding boxes on the* Zebra *sequence.*

Table 4.7: *Generic Multiple Object Tracking results.*

Sequence	Method	MOTA ↑	MOTP↑	MT ↑	ML ↓	FM ↓	IDS ↓
Antelope	eTLD	0.088	0.650	0.235	0.765	86	58
	GMOT-MTL	0.356	0.633	0.368	0.368	**33**	**19**
	GMOT-BLP	**0.622**	**0.714**	**0.691**	**0.177**	54	31
Goose	eTLD	0.621	0.611	0.286	0.179	169	58
	GMOT-MTL	0.798	0.604	0.643	0.071	52	**28**
	GMOT-BLP	**0.938**	**0.649**	**0.786**	**0.036**	**36**	33
Zebra	eTLD	0.587	0.645	0.159	0.420	100	24
	GMOT-MTL	**0.777**	0.668	0.435	0.304	36	**6**
	GMOT-BLP	0.743	**0.683**	**0.580**	**0.246**	**30**	7
	SPOT	0.661	**0.753**	0.750	0	-	-
	GMOT-BLP	**0.982**	0.747	**1.000**	0	-	-
Crab	eTLD	0.068	0.646	0.049	0.864	235	101
	GMOT-MTL	0.391	0.600	0.097	0.709	243	114
	GMOT-BLP	**0.497**	**0.692**	**0.214**	**0.689**	**205**	**63**
	SPOT	0.190	**0.766**	0.500	0.250	-	-
	GMOT-BLP	**0.924**	0.724	**1.000**	**0**	-	-
Flower	eTLD	0.053	0.677	0	0.632	244	126
	GMOT-MTL	0.186	0.650	0.053	0.421	176	85
	GMOT-BLP	**0.566**	**0.718**	**0.316**	**0.368**	**78**	**67**
	SPOT	0.372	0.730	0.500	0.250	-	-
	GMOT-BLP	**0.524**	**0.737**	0.500	**0**	-	-
Airshow	eTLD	0.013	0.596	0	0.733	92	54
	GMOT-MTL	0.028	**0.716**	0	0.867	**32**	**23**
	GMOT-BLP	**0.415**	0.646	0	**0.067**	121	58
	SPOT	-0.503	**0.676**	0	0.250	-	-
	GMOT-BLP	**0.346**	0.650	0	**0**	-	-
Sailing	eTLD	0.403	**0.737**	0.250	0.083	171	54
	GMOT-MTL	0.548	0.684	0.250	0.083	99	33
	GMOT-BLP	**0.819**	0.640	**0.833**	0.083	**45**	**12**
	SPOT	0.554	**0.731**	0.750	0.250	-	-
	GMOT-BLP	**0.786**	0.652	0.750	**0**	-	-
Hockey	eTLD	0.547	0.647	0.179	0.250	64	43
	GMOT-MTL	**0.803**	0.691	**0.679**	**0.107**	27	**17**
	GMOT-BLP	0.766	**0.736**	0.607	0.143	**24**	20
	Brendel et al.	0.797	0.600	-	-	-	-
	Breitenstein et al.	0.765	0.570	-	-	-	-
	Okuma et al.	0.678	0.510	-	-	-	-
Overall	eTLD	0.279	0.655	0.140	0.602	1161	518
	GMOT-MTL	0.410	0.637	0.310	0.427	698	325
	GMOT-BLP	**0.613**	**0.685**	**0.482**	**0.336**	**593**	**291**
	SPOT	0.235	**0.728**	0.500	0.200	-	-
	GMOT-BLP	**0.629**	0.703	**0.650**	**0**	-	-

Table 4.8: *Generic Multiple Object Tracking results. The table shows results in terms of four performance criteria from the literature (arrows indicating direction of better performance) on data sets of* Zebra *and* Antelope.

Sequence	**Method**	MOTA ↑	MOTP↑	MT ↑
	TLD	.529±.065	.667±.020	.188±.038
	mMST	.465±.282	.573±.081	.198±.268
Zebra	GMOT-MTL	.753±.028	**.668±.001**	**.420±.052**
	GMOT-BLP(BL)	.712±.046	.600±.018	.300±.044
	GMOT-BLP	**.758±.021**	.608±.022	.401±.085
	TLD	.089±.031	**.680±.041**	.015±.026
	mMST	.189±.005	.599±.001	.191±.062
Antelope	GMOT-MTL	.301±.122	.625±.043	.328±.056
	GMOT-BLP(BL)	.464±.136	.658±.048	.377±.191
	GMOT-BLP	**.521±.160**	.657±.034	**.446±.165**

Table 4.9: *Generic Multiple Object Tracking results. The table shows results in terms of four performance criteria from the literature (arrows indicating direction of better performance) on data sets of* Zebra *and* Antelope.

Sequence	**Method**	ML ↓	FM ↓	IDS ↓
	TLD	.435±.025	110.000±9.539	40.274±3.647
	mMST	.556±.230	59.250±16.681	22.454±4.787
Zebra	GMOT-MTL	.343±.044	**28.333±0.577**	**5.941±0.878**
	GMOT-BLP(BL)	.329±.051	103.667±12.503	32±4.583
	GMOT-BLP	**.300±.036**	68.667±15.176	12.333±2.887
	TLD	.775±.059	117.667±42.525	70.148±30.426
	mMST	.515±.021	93±2.828	60.067±2.145
Antelope	GMOT-MTL	.407±.081	**91±28.827**	**52.812±10.874**
	GMOT-BLP(BL)	.270±.073	148±56.152	94.667±39.311
	GMOT-BLP	**.284±.075**	112.667±45.358	69.667±39.577

CHAPTER 5

AUTOMATIC TOPIC DISCOVERY FOR GENERIC MOT

As discussed in the previous chapter, issues like tracking management are challenging in online solutions. This chapter therefore seeks a better solution. For generic multiple object tracking, the task is to link a number of given detection hypotheses to trajectories corresponding to different objects in a video. There has been significant progress in multi-object tracking [Zhang et al., 2008, Pellegrini et al., 2009, Xing et al., 2009, Pirsiavash et al., 2011, Milan et al., 2013b, Luo et al., 2014a, Leal-Taixé et al., 2014]. However, issues like tracking management and appearance variations remain challenging. Traditionally, the multi-object tracking task is cast as a data association problem in which detection hypotheses are associated into trajectories. Standard methods, such as the Hungarian algorithm, can be readily applied, however several practical considerations remain:

- Temporal gaps between observations may lead to disconnected trajectories of the same object [Zhang et al., 2008, Pirsiavash et al., 2011]. Determining the maximum allowable

gap is difficult: low values will cause more fragmentation while higher values lead to more incorrect associations (ID switches).

- Handling track initialization and termination (also known as *tracking management*) is often based on heuristics. An existing trajectory may be terminated in the case of a single missing detection, resulting in fragmentation in some sequential approaches [Luo and Kim, 2013, Luo et al., 2014a]. Heuristically, some approaches retain a "buffer" [Shu et al., 2012] to better initialize a trajectory. Here "buffer" means a trajectory will not be formally initialized until the length of a potential trajectory exceeds a pre-defined threshold.

- Appearance variation of objects may lead to fragmentation or ID switches as a result of inappropriate similarity measures.

- Physical constraints are rarely modeled explicitly, the work in [Milan et al., 2013b] being one exception. Uniqueness constraints model the fact that (a) at most one object per frame can be associated with each trajectory, and (b) no more than one trajectory can be assigned to the same detected object.

In this chapter I propose an alternative approach to temporal data association by clustering detection instances, where each cluster corresponds to a unique object. A text-document analogy is introduced, where an object is represented as a set of visual words. Based on the visual word representation, I observe that different instances of a unique object exhibit the same pattern while those of different objects show different patterns. Therefore an object corresponds to a semantic topic within a video sequence. The object is tracked as a topic which evolves over time and fades away.

A Dirichlet Process Mixture Model (DPMM) is employed to dynamically cluster detection responses into sets of objects (see Figure 5.1 for the graphical model of the proposed approach). The merit of applying a DPMM is that the number of semantic topics is learned automatically. Furthermore, it is naturally feasible to model dynamics in the clustering procedure for semantic topic discovery based on DPMM [Ahmed and Xing, 2008].

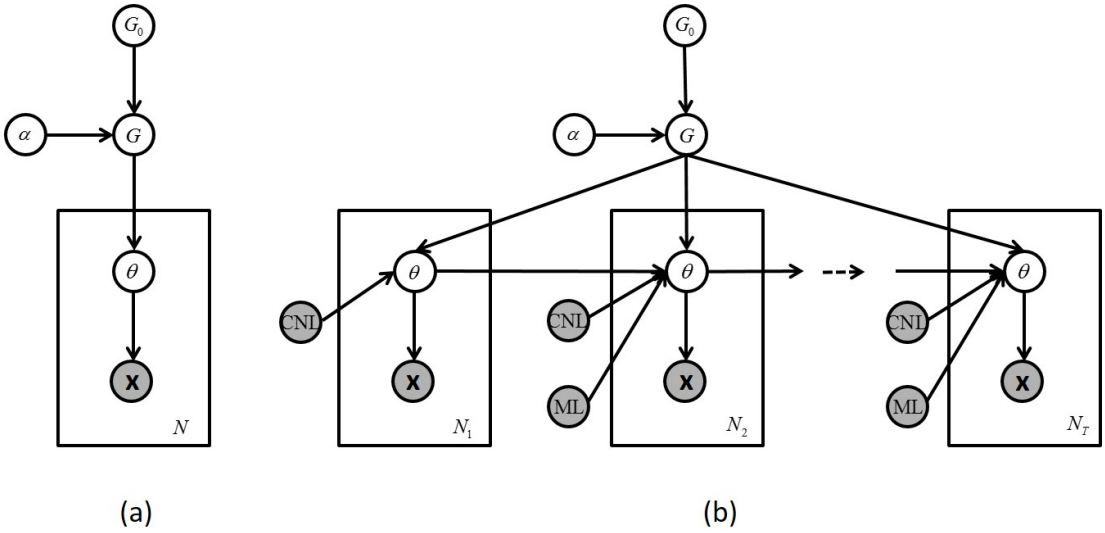

Figure 5.1: *Graphical model of the standard DPMM (a) and the proposed topic model (b). In the proposed model a document is temporally divided into epochs to model the temporal dynamics.* **CNL** *and* **ML** *denote the introduced cannot-link and must-link constraints.*

Specifically, a detection hypothesis is treated as a set of visual words. The uniqueness of word co-occurrence results in individual clusters, corresponding to individual objects by the application of DPMM. In a standard DPMM, when considering the assignment of a given instance, the prior of which cluster the instance should belong to depends only on the number of existing instances in the cluster. However, in the proposed solution, the temporal distances between clusters and the given instance are also taken into consideration. Therefore, instead of treating the whole video as a single document, I divide it into sequential *epochs* in order to model the dynamics of prior knowledge. On the other hand, adopting the temporal phenomena, the appearance variations of objects are dealt with by updating cluster parameters in different epochs in the clustering procedure.

In terms of the exclusive constraints, which are (a) at most one object per frame can be associated with each trajectory, and (b) no more than one trajectory can be assigned to the same detected object, by adopting clustering, the exclusivity constraint (b) is handled naturally by the assignment of each detection to only one cluster. To deal with the other constraint, the so-called *cannot-link* constraints are introduced to prohibit two detections in the same frame

being assigned to one trajectory.

By taking the scheme of dynamic clustering as a basic framework, I firstly tackle the problem of tracking rigid objects with the superpixel representation. Then the problem of tracking pedestrians is handled with the Deformable Part Model (DPM), within which not only the holistic but also the part-wise visual information are considered. On public data sets I conduct experiments of both rigid object tracking and pedestrian tracking and compare with state-of-the-art methods. The comparison could directly validate the effectiveness of the automatic topic discovery approach.

To summarize, the benefits of automatic topic discovery for multi-object tracking are (1) multi-object tracking is cast as dynamic and sequential clustering by the application of DPMM without heuristics like "buffer" or maximum allowable temporal gap and tracking management is handled automatically in the clustering procedure, (2) appearance variation of objects is modeled by the dynamics of cluster parameters, (3) exclusivity constraints are handled naturally as a result of the cluster assignments and the introduction of the *cannot-link* constraints to the model, (4) a dynamic clustering algorithm is provided as a tracking solution which could serve as a basic framework to integrate various kinds of observation models for improved tracking performance.

The remainder of the chapter is organized as follows: Section 5.1 and Section 5.2 briefly introduce topic model and DPMM. Section 5.3 presents the proposed dynamic clustering model for multiple object tracking, specifically for both rigid objects and non-rigid objects. Section 5.4 describes the inference of the proposed model. In Section 5.5 experiment results corresponding to the two problems mentioned in Section 5.3 are reported. Section 5.6 concludes the chapter.

Note that, I did not develop a detection method in this chapter. This chapter focuses only on developing a batch solution to tracking multiple generic objects based on existing given detection results from the previous chapter.

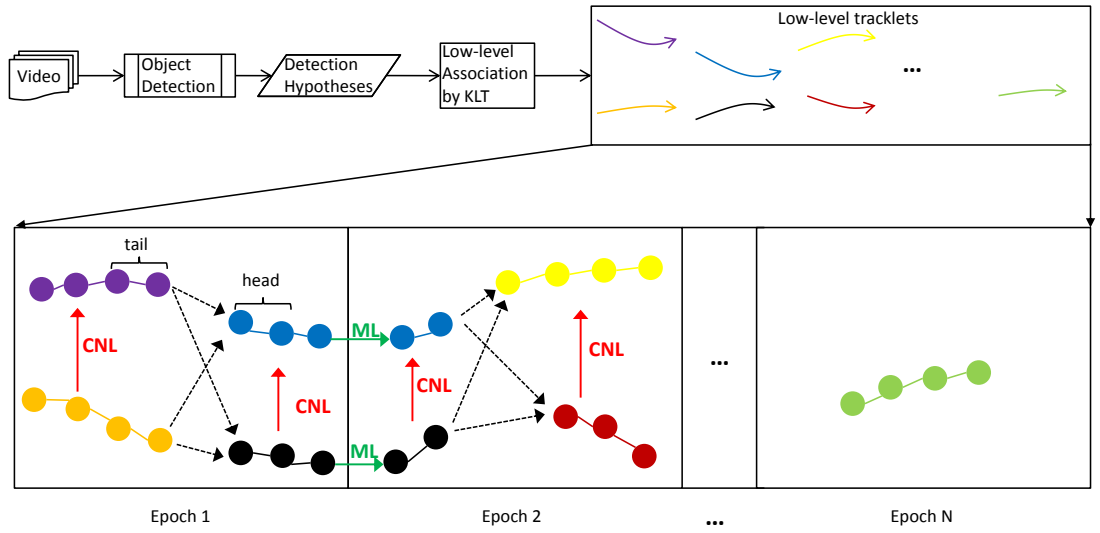

Figure 5.2: *Schematic of the proposed method. Tracklets are shown in different colors. Potential assignments are shown by dashed arrows. Temporally overlapping tracklets cannot be clustered together due to the* **cannot-link constraint** *(solid red arrows). The black tracklet and the blue tracklet temporally cross continuous epochs, and the segments of them are connected by the* **must-link constraint** *(solid green arrow). Note that, the purple and the orange tracklets in Epoch 1 could be directly connected to the yellow tracklet and the dark red tracklet in Epoch 2. In the last epoch, there is only one tracklet. Considering the temporal damping effect, the prior that this tracklet is linked to tracklets in previous epochs is limited if there is no intermediate tracklet bridging them. Some possible assignment arrows (for example the purple and the yellow tracklets could be possibly associated without linking the blue one) are dismissed for the clarity of the figure (best viewed in color).*

5.1 Topic Model

Topic model has a long history in natural language processing, pattern recognition and computer vision. It can at least be dated back to Latent Semantic Indexing (LSI) [Dumais et al., 1995] or Latent Semantic Analysis (LSA), which is applied in text analysis. This method is based on the principle that terms (or words) used in the same context are probably related to an identical topic. Based on LSA, probabilistic Latent Semantic Analysis (pLSA) [Hofmann, 1999] is proposed. pLSA models the co-occurrence of words and documents as a mixture of conditionally independent multinomial distributions. Further, Latent Dirichlet Allocation (LDA) [Blei et al., 2003] is developed for the task of text analysis. It is a generative model based on the bag-of-word assumption. Each document is assumed to be a combination of mul-

tiple topics, and each word is generated by one of the topics the document includes. Compared with pLSA, the distribution of topics in LDA has a Dirichlet prior. Dirichlet Process [Teh, 2010] is a distribution over distributions, i.e., each draw from Dirichlet process is a distribution. It is particularly employed by Dirichlet process mixture model, which is also called the infinite mixture model. Dirichlet process is the basis of the hierarchical Dirichlet process [Teh et al., 2006], within which the distribution of child is itself a Dirichlet process.

Topic models typically employ the concepts of words, topics and documents. Specifically, by treating a document as a bag of exchangeable words, documents are modeled as distributions over topics and topics are modeled as distributions over words. Thus topic models are naturally employed to deal with tasks of text analysis and natural language processing. On the other hand, they have been adopted to computer vision tasks in recent years due to the merits of these methods for discovering thematic structure. For example, a latent topic model is developed for object segmentation and classification [Cao and Fei-Fei, 2007]. This so-called Spatial LTM enforces the spatial coherency of the model and can simultaneously segment and classify objects. Similarly, spatial information is also integrated into a LDA model in [Wang and Grimson, 2008] for image segmentation by Wang and Grimson. Topic models have been applied to numerous other tasks, such as region classification [Verbeek and Triggs, 2007], trajectory analysis [Wang et al., 2011], image annotation [Wang et al., 2009a] and image scene categorization [Fei-Fei and Perona, 2005].

Topic model has been applied to multiple object tracking. For example, in [Topkaya et al., 2013], DPMM is adopted to cluster foreground pixels into parts of objects and MRF is employed to refine object boundary. However, this method does not model object dynamics in the tracking stage. This method is supposed to work only in the case of fixed background because of the requirement of foreground pixel extraction. At the same time, this method is online, which basically would not outperform batch methods.

5.2 DPMM

The Dirichlet Process Mixture Model (DPMM) [Blei et al., 2006] is a non-parametric model which assumes the data is governed by an infinite number of mixtures while only a fraction of these mixtures are activated by the data. Figure 5.1(a) shows the graphical model of a DPMM. Assuming that the k-th mixture is parameterized by θ_k, each sample $\mathbf{x}_i$ is generated as follows:

$$\begin{aligned} G|\alpha, G_0 &\sim DP(\alpha, G_0) , \\ \theta_k|G &\sim G , \\ \mathbf{x}_i|\theta_{z_i} &\sim F(\theta_{z_i}) , \end{aligned} \tag{5.1}$$

where $DP(\bullet)$ is a Dirichlet process, G_0 is a base distribution, α is a concentration parameter, θ_k is drawn from G, which itself is a distribution drawn from the Dirichlet process, and $F(\theta_{z_i})$ denotes the distribution of observation $\mathbf{x}_i$ given θ_{z_i}, where z_i is the mixture indicator of $\mathbf{x}_i$. When this model is applied to clustering, z_i is the cluster index. Note that the number of mixtures in the model is determined by the data, i.e. the number of clusters is learned automatically, in contrast to parametric models such as K-means.

The Chinese Restaurant Process (CRP) illustrates the DPMM intuitively: Assuming an infinite number of tables (clusters), a new customer (observation) chooses an empty table with probability depending on α or joins an occupied table with a probability proportional to the number of people seated at that table. Formally,

$$\theta_i|\theta_{-i}, G_0, \alpha \sim \sum_k \frac{n_k}{i-1+\alpha}\delta(\phi_k - \theta_i) + \frac{\alpha}{i-1+\alpha}G_0 , \tag{5.2}$$

where ϕ_k is the parameter of cluster k, θ_{-i} is the set of associated parameters of $\mathbf{x}_{-i}$, i.e. observations except $\mathbf{x}_i$, n_k is the number of customers already at table k and $\delta(\cdot)$ is the Dirac delta function centered at 0. $\phi_{1:k}$ is the discrete set of values of $\{\theta_i\}$. It can also be written as $\theta_i = \phi_k$ with probability $\frac{n_k}{i-1+\alpha}$, and $\theta_i = \phi_{new}, \phi_{new} \sim G_0$ with probability $\frac{\alpha}{i-1+\alpha}$.

5.3 Automatic Topic Discovery

In this section a topic model addressing the multi-object tracking problem is presented. For different types of objects, i.e. rigid objects or non-rigid objects, different kinds of representation are adopted. Videos and trajectories/objects are considered as documents and discovered topics. Coherent detection hypotheses (word co-occurrences) are clustered into trajectories (topics). As the number of objects/trajectories is not known in advance, it is learned from the data using a DPMM. Figure 5.2 illustrates the schematic of the proposed approach. Given a video, detection hypotheses are obtained by applying a ready-to-use object detector. Then these detection hypotheses are linked via KLT as low-level but reliable tracklets. These tracklets are the input of the dynamic clustering procedure, which groups tracklets belonging to an identical object into a cluster. In this stage, the exclusivity (cannot-link and must-link constraints) and the temporal damping effect are taken into account.

To adopt DPMM, the following analogues described in Table 5.1 are made. The left column lists some concepts in the multi-object tracking problem, and the right column represents the corresponding entities in the proposed approach of automatic topic discovery. Classical text-analysis applications of DPMM assume that the document consists of a bag of exchangeable words, i.e. without specific order and without any dynamic modeling. In the MOT problem, words are not assumed to be exchangeable as the set of visual words in a detection hypothesis is jointly considered and the representation of visual words in this chapter additionally encodes the spatial information. As appearance of object varies (temporal dynamics of word-occurrence), the distributions of visual words in an object (topic) are dynamic across the video. In light of this, the video is divided temporally into epochs and each epoch is modeled by a DPMM with associated hyper-parameters. During the clustering procedure, the states of clusters are updated across epochs.

Further, as objects appear and disappear, corresponding to the birth and death of topics, the distributions of topics also vary across different epochs. It is also observed that between two adjacent epochs, the distribution of words in a topic and the distributions of topics in

Table 5.1: *This table lists the correspondences between the topic model and the multi-object tracking problem.*

Multi-object tracking	Automatic topic discovery
Detections	Word occurrence
Trajectory	Topic
Video	Document
Video segment	Epoch
Exclusive constraints	Cluster membership exclusivity and cannot links
Data association	Dynamic clustering

a document are closely related to each other due to temporal continuity. Thus the relation between continuous DPMMs is modeled as a first-order Markov process. Figure 5.1(b) shows the graphical model of the proposed approach. Each epoch (except the first epoch) is closely related to its previous epoch. Additionally, the spatio-temporal exclusivity, i.e. the cannot-link and the must-link constraints, is taken into consideration.

5.3.1 DPMM-SP for Generic MOT

In this section, the visual representation based on super pixels and the likelihood computation in the tracking problem of multiple rigid objects are described.

5.3.1.1 Visual Representation

Figure 5.3 shows the visual representation of rigid objects in the GMOT problem. I adopt superpixels, i.e. pixel groups of similar color and location [Achanta et al., 2012], for representing visual appearance. In the implementation, a detection bounding box is segmented into approximately 200 SLIC superpixels [Achanta et al., 2012], each described as a 5-dimensional vector (r, g, b, x, y), where (r, g, b) and (x, y) are the mean color and position, respectively. All superpixels from all frames in the video are clustered by K-means and a dictionary is defined from the cluster prototypes. Each bounding box is quantized using this dictionary and represented

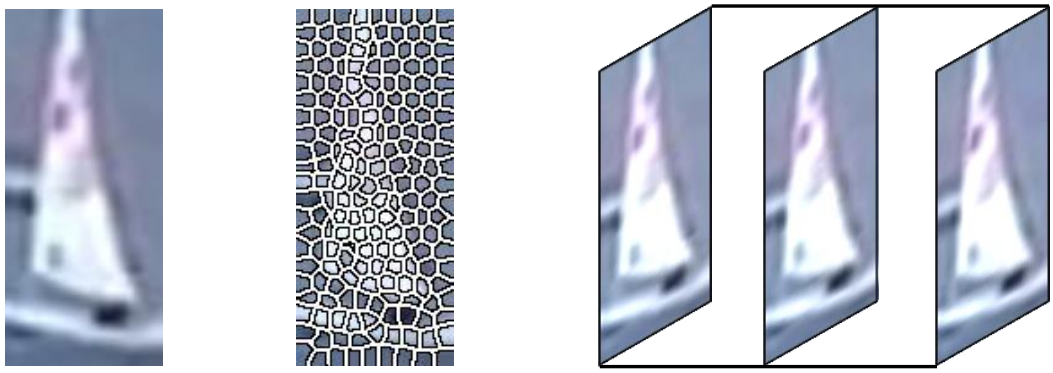

Figure 5.3: *Visual representation based on superpixel. A detection bounding box (a) is segmented into a set of super-pixels shown in (b). The rightmost side (c) shows an exemplar tracklet.*

as a histogram. Similar to part-based models [Felzenszwalb et al., 2010] this object representation exhibits some robustness to partial occlusion since some superpixels representing the object will remain visible.

Usually the detection responses are linked into low-level reliable tracklets [Kuo et al., 2010] in a pre-processing step. Here KLT tracking is employed to obtain N low-level tracklets, $\mathbf{x}_{1:N}$. Each tracklet is represented as a tuple $\mathbf{x}_i = \left\langle A_i^{head}, A_i^{tail}, T_i^{head}, T_i^{tail}, \widetilde{A}_i, \widetilde{V}_i \right\rangle$, where A_i^{head} and A_i^{tail} are the appearance representations (histograms) of the head and tail element within tracklet $\mathbf{x}_i$, $\widetilde{A}_i$ and $\widetilde{V}_i$ are the average appearance (center) and the covariance of histograms of the complete tracklet, T_i^{head} and T_i^{tail} are the time indexes of the head and tail element in $\mathbf{x}_i$.

5.3.1.2 Likelihood

Based on the object representation, parameters of the cluster k at epoch t, $\phi_{t,k}$, including the center $\widetilde{A}_{t,k}$ and covariance matrix $\widetilde{V}_{t,k}$, are computed from the super-pixel representation of all the detection within the concerned cluster up to the current epoch. Given $\mathbf{x}_{t,i}$, the likelihood of an observation given a cluster is estimated as

$$f\left(\mathbf{x}_{t,i}|\phi_{t,k}, \mathbf{x}_{t,k,\cdot}\right) \propto s\left(A_{t,i}^{head}, A_{t,k,m}^{tail}\right) s\left(A_{t,i}^{tail}, A_{t,k,n}^{head}\right) p\left(\widetilde{A}_{t,i}; \phi_{t,k}\right) , \tag{5.3}$$

where $\mathbf{x}_{t,k,\cdot}$ is the set of observations associated with $\phi_{t,k}$, $A^{tail}_{t,k,m}$ and $A^{head}_{t,k,n}$ are the feature vector of tail tracklet $\mathbf{x}_{t,k,m}$ and head tracklet $\mathbf{x}_{t,k,n}$ which are closest to $\mathbf{x}^{head}_{t,i}$ and $\mathbf{x}^{tail}_{t,i}$ respectively regarding temporal distance.

$s(\cdot,\cdot)$ is the similarity between two histograms. It has the following form:

$$s(A_1, A_2) = \exp\left(-Bhatt(A_1, A_2)\right) . \tag{5.4}$$

$p(\widetilde{A}_{t,i};\phi_{t,k})$ is the likelihood of $\widetilde{A}_{t,i}$ given $\phi_{t,k}$. It is computed with regard to the distance between two Gaussian distributions, one corresponding to the cluster and the other one corresponding to the concerned tracklet. To be concrete, it is reversely proportional to the distance D between the cluster and tracklet, as

$$p\left(\widetilde{A}_{t,i};\phi_{t,k}\right) \propto \exp\left(-D\left(\widetilde{A}_{t,i}, \widetilde{V}_{t,i}, \widetilde{A}_{t,k}, \widetilde{V}_{t,k}\right)\right) , \tag{5.5}$$

where $D\left(\widetilde{A}_{t,i}, \widetilde{V}_{t,i}, \widetilde{A}_{t,k}, \widetilde{V}_{t,k}\right)$ is with the following form

$$\begin{aligned} &D\left(\widetilde{A}_{t,i}, \widetilde{V}_{t,i}, \widetilde{A}_{t,k}, \widetilde{V}_{t,k}\right) \\ &= \left(\widetilde{A}_{t,i} - \widetilde{A}_{t,k}\right)^T \left(\frac{\widetilde{V}_{t,i} + \widetilde{V}_{t,k}}{2}\right)^{-1} \left(\widetilde{A}_{t,i} - \widetilde{A}_{t,k}\right) . \end{aligned} \tag{5.6}$$

Note that in Equation 5.3 the first two terms compute the local affinity and the last term computes the global affinity in terms of temporal span.

5.3.2 (DPM)^2 for Multiple Pedestrian Tracking

In this section, DPMM is combined with the Deformable Part Model (DPM) [Felzenszwalb et al., 2010] to deal with the problem of tracking multiple pedestrians. Compared with tracking multiple rigid objects in Section 5.3.1, the visual representation and the likelihood computation are different, which would be illustrated as follows.

5.3.2.1 Visual Representation

Due to the non-rigid property of pedestrians, rather than the super-pixel representation, DPM [Felzenszwalb et al., 2010] is adopted to represent objects. DPM has been very successful in object detection while not been fully exploited in object tracking, [Shu et al., 2012, Izadinia et al., 2012b] being exception. The DPM model represents an object with a root filter and a set of part filter. The final score accounts for the appearance feature in the parts along with the deformation cost of the parts regarding the root filter. It is observed that, the position, size, appearance of parts of pedestrians exhibit unique pattern from person to person. On the other hand, the co-occurrence of these parts are similar if concerning the same person. From this perspective, parts are treated as words in the document, and the tracking problem is naturally a topic discovering problem when pedestrians are treated analogously to topics.

To be specific, an object is represented based on the holistic bounding box along with a set of parts (the number of parts is 8), which are the outputs of a DPM detector. HOG and color features are extracted from the holistic bounding box and the associated parts as the appearance information for this detection hypothesis. Additionally, the configuration of the set of parts within the holistic bounding box is exploited. Generally, the head part is the part which is visible almost all the time, even in case of (partial) occlusion. By taking the head part as an anchor, the offset of the other parts could be calculated. The spatial offset values are stacked as feature and termed as deformable feature. The appearance feature and the deformable feature constitute the visual representation. Similar to the case of tracking rigid objects, KLT is employed to link detections into low-level tracklets. For one tracklet, the head and the tail of the tracklet are represented separately. Each tracklet is represented as a tuple as $\mathbf{x}_i = \left\langle A_i^{head}, A_i^{tail}, \widetilde{D}_i, \widetilde{V}_i, T_i^{head}, T_i^{tail} \right\rangle$, where A_i^{head} and A_i^{tail} are the appearance feature of the head and the tail, $\widetilde{D}_i$ and $\widetilde{V}_i$ are the mean and the covariance of the deformable configuration of the tracklet, T_i^{head} and T_i^{tail} are the frame indexes of the starting frame and the ending frame of the tracklet.

Figure 5.4: *Visual representation based on DPM. (a) Detection samples of continuous frames from the TUD-Stadtmitte data set. Note the part configuration of the same person and different persons. (b) and (c) show the visualization results of the part configuration of the same person at different times, while (d) shows the visualization result of a different person. Based on the likelihood represented in the following, similarity value between (b) and (c) is larger than that between (c) and (d).*

5.3.2.2 Likelihood

Concerning the representation of pedestrian, parameters associated to cluster k is the center $\widetilde{D}_t^k$ and covariance $\widetilde{V}_t^k$ of the deformable feature, which are computed from all the deformable configurations within the cluster up to the current epoch. Given a tracklet $\mathbf{x}_{t,i}$, the likelihood of the tracklet belonging to the concerned cluster is

$$f\left(\mathbf{x}_{t,i}|\phi_{t,k}, \mathbf{x}_{t,k,\cdot}\right) \propto s\left(A_{t,i}^{head}, A_{t,k,m}^{tail}\right) s\left(A_{t,i}^{tail}, A_{t,k,n}^{head}\right) p\left(\widetilde{D}_{t,i}, \widetilde{V}_{t,i}; \phi_{t,k}\right), \tag{5.7}$$

where $\mathbf{x}_{t,k,\cdot}$ is the set of observations associated with $\phi_{t,k}$, $A_{t,k,m}^{tail}$ and $A_{t,k,n}^{head}$ are visual representation of the tail tracklet $\mathbf{x}_{t,k,m}$ and head tracklet $\mathbf{x}_{t,k,n}$ which are closest to $\mathbf{x}_{t,i}^{head}$ and $\mathbf{x}_{t,i}^{tail}$ respectively regarding temporal distance.

$s\left(\bullet, \bullet\right)$ is the appearance similarity between the head of one tracklet and the tail of another tracklet by considering both the holistic bounding box and the parts. To be specific, it is computed as:

$$s\left(A_1, A_2\right) = l\left(\mathbf{h}_1^B, \mathbf{h}_2^B\right) + \frac{1}{8}\sum_{j=1}^{8} l\left(\mathbf{h}_{1,j}^P, \mathbf{h}_{2,j}^P\right) , \tag{5.8}$$

where the first term corresponds to appearance of the holistic bounding box, the second term considers the appearance information of the set of parts and $\mathbf{h}$ is the feature vector. B and P abbreviate "box" and "part" respectively. $l\left(\bullet, \bullet\right)$ is a similarity measurement between two feature histograms. The similar formula in Equation 5.4 is adopted as the measurement.

$p\left(\widetilde{D}_{t,i}, \widetilde{V}_{t,i}; \phi_{t,k}\right)$ is the likelihood of $\mathbf{x}_{t,i}$ given cluster parameter $\phi_{t,k}$ considering the deformable configuration of parts. It is formulated as

$$p\left(\widetilde{D}_{t,i}, \widetilde{V}_{t,i}; \phi_{t,k}\right) \propto \exp\left(-d\left(\widetilde{D}_{t,i}, \widetilde{V}_{t,i}, \widetilde{D}_{t,k}, \widetilde{V}_{t,k}\right)\right) , \tag{5.9}$$

where $d\left(\widetilde{D}_{t,i}, \widetilde{V}_{t,i}, \widetilde{D}_{t,k}, \widetilde{V}_{t,k}\right)$ is the distance between the cluster and the concerned tracklet considering the deformable configuration as the following

$$\begin{aligned} & d\left(\widetilde{D}_{t,i}, \widetilde{V}_{t,i}, \widetilde{D}_{t,k}, \widetilde{V}_{t,k}\right) \\ & = \frac{1}{7}\sum_{j=1}^{7}\left(\widetilde{D}_{t,i,j} - \widetilde{D}_{t,k,j}\right)^T \left(\frac{\widetilde{V}_{t,i,j} + \widetilde{V}_{t,k,j}}{2}\right)^{-1} \left(\widetilde{D}_{t,i,j} - \widetilde{D}_{t,k,j}\right) . \end{aligned} \tag{5.10}$$

Here the deformable configuration is computed in a part-wise fashion. Note that the number of parts under consideration is 7, rather than 8 in Equation 5.8. This is because the head part is adopted as the anchor, which is not taken into account. It is also worthy to note that in Equation 5.7 the first two terms locally account for the appearance information and the third term globally considers the deformable information in terms of temporal span.

5.3.3 Cannot Links & Must Links

The first temporal exclusion constraint, that at most one object can be assigned to each trajectory, is modeled by the exclusive property of cluster membership of each object detection.

The second one, i.e. one trajectory cannot be assigned more than one detection within the same frame, is modeled by a cannot-link constraint. If two tracklets in the same epoch overlap temporally, they cannot have the same cluster label, i.e. they cannot be linked to be part of an identical object. The set of cannot-link constraints in epoch t is represented as

$$\mathbf{CNL}_t = \left\{ \left(\mathbf{x}_{t,i}, \mathbf{x}_{t,j} \right) | z_{t,i} \neq z_{t,j} \right\} , \tag{5.11}$$

where $z_{t,i}$ and $z_{t,j}$ are cluster membership indicators of tracklets $\mathbf{x}_{t,i}$ and $\mathbf{x}_{t,j}$ which overlap in epoch t. The partitioning of the video into epochs may split tracklets into segments. The must-link constraints between tracklets from adjacent epochs are used to connect these. This kind of constraints for epoch t is given by

$$\mathbf{ML}_t = \left\{ \left(\mathbf{x}_{t,i}, \mathbf{x}_{t-1,j} \right) | z_{t,i} = z_{t-1,j} \right\} . \tag{5.12}$$

Figure 5.2 shows some examples of the cannot links and the must links. Note that there are no must-link constraints for the first epoch.

5.3.4 Temporal Damping

Temporal effects should be considered during the process of clustering observations. Let me illustrate this by the Chinese Restaurant Process (CRP) representation. In CRP, prior knowledge only depends on the existing number of customers belonging to the table. However, in the multi-object tracking problem, this is not sufficient. When searching the cluster to which a tracklet belongs, the temporal gap between this new tracklet and existing clusters should be additionally taken into consideration. For example, considering a cluster which is temporally distant from the given tracklet, the probability that the tracklet is assigned to this cluster is low, even if there are already many tracklets assigned to this cluster. In other words, the assignment prior probability should decay with the temporal gap between a cluster and the tracklet. Considering a tracklet at epoch t and supposing some clusters already exist, the number of members belonging to cluster k at epoch τ is damped by a weight, similar to [Zhu et al., 2005], as:

$$n_{k,\tau} = \sum_j \delta \left(z_{\tau,j} - k \right) \exp \left(-\eta \left(t - \tau \right) \right) , \quad \tau < t , \tag{5.13}$$

where z is the cluster membership indicator and η is a damping factor.

5.4 Inference

Assuming there are N tracklets $\mathbf{x}_{1:N}$ and T epochs, let me denote the observations in epoch t as $\mathbf{x}_{1:N_t}$ and the corresponding estimations as $\theta_{1:N_t}$. The first-order relation is considered in the proposed model, i.e. the first epoch is a normal DPMM and subsequent DPMMs are closely related to the previous DPMM. The posterior probability is written as

$$\begin{aligned}
&P\left(\theta_{1:N}|\mathbf{x}_{1:N},\alpha,G_0,\mathbf{CNL},\mathbf{ML}\right)\\
&=P\left(\theta_{1:N_1}|\mathbf{x}_{1:N_1},\alpha,G_0,\mathbf{CNL}_1\right)\times\\
&\prod_{t=2}^{T}P\left(\theta_{1:N_t}|\mathbf{x}_{1:N_t},\theta_{1:N_{t-1}},\alpha,G_0,\mathbf{CNL}_t,\mathbf{ML}_t\right)\\
&\propto P\left(\theta_{1:N_1}|\mathbf{x}_{1:N_1},\alpha,G_0,\mathbf{CNL}_1\right)\times\\
&\prod_{t=2}^{T}f\left(\mathbf{x}_{1:N_t}|\theta_{1:N_t}\right)P\left(\theta_{1:N_t}|\theta_{1:N_{t-1}},\alpha,G_0,\mathbf{CNL}_t,\mathbf{ML}_t\right),
\end{aligned} \tag{5.14}$$

where $f(\cdot)$ is the likelihood function, $P\left(\theta_{1:N_t}|\theta_{1:N_{t-1}},\alpha,G_0,\mathbf{CNL}_t,\mathbf{ML}_t\right)$ encodes the evolution over time.

Computing the posterior is intractable, thus Gibbs sampling is employed for inference [Ahmed and Xing, 2008], introducing the latent cluster indicator variable of $\mathbf{x}_{t,i}$ as $z_{t,i}$. For each epoch, the input are tracklets in this epoch and existing clusters up to this epoch; the output are the clusters after being assigned tracklets in the current epoch. The state of the sampler contains both the cluster indicators $z_{t,\cdot}$ of all observations and the states $\phi_{t,\cdot}$ of all clusters. Two steps are iterated: (1) given the current states of clusters, sample cluster indicators for all the observations, (2) given all cluster indicators of observations, update the states of clusters.

(1) Enforcing must-link and cannot-link constraints, cluster indicators are sampled as follows:

(a) if $\mathbf{x}_{t,i}$ is a member of the must-link set, i.e. $\mathbf{ML}_t$, the cluster indicator of $\mathbf{x}_{t,i}$ should be identical to that of its must-link counterpart $\mathbf{x}_{t-1,j}$;

(b) else the cluster indicator of $\mathbf{x}_{t,i}$ is sampled according to the conditional posterior as $P\left(z_{t,i}|z_{1:t-1}, z_{t,-i}, \mathbf{x}_{t,i}, \mathbf{x}_{t,k,\cdot}, \phi_{t,1:k}, \alpha, G_0\right)$. This is analogous to standard DPMM sampling, thus this probability can be written as:

$$P\left(z_{t,i} = k|\cdots\right) \propto \frac{n_{k,1:t-1} + n_{k,t,-i}}{N_{1:t-1} + N_t + \alpha - 1} f\left(\mathbf{x}_{t,i}|\phi_{k,t}, \mathbf{x}_{t,k,\cdot}\right) , \tag{5.15}$$

where $n_{k,1:t-1} = \sum_{\tau=1}^{t-1} n_{k,\tau}$ is the number of past observations with cluster indicator k, $n_{k,t,-i} = \sum_{j\in-i} \delta\left(z_{t,j} - k\right)$, $N_{1:t-1} = \sum_{k\in\mathbf{K}} n_{k,1:t-1}$, $\mathbf{K}$ is the set of indicators of existing clusters.

The emergence of a new cluster is also allowed with probability

$$P\left(z_{t,i} = new\ cluster|\cdots\right) \propto \frac{\alpha}{N_{1:t-1} + N_t + \alpha - 1} \int_\theta f\left(\mathbf{x}_{t,i}|\theta\right) dG_0(\theta) . \tag{5.16}$$

(c) due to the cannot-link set, if $\mathbf{x}_{t,i}$ belongs to $\mathbf{CNL}_t$, then $z_{t,i}$ must be different from all its cannot-link counterparts. Thus $z_{t,i}$ should be sampled from the indicators of all existing clusters excluding those of all $\mathbf{x}_{t,i}$'s cannot-link counterparts. According to this, when computing the probability, $\phi_{t,1:k}$ is replaced with $\phi_{t,1:k}\backslash\phi_{t,-i}$, where $\phi_{t,-i}$ is the set of clusters which $\mathbf{x}_{t,i}$'s cannot-link counterparts belong to, and \ means the set difference operation.

(2) Given cluster indicators, cluster parameters are updated by estimating $P\left(\phi_{t,k}|z_{t,\cdot}, \mathbf{x}_{t,\cdot}, \phi_{t-1,k}\right)$. Since a cluster is conditionally independent from other clusters given the cluster indicators, this probability can be written as $P\left(\phi_{t,k}|\mathbf{x}_{t,k,\cdot}, \phi_{t-1,k}\right) \propto G_0\left(\phi_{t,k}\right) f\left(\mathbf{x}_{t,k,\cdot}|\phi_{t,k}\right) P\left(\phi_{t,k}|\phi_{t-1,k}\right)$, where $\mathbf{x}_{t,k,\cdot}$ is the set of observations associated with $\phi_{t,k}$ and $f\left(\mathbf{x}_{t,k,\cdot}|\phi_{t,k}\right)$ is the likelihood. $P\left(\phi_{t,k}|\phi_{t-1,k}\right)$ encodes the cluster parameter dynamics, which is inversely proportional to the distance between the two Gaussian distributions corresponding to $\phi_{t,k}$ and $\phi_{t-1,k}$. To be more specific, it is with the following form:

$$\phi_{t,k}|\phi_{t-1,k} \sim \mathcal{N}\left(\phi_{t-1,k}, \gamma\mathbf{I}\right) . \tag{5.17}$$

Next I sample to update the parameters of the cluster.

These two steps are carried out iteratively in each epoch, resulting in observations with the same cluster indicator being linked into one trajectory, which corresponds to one object.

Table 5.2: *Multi-object tracking results. The proposed method is compared with GMOT-MTL [Luo and Kim, 2013] and GMOT-BLP [Luo et al., 2014a], in terms of MOTA, MOTP and MT values. Results of the proposed method are in the shaded columns. The arrows next to the metrics indicate the direction of better performance, e.g. ↑ means larger values are better.*

Sequence	*MOTA*↑				*MOTP*↑				*MT*↑			
	GMOT-MTL	GMOT-ATD	GMOT-BLP	GMOT-ATD	GMOT-MTL	GMOT-ATD	GMOT-BLP	GMOT-ATD	GMOT-MTL	GMOT-ATD	GMOT-BLP	GMOT-ATD
Zebra	0.78	0.66	0.74	0.74	0.67	0.69	0.68	0.68	0.44	0.43	0.58	0.61
Crab	0.39	0.47	0.50	0.50	0.60	0.62	0.69	0.69	0.10	0.15	0.21	0.25
Antelope	0.37	0.36	0.62	0.43	0.63	0.68	0.71	0.71	0.37	0.38	0.69	0.74
Goose	0.80	0.87	0.94	0.90	0.60	0.70	0.65	0.65	0.64	0.71	0.79	0.79
Sailing	0.55	0.54	0.82	0.82	0.68	0.69	0.64	0.64	0.25	0.50	0.83	0.83
Hockey	0.80	0.80	0.77	0.76	0.69	0.69	0.74	0.73	0.68	0.71	0.61	0.68
Overall	0.51	0.37	0.64	0.61	0.64	0.66	0.68	0.68	0.34	0.38	0.51	0.55

Table 5.3: *Multi-object tracking results. The proposed method is compared with GMOT-MTL [Luo and Kim, 2013] and GMOT-BLP [Luo et al., 2014a], in terms of ML, FM and IDS values. Results of the proposed method are in the shaded columns. The arrows next to the metrics indicate the direction of better performance, e.g. ↓ means larger values are better.*

Sequence	*ML*↓				*FM*↓				*IDS*↓			
	GMOT-MTL	GMOT-ATD	GMOT-BLP	GMOT-ATD	GMOT-MTL	GMOT-ATD	GMOT-BLP	GMOT-ATD	GMOT-MTL	GMOT-ATD	GMOT-BLP	GMOT-ATD
Zebra	0.29	0.30	0.25	0.25	36	27	30	26	6	3	7	1
Crab	0.71	0.68	0.69	0.69	243	134	205	163	114	77	63	15
Antelope	0.37	0.37	0.18	0.16	33	28	54	32	19	16	31	6
Goose	0.07	0.07	0.04	0.04	52	38	36	19	28	27	33	12
Sailing	0.08	0.08	0.08	0.08	99	85	45	40	33	11	12	8
Hockey	0.11	0.11	0.14	0.11	27	23	24	10	17	9	20	3
Overall	0.41	0.39	0.34	0.34	490	335	394	290	217	143	166	45

5.5 Experiments

In this section, experiment settings, metrics, and results of DPMM applications based on the described two kinds of visual representations are reported. Note that, comparison in all experiments is based on the same detection results, either from the previous chapter or from the DPM model.

5.5.1 Settings

Videos are divided into epochs which are composed of approximate $50 - 200$ frames, depending on the length of the video. I set the dictionary dimension to 50, η to 0.2 and γ in Equation 5.17 to 0.01 in all experiments. In the inference stage, for each epoch Gibbs sampling is run for 500 iterations and results are reported after the last iteration.

5.5.2 MOT by DPMM-SP

5.5.2.1 Data Sets

The proposed algorithm is applied to two problems, (1) generic multi-object tracking [Luo and Kim, 2013, Luo et al., 2014a, Zhao et al., 2012], where multiple objects of any type are detected and tracked and (2) multi-pedestrian tracking, requiring the output of an off-line trained pedestrian detector as input. For the first problem, I employ public six data sets from [Luo et al., 2014a] named *Zebra, Crab, Goose, Hockey, Sailing* and *Antelope*. For the second problem, I use the public *ETHMS* and *TUD Stadtmitte* data sets.

5.5.2.2 Results

In this part, the results of tracking multiple rigid objects based on the super-pixel visual representation are represented. The experiments are conducted in three parts. In the first part

I compare it with existing sequential approaches [Luo and Kim, 2013, Luo et al., 2014a] in solving the GMOT problem. The second part compares the algorithm with several state-of-the-art data association algorithms [Pirsiavash et al., 2011, Xing et al., 2009] using the same detection results and visual representation. I additionally conduct the experiment of tracking multiple pedestrians based on the super-pixel visual representation, which is intended to check the effectiveness of the dynamic clustering solution. The results are reported in the third part, and are compared with those of other approaches to multi-pedestrian tracking [Pellegrini et al., 2009, Zhang et al., 2008, Milan et al., 2013b, Leal-Taixé et al., 2014].

Part 1 – Comparison with generic multi-object trackers. In this part, I compare the automatic topic discovery (**GMOT-ATD**) algorithm with two state-of-the-art generic multi-object trackers, **GMOT-MTL** [Luo and Kim, 2013] and **GMOT-BLP** [Luo et al., 2014a]. For fairness I use the same detection results as used in the methods which are compared with, allowing direct comparison of the association performance. The results are shown in Table 5.2 and 5.3. The proposed **GMOT-ATD** algorithm is based on the same detection results from the corresponding counterparts. The results of **GMOT-MTL** and **GMOT-BLP** are quoted from [Luo and Kim, 2013] and [Luo et al., 2014a] respectively. Compared with **GMOT-MTL**, the proposed algorithm reduces the quantity of FM and IDS by 32% and 34%. Compared with **GMOT-BLP**, the FM and IDS values are reduced by 26% and 73% respectively. This means that the proposed algorithm tracks objects more consistently in the test sequences. Note however, that the proposed algorithm is a batch algorithm while both **GMOT-MTL** and **GMOT-BLP** process the data sequentially. The next set of experiments therefore directly compares with batch data association methods.

Part 2 – Comparison with data association algorithms. In this section I compare the proposed method with a number of data association algorithms, including (1) **DA-H**: the Hungarian algorithm [Xing et al., 2009], (2) **DA-DP**: dynamic programming in network flow [Pirsiavash et al., 2011], (3) **DA-SSP**: successive shortest path in network flow [Pirsiavash et al., 2011], (4) **BL**: a baseline method of the proposed algorithm without temporal dynamics, i.e. the video sequence is treated as a single document without division into epochs. This can be viewed as the application of standard DPMM to the GMOT problem. For fairness, all algo-

Table 5.4: *Data association comparison, in terms of MOTA, MOTP and MT values. The best results are shown in bold.*

Sequence	*MOTA*↑					*MOTP*↑					*MT*↑				
	DA-H	DA-DP	DA-SSP	BL	GMOT-ATD	DA-H	DA-DP	DA-SSP	BL	GMOT-ATD	DA-H	DA-DP	DA-SSP	BL	GMOT-ATD
Zebra	**0.75**	0.72	0.72	0.74	0.74	0.68	0.68	0.68	0.68	0.68	0.59	0.55	0.54	0.60	**0.61**
Crab	**0.50**	0.48	0.48	**0.50**	**0.50**	0.69	0.69	0.69	0.69	0.69	0.24	0.19	0.19	**0.25**	**0.25**
Antelope	0.41	**0.65**	**0.65**	0.43	0.44	0.71	**0.72**	**0.72**	0.71	0.71	**0.75**	0.63	0.63	0.72	0.74
Goose	0.88	0.86	0.85	**0.90**	**0.90**	0.65	0.65	0.65	0.65	0.65	**0.79**	0.64	0.68	**0.79**	**0.79**
Sailing	**0.82**	0.81	0.81	**0.82**	**0.82**	0.64	0.64	0.64	0.64	0.64	0.83	0.83	0.83	0.83	0.83
Hockey	0.78	0.72	0.71	**0.80**	0.76	**0.74**	**0.74**	**0.74**	0.73	0.73	0.64	0.54	0.54	**0.68**	**0.68**
Overall	0.61	**0.63**	0.62	0.62	0.61	0.68	0.68	0.68	0.68	0.68	0.54	0.47	0.46	0.54	**0.55**

Table 5.5: *Data association comparison, in terms of ML, FM and IDS values. The best results are shown in bold.*

Sequence	*ML*↓					*FM*↓					*IDS*↓				
	DA-H	DA-DP	DA-SSP	BL	GMOT-ATD	DA-H	DA-DP	DA-SSP	BL	GMOT-ATD	DA-H	DA-DP	DA-SSP	BL	GMOT-ATD
Zebra	**.25**	.35	.35	**.25**	**.25**	28	32	31	27	**26**	3	2	7	3	**1**
Crab	**.69**	.70	.70	**.69**	**.69**	170	168	166	168	**163**	27	31	30	28	**15**
Antelope	**.15**	.27	.27	**.15**	.16	36	33	**32**	37	**32**	14	10	10	25	**6**
Goose	**.03**	.25	.32	.04	.04	34	31	29	25	**19**	25	20	18	14	**12**
Sailing	.08	.08	.08	.08	.08	42	45	44	**40**	**40**	10	9	**8**	**8**	**8**
Hockey	.14	.18	.18	**.11**	**.11**	12	11	**10**	12	**10**	11	7	6	6	**3**
Overall	0.34	0.41	0.42	**0.33**	0.34	322	320	312	309	**290**	90	79	73	84	**45**

Table 5.6: *Multi-person tracking results compared with other state-of-the-art methods in terms of MT, ML, FM and IDS values. The best results are shown in bold.*

Sequence	*TUD-Stadtmitte*		*ETHMS*				
	[Milan et al., 2013b]	GMOT-ATD	[Pellegrini et al., 2009]	[Zhang et al., 2008]	[Milan et al., 2013b]	[Leal-Taixé et al., 2014]	GMOT-ATD
MT↑	0.400	**0.900**	0.516	0.556	0.664	**0.720**	0.589
ML↓	0	0	0.056	0.062	0.082	**0.047**	0.073
FM↓	**13**	16	206	178	**69**	85	156
IDS↓	15	**13**	77	138	**57**	71	103

rithms are given the same detection results from [Luo et al., 2014a]. The results of **DA-DP** and **DA-SSP** are obtained using the code from [Pirsiavash et al., 2011].

Results in Table 5.4 and 5.4 indicate that (1) generally **DA-H** tends to achieve good MT and ML values, meaning it is able to track objects more completely. On the other hand, the performance in terms of FM and IDS is worse than the performance of proposed method; (2) **DA-DP** and **DA-SSP** obtain good FM and IDS values, indicating that they can track objects more robustly and consistently. **DA-SSP** achieves slightly better FM and IDS than **DA-DP**. However, compared with **DA-H**, they tend to ignore parts of trajectories, thus MT and ML values are worse than those of **DA-H**; (3) compared with **DA-H**, **BL** has similar MT and ML values while achieving better FM and IDS values, showing the effectiveness of applying a DPMM; (4) the proposed method (**GMOT-ATD**) achieves the best performance. Compared with **BL**, it further reduces the IDS and FM values.

Some exemplar images of the results on the six data sets for GMOT problem are shown from Figure 5.5 to Figure 5.10.

Part 3 – Comparison with pedestrian trackers. In this part, I evaluate the proposed method on the multiple pedestrian tracking problem where the raw detection results are those

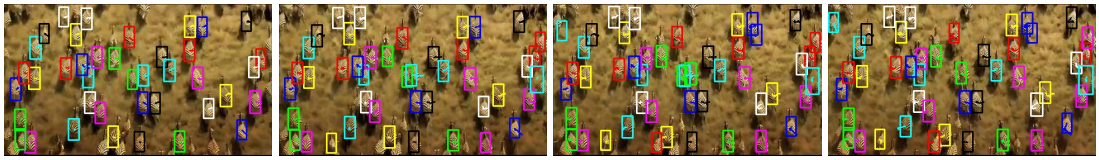

Figure 5.5: *Exemplar qualitative results of GMOT-ATD. They are shown on frames excerpted from the sequence of Zebra. Different colors stand for different objects. As only 8 colors are employed for visualization, some different objects may have the same color. This representation is also applicable to the following figures.*

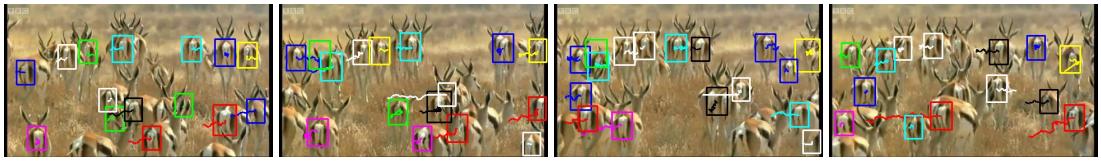

Figure 5.6: *Exemplar qualitative results of GMOT-ATD. They are shown on frames excerpted from the sequence of Antelope. Different colors stand for different objects. As only 8 colors are employed for visualization, some different objects may have the same color.*

Figure 5.7: *Exemplar qualitative results of GMOT-ATD. They are shown on frames excerpted from the sequence of Crab. Different colors stand for different objects. As only 8 colors are employed for visualization, some different objects may have the same color.*

in [Milan et al., 2013b]. I compare the results with those in [Pellegrini et al., 2009, Zhang et al., 2008, Milan et al., 2013b, Leal-Taixé et al., 2014]. Pellegrini et al. develop a sophisticated dynamic model based on social forces during association [Pellegrini et al., 2009]. Zhang et al. cast data association as finding the min-cost in network flow [Zhang et al., 2008]. Milan et al. adopt a CRF model for data association [Milan et al., 2013b].

Qualitative results (result images) are shown in Figure 5.11, 5.12 and 5.13. The *ETHMS* data set is composed of two sub sets. The results are shown separately in Figure 5.12 and Figure 5.13. Please notice the green bounding box and the blue one in Figure 5.11. These two pedestrians involve in occlusion. The proposed method maintains the identities correctly in the

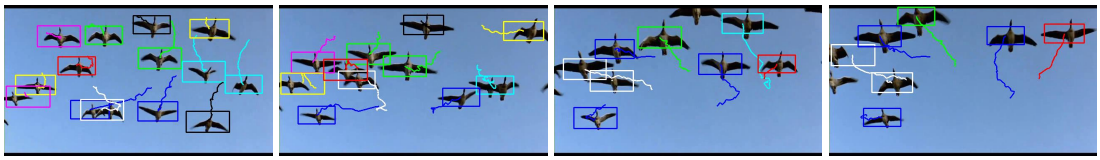

Figure 5.8: *Exemplar qualitative results of GMOT-ATD. They are shown on frames excerpted from the sequence of Goose. Different colors stand for different objects. As only 8 colors are employed for visualization, some different objects may have the same color.*

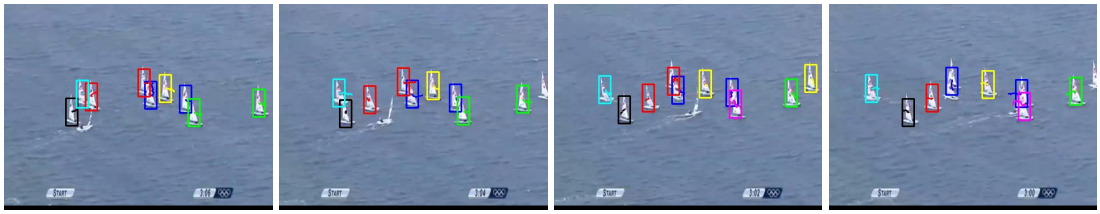

Figure 5.9: *Exemplar qualitative results of GMOT-ATD. They are shown on frames excerpted from the sequence of Sailing. Different colors stand for different objects. As only 8 colors are employed for visualization, some different objects may have the same color.*

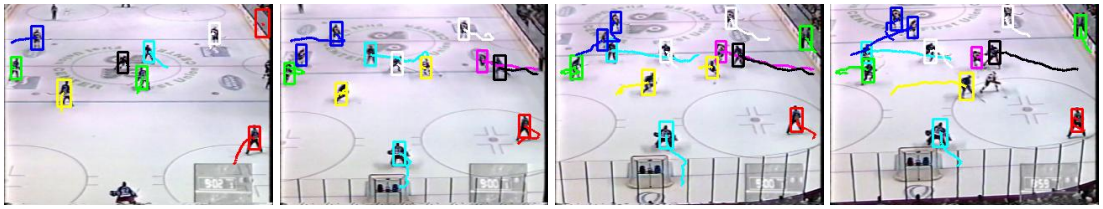

Figure 5.10: *Exemplar qualitative results of GMOT-ATD. They are shown on frames excerpted from the sequence of Hockey. Different colors stand for different objects. As only 8 colors are employed for visualization, some different objects may have the same color.*

occlusion. However, the proposed approach fails in some cases. For example, in Figure 5.13, the red bounding box and white bounding box correspond to an identical person. The proposed method fails to link them due to miss detections. This case is expected to be solved if some sophisticated observation models are included.

Quantitative results are shown in Table 5.6. On the *TUD-Stadtmitte* data set, the proposed method achieves better ML and IDS performance while obtaining worse FM performance. On the *ETHMS* data set, the results of the proposed method are comparable to those of [Pellegrini

et al., 2009] and [Zhang et al., 2008] but worse than those of [Milan et al., 2013b] and [Leal-Taixé et al., 2014], which are all methods tailored to the task of pedestrian tracking.

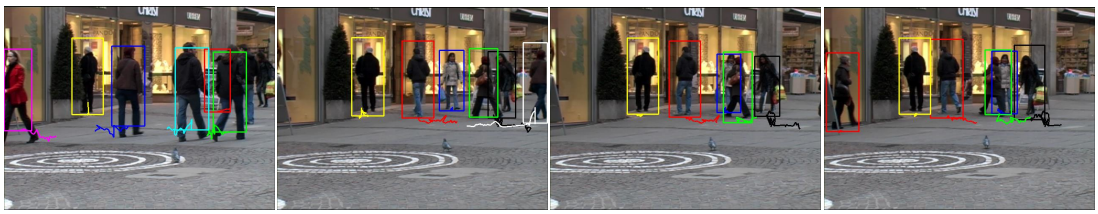

Figure 5.11: *Exemplar qualitative results of the proposed approach on the TUD-stadtmitte data set.*

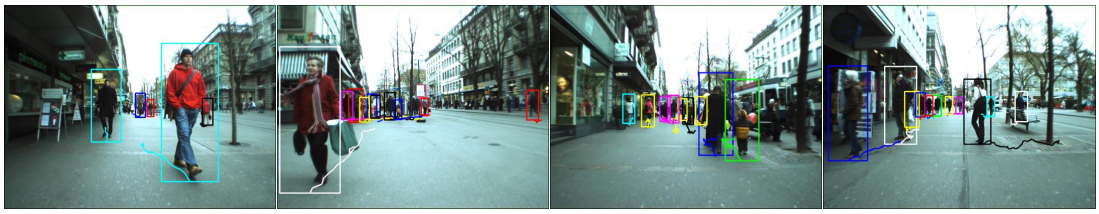

Figure 5.12: *Exemplar qualitative results of the proposed approach on the ETHMS data set.*

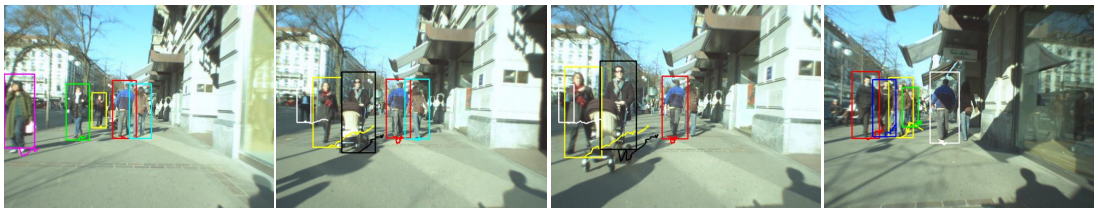

Figure 5.13: *Exemplar qualitative results of the proposed approach on the ETHMS data set.*

The reason is supposed to be that although the same raw detection hypotheses are taken as input, the proposed approach does not include sophisticated appearance or motion models. In contrast, the motion model in [Pellegrini et al., 2009] takes the effect of pedestrians in a group into account, which is helpful in reducing IDS in the case of occlusion. The method in [Zhang et al., 2008] includes a model named Explicit Occlusion Model (EOM) which especially handles occlusion by generating occlusion hypotheses and integrating them in the network. Besides considering exclusivity constraints, a motion model based on angular velocity is taken into consideration in [Milan et al., 2013b]. [Leal-Taixé et al., 2014] achieves the

best MT and ML performance as a result of their contextual motion model, which is able to recover more trajectory components even in the case of missed detections, by learning a dictionary of interaction features among objects. In the proposed method, only the plain but general super-pixel representation is considered for appearance modeling. The super-pixel representation could perform well in representing rigid objects in the first application - generic multiple object tracking, while inevitably suffers from clutters from backgrounds in representing non-rigid objects such as pedestrians. On the other hand, the proposed approach can serve as a basic model to include more sophisticated appearance or motion models.

It is also observed that, in general the number of topics discovered (individual objects) is larger than the true number of objects (groundtruth). I suspected that, it should be due to issues like occlusion or missed detections. These kinds of issues would result in fragmentation in trajectories and consequently more number of discovered topics.

In the next section, I would demonstrate the results of a model called $(\text{DPM})^2$, which includes visual representation developed by specifically considering the non rigidness of pedestrian.

5.5.3 MOT by $(\text{DPM})^2$

Results of multiple pedestrian tracking indicate that, performance of pedestrian tracking partly relies on the representation model. This motivates me to develop a better visual representation model (Section 5.3.2) than the plain super-pixel representation. In this part, I report the results of tracking multiple pedestrians based on the proposed $(\text{DPM})^2$ model.

5.5.3.1 Data Sets

Two data sets are employed in the experiment. The first one is *TUD-Stadtmitte*, the same as the one used in the last part of the previous section. The other one is named *ParkingLot*. The reason of the usage of this data set is that it is employed in [Izadinia et al., 2012b, Shu et al., 2012],

Table 5.7: *Multi-person tracking results compared between DPMM-SP and (DPM)^2 in terms of MT, ML, FM and IDS values on the TUD-Stadtmitte data set. The best results are shown in bold.*

Method	*MT*↑	*ML*↓	*FM*↓	*IDS*↓
DPMM-SP	0.900	0	16	13
(DPM)^2	0.900	0	**9**	**11**

which handle the problem of multiple pedestrian tracking by the employment of Deformable Part Model. Thus, by adopting this data set, I can directly compare the proposed method with the other two [Izadinia et al., 2012b, Shu et al., 2012].

5.5.3.2 Results

In this section, the results are presented in two parts. In the first part, I show the comparison between **(DPM)**^{2} and **DPMM-SP** for the task of multiple pedestrian tracking on the *TUD-Stadtmitte* data set. In the second part, the comparison between the performance of the proposed **(DPM)**^{2} and some other state-of-the-art methods for multi-pedestrian tracking is presented.

Part 1 – Comparison between DPMM-SP and (DPM)^2. As shown in Table 5.7, **(DPM)**^{2} outperforms **DPMM-SP** on the data set of *TUD-Stadtmitte*. More specifically, the values of MT and ML remain the same while the values of FM and IDS decrease by 43.8% and 15.4%. The improvement of performance is supposed to result from the DPM representation in (DPM)^2 as these two methods differ only from the representation of pedestrians.

Part 2 – Comparison between (DPM)^2 and other pedestrian trackers. Table 5.8 shows the comparison between the proposed method **(DPM)**^{2} and the other two state-of-the-art methods, which are termed as **(MP)**^{2} [Izadinia et al., 2012b] and **PMT** [Shu et al., 2012]. The results reveal that, 1) **DPM** outperforms the **PMT** method [Shu et al., 2012] and 2) except the value of DP, **(DPM)**^{2} does not outperform the **(MP)**^{2}, but the values could still be comparable.

The comparison above suggests that, 1) there is advance of automatic topic discovery over specific SVM classifier for individual person which is adopted in **PTM** [Shu et al., 2012] and

Table 5.8: *Multi-pedestrian tracking results on the parking lot data set, compared with other state-of-the-art methods in terms of MOTA, MOTP, DA and DP values. The best results are shown in bold.*

Method	MOTA ↑	MOTP ↑	DA ↑	DP ↑
PMT [Shu et al., 2012]	0.793	0.741	0.798	0.742
$(MP)^2$ [Izadinia et al., 2012b]	**0.889**	**0.775**	**0.965**	0.936
$(DPM)^2$	0.830	0.751	0.886	**0.959**

2) approach depending on only appearance information can hardly outperform the counterparts which employ not only appearance information but also information such as motion or occlusion [Izadinia et al., 2012b]. As mentioned before, motion model or other specific kinds of models can be integrated into the proposed automatic topic discovery framework, while it is not the focus of this section.

5.6 Remarks

This chapter has introduced a topic model for the multi-object tracking problem. Thanks to the DPMM, tracking management is addressed by dynamical clustering. Along with the introduced cannot-link constraints, the exclusivity constraints are handled naturally. The dynamics of object appearance variation is modeled by segmenting the video into temporal epochs. As a basis, two types of visual representation methods have been developed and integrated into the proposed dynamic clustering procedure to track rigid and non-rigid objects. Experiments on public data sets show the advantages of topic discovery method over sequential solutions and other data association ones.

As experimental results show, integrating more advanced models like motion or occlusion models could further improve the performance. Future work may include the integration of such models.

CHAPTER 6

CONCLUSION AND FUTURE WORK

This thesis proposes several approaches in handling the problem of generic multiple object tracking. To begin with, a sequential method based on multi-task learning is proposed and the tracking of multiple objects is treated as multiple tasks on top of a generic object detector. Then, another sequential approach based on label propagation is presented. In this approach, detection and tracking are re-formulated as the propagation of class label and object label respectively. Finally, I introduce a dynamic clustering strategy named as automatic topic discovery. In contrast with traditional data-association solutions, the proposed dynamic clustering method demonstrates advantages from several perspectives in solving the problem of GMOT.

Details of each work are summarized in Section 6.1, and a discussion about the relationship among the proposed three approaches is conducted in Section 6.2. In Section 6.3, a discussion about the limitations of each work is presented, followed by some ideas of the future directions.

6.1 Summary

Chapter 3 presents a sequential solution to the problem of generic multiple object tracking. Following the popular tracking-by-detection strategy, the problem is decomposed into two main tasks – detection and tracking. These two tasks are formulated under the framework of Multiple Task Learning (MTL). In particular, a binary detector is learned to detect objects in images while multiple trackers are learned on top of the detector by MTL to trace detected objects in the subsequent frames. The detector is utilized to anchor the multiple trackers and the multiple trackers are jointly learned by sharing common features. The proposed method outperforms the state-of-the-art methods on several benchmark data sets.

Regarding a concerned video as a cuboid, a bi-label propagation method is proposed in Chapter 4 to track objects of the same type. The term "bi-labels" refers to a binary class label for detection and individual object labels for tracking. To propagate class label, I employ the clustered Multiple Task Learning (cMTL) while enforcing the spatio-temporal consistency. The proposed idea shows considerable improvement when the scale of training data is limited. To track objects, object labels are propagated from trajectories to detections based on affinity computed with regard to appearance, motion and context. Experiments on public and challenging new data sets show that the bi-label propagation method improves over the current state of the art.

Chapter 5 presents a new approach to multi-object tracking by semantic topic discovery. Frame-by-frame detections are dynamically clustered into topics which are treated as objects. During the clustering procedure, the Dirichlet Process Mixture Model (DPMM) is applied. The tracking problem is transformed into a topic-discovery task for which the video sequence is treated analogously to a document. This clustering formulation addresses exclusivity constraints of objects and *cannot-link constraints* without the need of heuristic thresholds. Variation of object appearance is modeled as the dynamics of word co-occurrence and handled via updating the cluster parameters across the sequence in the dynamical clustering procedure. Two kinds of visual representations are introduced and incorporated into the model of auto-

matic topic discovery for tracking of rigid and non-rigid objects respectively. The effectiveness of these two kinds of visual representations are well verified with extensive experiments on several public data sets.

6.2 Relationship between Chapters

The proposed GMOT problem has been handled in three approaches (two sequential methods and one batch method), and experimental results have shown the effectiveness of these approaches compared with the state-of-the-art methods.

The first approach (GMOT-MTL) has proved effectiveness of MTL in dealing with the GMOT problem. MTL has been explored to model relevance among related learning tasks from both global and local aspects. However, the online detector based on Laplacian SVM suffers from scarcity of training data. The bi-label propagation framework (GMOT-BLP) overcomes this difficulty by employing multiple detectors and simultaneously learn them. By doing so, the detection module in GMOT-BLP detects objects with higher precision and accuracy. Moreover, more cues (appearance, motion and context) are considered in the tracking stage of GMOT-BLP. This leads to better performance. Both of these methods are sequential ones, which are prone to false positive and false negative hypotheses of detection. Thus, the last method adopts the batch mode for the GMOT problem. By viewing observation globally and clustering the provided detection hypotheses, the batch method especially reduces the value of FM, which counts the times of interruption of trajectories.

6.3 Future Work

The GMOT problem concerned in this thesis is addressed by both sequential and batch methods. The first and the second methods, as sequential approaches, are suitable for the scenario where the video stream arrives sequentially. However, as mentioned before, they may need

some heuristics to handle the false positive and false negative hypotheses in object detection. The batch method views observations globally, thus can achieve better performance. However, it is based on an assumption that detection hypotheses have already been provided. As a trend, batch methods is more promising because object detection has become better and better. Thus, in the future more attention would be paid to batch method for GMOT. Moreover, it is also worthy noticing the following two aspects.

6.3.1 Combination of Sequential and Batch Methods

It is appealing to iteratively combine the detection procedure and data association in a unified framework. Specifically, an object detector is trained by collecting training samples. The detection result of the object detector becomes the input of the data-association tracking algorithm. On the other hand, the output of tracking can provide more information to the object detector, which enables the object detector to improve gradually, so as to boost the overall performance. This iterative procedure enables the detection and data association to benefit each other.

6.3.2 GMOT without Manual Intervention

Existing works of multiple object tracking set primary focus on pedestrians. In this thesis this problem is extended to a more generic scenario of applications. However, it still requires one initial bounding box to specify the concerned object type. In fact, even this initial bounding box is not necessary if the multiple object tracking problem is considered from a more general perspective. Due to the progress of general object detection, objects in images can be detected without any initial labeling and then be tracked. GMOT without manual intervention is challenging yet important to application in reality, thus deserves more research attention.

Appendix

This part illustrates how to optimize the clustered Multiple Task Learning (cMTL) problem with spatio-temporal consistency (Equation 4.5 in Chapter 4).

Optimization of cMTL with spatio-temporal consistency

To be clear, the time index t in the objective function is neglected. It is written as:

$$\mathcal{L}_C(\mathbf{W}) = \underbrace{\alpha tr(\mathbf{W}(\eta\mathbf{I}+\mathbf{M})^{-1}\mathbf{W}^T)}_{regularization} + \underbrace{\frac{\lambda}{2}\|\frac{1}{m}\sum_{i=1}^{m}\mathbf{X}_u^T\mathbf{w}_i - \Psi(P)\|^2}_{spatio-temporal\ consistency} + \underbrace{\sum_{i=1}^{m}\frac{1}{2N_i}\|\mathbf{X}_{l,i}^T\mathbf{w}_i - \mathbf{Y}_i\|^2}_{loss},$$
$$s.t.\ tr(\mathbf{M}) = k, \mathbf{M} \preceq \mathbf{I}, \mathbf{M} \in \mathbf{S}_+^m. \tag{A1}$$

Compared with the original clustered MTL formula in [Zhou et al., 2011b], there is a spare term which represents the spatio-temporal consistency. This term is only involved with $\mathbf{W}$, and it is smooth. The objective function can be rewritten as a combination of a smooth part $\Omega(\mathbf{W})$ and a non-smooth part $\Lambda(\mathbf{W},\mathbf{M})$:

$$\begin{aligned} &\mathcal{L}_C(\mathbf{W}) = \Omega(\mathbf{W}) + \Lambda(\mathbf{W},\mathbf{M}), \\ &s.t.\ tr(\mathbf{M}) = k, \mathbf{M} \preceq \mathbf{I}, \mathbf{M} \in \mathbf{S}_+^m, \end{aligned} \tag{A2}$$

where

$$\Omega(\mathbf{W}) = \sum_{i=1}^{m}\frac{1}{2N_i}\|\mathbf{X}_{l,i}^T\mathbf{w}_i - \mathbf{Y}_i\|^2 + \frac{\lambda}{2}\|\frac{1}{m}\sum_{i=1}^{m}\mathbf{X}_u^T\mathbf{w}_i - \Psi(P)\|^2, \tag{A3}$$

and

$$\Lambda(\mathbf{W},\mathbf{M}) = \alpha tr(\mathbf{W}(\eta\mathbf{I}+\mathbf{M})^{-1}\mathbf{W}^T). \tag{A4}$$

The Accelerated Project Gradient (APG) method is adopted to optimize this problem. Following [Zhou et al., 2011b], the key component of APG solution is to compute a proximal operator as follows:

$$\begin{aligned} &\min_{\mathbf{W}_Z,\mathbf{M}_Z} \parallel \mathbf{W}_Z - \hat{\mathbf{W}}_S \parallel^2 + \parallel \mathbf{M}_Z - \hat{\mathbf{M}}_S \parallel^2, \\ &s.t.\ tr(\mathbf{M}_Z) = k, \mathbf{M}_Z \preceq \mathbf{I}, \mathbf{M}_Z \in \mathbf{S}_+^m, \end{aligned} \tag{A5}$$

where $\hat{\mathbf{W}}_S$ and $\hat{\mathbf{M}}_S$ are two search points, which will be illustrated later.

To obtain $\mathbf{W}_Z$, the following problem needs to be solved:

$$\min_{\mathbf{W}_Z} \parallel \mathbf{W}_Z - \hat{\mathbf{W}}_S \parallel^2 . \tag{A6}$$

It is clear that the optimal solution $\mathbf{W}_Z$ to Equation A6 equals $\hat{\mathbf{W}}_S$.

To obtain $\mathbf{M}_Z$, one needs to solve:

$$\begin{aligned} &\min_{\mathbf{M}_Z} \parallel \mathbf{M}_Z - \hat{\mathbf{M}}_S \parallel^2, \\ &s.t.\ tr(\mathbf{M}_Z) = k, \mathbf{M}_Z \preceq I, \mathbf{M}_Z \in \mathbf{S}_+^m. \end{aligned} \tag{A7}$$

In the following, I show how to construct the search points $\hat{\mathbf{W}}_S$ and $\hat{\mathbf{M}}_S$, and outline the way to solve Equation A7.

Construction of $\hat{\mathbf{W}}_S$

The whole optimization is an iterative process. At the k step, a point $\mathbf{W}_Z^k$ is obtained based on points in the previous two steps $\mathbf{W}_Z^{k-1}$ and $\mathbf{W}_Z^{k-2}$ as $\mathbf{W}_Z^k = (1+\beta)\mathbf{W}_Z^{k-1} - \mathbf{W}_Z^{k-2}$, and the current search point $\hat{\mathbf{W}}_S$ is

$$\hat{\mathbf{W}}_S = \mathbf{W}_Z^k - \frac{1}{\gamma}\nabla\mathcal{L}_C(\mathbf{W}), \tag{A8}$$

where γ is the parameter to control the search step, and $\nabla\mathcal{L}_C(\mathbf{W})$ is the gradient of $\mathcal{L}_C(\mathbf{W})$. Obviously, the gradient of $\mathcal{L}_C(\mathbf{W})$ has two parts, one is from $\Omega(\mathbf{W})$ and the other one is

from $\Lambda(\mathbf{W},\mathbf{M})$. As $\mathbf{W}$ is a combination of all the $\mathbf{w}_i$, the gradient of $\Omega(\mathbf{W})$ can be computed separately. The gradient of $\Omega(\mathbf{W})$ with regard to $\mathbf{w}_i$ is

$$\frac{\partial\Omega(\mathbf{W})}{\partial\mathbf{w}_i} = \frac{1}{N_i}\mathbf{X}_{l,i}(\mathbf{X}_{l,i}^T\mathbf{w}_i - \mathbf{Y}_i) + \frac{\lambda}{m}\mathbf{X}_u(\frac{1}{m}\sum_{i=1}^{m}\mathbf{X}_u^T\mathbf{w}_i - \Psi(P))\,. \tag{A9}$$

As the trace norm is non-smooth, the sub-gradient of $\Lambda(\mathbf{W},\mathbf{M})$ with regard to $\mathbf{W}$ is obtained as $2\alpha(\eta\mathbf{I}+\mathbf{M})^{-1}\mathbf{W}^T$.

Thus the sub-gradient of $\mathcal{L}_C(\mathbf{W})$ with regard to $\mathbf{W}$ is

$$\nabla\mathcal{L}_C(\mathbf{W}) = \frac{\partial\Omega(\mathbf{W})}{\partial\mathbf{W}} + 2\alpha(\eta\mathbf{I}+\mathbf{M})^{-1}\mathbf{W}^T\,, \tag{A10}$$

where $\frac{\partial\Omega(\mathbf{W})}{\partial\mathbf{W}} = [\frac{\partial\Omega(\mathbf{W})}{\partial\mathbf{w}_1}\frac{\partial\Omega(\mathbf{W})}{\partial\mathbf{w}_2}...\frac{\partial\Omega(\mathbf{W})}{\partial\mathbf{w}_m}]$

Construction of $\hat{\mathbf{M}}_S$

Similarly, a point $\mathbf{M}_Z^k$ is obtained based on two points of the previous two steps, $\mathbf{W}_Z^{k-1}$ and $\mathbf{W}_Z^{k-2}$. It is $\mathbf{M}_Z^k = (1+\beta)\mathbf{M}_Z^{k-1} - \mathbf{M}_Z^{k-2}$, and the current search point $\hat{\mathbf{M}}_S$ is

$$\hat{\mathbf{M}}_S = \mathbf{M}_Z^k - \frac{1}{\gamma}\nabla\mathcal{L}_C(\mathbf{M})\,. \tag{A11}$$

Since only $\Lambda(\mathbf{W},\mathbf{M})$ is involved with $\mathbf{M}$, $\Omega(\mathbf{W})$ can be ignored. Thus the sub-gradient of $\mathcal{L}_C(\mathbf{W})$ with regard to $\mathbf{M}$ is

$$\nabla\mathcal{L}_C(\mathbf{M}) = -\alpha\mathbf{W}^T\mathbf{W}(\eta\mathbf{I}+\mathbf{M})^{-1}(\eta\mathbf{I}+\mathbf{M})^{-1}\,. \tag{A12}$$

The following procedure to solve Equation A7 is the same as that in [Zhou et al., 2011b].

REFERENCES

[Achanta et al., 2012] Achanta, R., Shaji, A., Smith, K., Lucchi, A., Fua, P., and Susstrunk, S. (2012). Slic superpixels compared to state-of-the-art superpixel methods. *IEEE Transactions on Pattern Analysis and Machine Intelligence (TPAMI)*, 34(11):2274–2282. 97

[Adam et al., 2006] Adam, A., Rivlin, E., and Shimshoni, I. (2006). Robust fragments-based tracking using the integral histogram. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 798–805. 72

[Ahmed and Xing, 2008] Ahmed, A. and Xing, E. P. (2008). Dynamic non-parametric mixture models and the recurrent chinese restaurant process: with applications to evolutionary clustering. In *SDM*, pages 219–230. SIAM. 90, 104

[Alexe et al., 2010] Alexe, B., Deselaers, T., and Ferrari, V. (2010). What is an object? In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. 47, 48, 56

[Ali et al., 2011] Ali, K., Hasler, D., and Fleuret, F. (2011). FlowBoost-appearance learning from sparsely annotated video. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. 65

[Ali and Shah, 2008] Ali, S. and Shah, M. (2008). Floor fields for tracking in high density crowd scenes. In *European Conference on Computer Vision (ECCV)*, pages 1–14. 26, 27, 33

[Andriyenko and Schindler, 2010] Andriyenko, A. and Schindler, K. (2010). Globally optimal multi-target tracking on a hexagonal lattice. In *European Conference on Computer Vision (ECCV)*, pages 466–479. 18

[Andriyenko and Schindler, 2011] Andriyenko, A. and Schindler, K. (2011). Multi-target tracking by continuous energy minimization. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. 22, 26, 29

[Andriyenko et al., 2012] Andriyenko, A., Schindler, K., and Roth, S. (2012). Discrete-continuous optimization for multi-target tracking. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 1926–1933. 34

[Argyriou et al., 2007] Argyriou, A., Pontil, M., Ying, Y., and Micchelli, C. A. (2007). A spectral regularization framework for multi-task structure learning. In *Advances in Neural Information Processing Systems (NIPS)*. 70

[Badrinarayanan et al., 2010] Badrinarayanan, V., Galasso, F., and Cipolla, R. (2010). Label propagation in video sequences. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. 64

[Belkin et al., 2006] Belkin, M., Niyogi, P., and Sindhwani, V. (2006). Manifold regularization: A geometric framework for learning from labeled and unlabeled examples. *The Journal of Machine Learning Research (JMLR)*, 7:2399–2434. 42, 48, 49

[Benfold and Reid, 2011] Benfold, B. and Reid, I. (2011). Stable multi-target tracking in real-time surveillance video. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. 22, 25

[Berclaz et al., 2006] Berclaz, J., Fleuret, F., and Fua, P. (2006). Robust people tracking with global trajectory optimization. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 744–750. 28

[Berclaz et al., 2009] Berclaz, J., Fleuret, F., and Fua, P. (2009). Multiple object tracking using flow linear programming. In *Proc. IEEE Int. Workshop Perform. Eval. Track. Surveillance*, pages 1–8. 18

[Berclaz et al., 2011] Berclaz, J., Fleuret, F., Turetken, E., and Fua, P. (2011). Multiple object tracking using k-shortest paths optimization. *IEEE Transactions on Pattern Analysis and Machine Intelligence (TPAMI)*, 33(9):1806–1819. 23, 27

[Betke et al., 2007] Betke, M., Hirsh, D. E., Bagchi, A., Hristov, N. I., Makris, N. C., and Kunz, T. H. (2007). Tracking large variable numbers of objects in clutter. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 1–8. 8

[Blei et al., 2006] Blei, D. M., Jordan, M. I., et al. (2006). Variational inference for dirichlet process mixtures. *Bayesian analysis*, 1(1):121–143. 95

[Blei et al., 2003] Blei, D. M., Ng, A. Y., and Jordan, M. I. (2003). Latent dirichlet allocation. *Journal of Machine Learning Research (JMLR)*, 3:993–1022. 93

[Bose et al., 2007] Bose, B., Wang, X., and Grimson, E. (2007). Multi-class object tracking algorithm that handles fragmentation and grouping. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 1–8. 19

[Breitenstein et al., 2009] Breitenstein, M., Reichlin, F., Leibe, B., Koller-Meier, E., and Van Gool, L. (2009). Robust tracking-by-detection using a detector confidence particle filter. In *International Conference on Computer Vision (ICCV)*. xxii, 17, 18, 21, 22, 26, 29, 56, 59, 81

[Brendel et al., 2011] Brendel, W., Amer, M., and Todorovic, S. (2011). Multiobject tracking as maximum weight independent set. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. xxii, 18, 22, 23, 28, 56, 59, 81

[Brostow and Cipolla, 2006] Brostow, G. and Cipolla, R. (2006). Unsupervised bayesian detection of independent motion in crowds. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. 22, 25, 38

[Butt and Collins, 2013a] Butt, A. and Collins, R. (2013a). Multi-target tracking by lagrangian relaxation to min-cost network flow. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 1846–1853. 18, 34

[Butt and Collins, 2013b] Butt, A. A. and Collins, R. T. (2013b). Multi-target tracking by lagrangian relaxation to min-cost network flow. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. 23

[Cannons, 1991] Cannons, K. (1991). A review of visual tracking. Technical Report CSE-2008-07, Dept. Comput. Sci. Eng., York Univ. 19

[Cao and Fei-Fei, 2007] Cao, L. and Fei-Fei, L. (2007). Spatially coherent latent topic model for concurrent segmentation and classification of objects and scenes. In *International Conference on Computer Vision (ICCV)*, pages 1–8. IEEE. 94

[Caruana, 1997] Caruana, R. (1997). Multitask learning. *Machine Learning*, 28(1):41–75. 43

[Chen et al., 2010] Chen, X., Kim, S., Lin, Q., Carbonell, J., and Xing, E. (2010). Graph-structured multi-task regression and an efficient optimization method for general fused lasso manuscript. *arXiv:1005.3579*. 45

[Chen et al., 2009] Chen, X., Pan, W., Kwok, J., and Carbonell, J. (2009). Accelerated gradient method for multi-task sparse learning problem. In *International Conference on Data Mining (ICDM)*. 53

[Choi et al., 2013] Choi, W., Pantofaru, C., and Savarese, S. (2013). A general framework for tracking multiple people from a moving camera. *IEEE Transactions on Pattern Analysis and Machine Intelligence (TPAMI)*, 35(7):1577–1591. 18

[Choi and Savarese, 2010] Choi, W. and Savarese, S. (2010). Multiple target tracking in world coordinate with single, minimally calibrated camera. In *European Conference on Computer Vision (ECCV)*. 22, 23, 32

[Choi and Savarese, 2012] Choi, W. and Savarese, S. (2012). A unified framework for multi-target tracking and collective activity recognition. In *European Conference on Computer Vision (ECCV)*, pages 215–230. 6, 18, 26

[Cong et al., 2011] Cong, Y., Yuan, J., and Liu, J. (2011). Sparse reconstruction cost for abnormal event detection. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 3449–3456. 6

[Dalal and Triggs, 2005] Dalal, N. and Triggs, B. (2005). Histograms of oriented gradients for human detection. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. 3, 20, 26, 55, 63, 74

[Dicle et al., 2013] Dicle, C., Sznaier, M., and Camps, O. (2013). The way they move: Tracking multiple targets with similar appearance. In *International Conference on Computer Vision (ICCV)*, pages 2304–2311. 38

[Duan et al., 2012] Duan, G., Ai, H., Cao, S., and Lao, S. (2012). Group tracking: exploring mutual relations for multiple object tracking. In *European Conference on Computer Vision (ECCV)*. 21

[Dumais et al., 1995] Dumais, S., Furnas, G., Landauer, T., Deerwester, S., Deerwester, S., et al. (1995). Latent semantic indexing. In *Proceedings of the Text Retrieval Conference*. 93

[Ess et al., 2008] Ess, A., Leibe, B., Schindler, K., and Van Gool, L. (2008). A mobile vision system for robust multi-person tracking. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 1–8. 26

[Ess et al., 2009] Ess, A., Leibe, B., Schindler, K., and Van Gool, L. (2009). Robust multiperson tracking from a mobile platform. *IEEE Transactions on Pattern Analysis and Machine Intelligence (TPAMI)*, 31(10):1831–1846. 26

[Ess et al., 2007] Ess, A., Leibe, B., and Van Gool, L. (2007). Depth and appearance for mobile scene analysis. In *International Conference on Computer Vision (ICCV)*, pages 1–8. 26

[Evgeniou and Pontil, 2004] Evgeniou, T. and Pontil, M. (2004). Regularized multi–task learning. In *Proceedings of the tenth ACM SIGKDD international conference on Knowledge discovery and data mining*. 44, 65

[Fei-Fei and Perona, 2005] Fei-Fei, L. and Perona, P. (2005). A bayesian hierarchical model for learning natural scene categories. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, volume 2, pages 524–531. 94

[Felzenszwalb et al., 2010] 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 (TPAMI)*, 32(9):1627–1645. 3, 10, 20, 63, 98, 99, 100

[Fleuret et al., 2008] Fleuret, F., Berclaz, J., Lengagne, R., and Fua, P. (2008). Multicamera people tracking with a probabilistic occupancy map. *IEEE Transactions on Pattern Analysis and Machine Intelligence (TPAMI)*, 30(2):267–282. 27

[Gammeter et al., 2008] Gammeter, S., Ess, A., Jäggli, T., Schindler, K., Leibe, B., and Van Gool, L. (2008). Articulated multi-body tracking under egomotion. In *European Conference on Computer Vision (ECCV)*, pages 816–830. 6

[Gavrila and Munder, 2007] Gavrila, D. M. and Munder, S. (2007). Multi-cue pedestrian detection and tracking from a moving vehicle. *International Journal of Computer Vision (IJCV)*, 73(1):41–59. 29

[Giebel et al., 2004] Giebel, J., Gavrila, D. M., and Schnörr, C. (2004). A bayesian framework for multi-cue 3d object tracking. In *European Conference on Computer Vision (ECCV)*, pages 241–252. 26, 28

[Han et al., 2007] Han, B., Joo, S.-W., and Davis, L. S. (2007). Probabilistic fusion tracking using mixture kernel-based bayesian filtering. In *International Conference on Computer Vision (ICCV)*, pages 1–8. 18

[Helbing and Molnar, 1995] Helbing, D. and Molnar, P. (1995). Social force model for pedestrian dynamics. *Phys. Rev. E*, 51(5):4282–4286. 31

[Henriques et al., 2011] Henriques, J., Caseiro, R., and Batista, J. (2011). Globally optimal solution to multi-object tracking with merged measurements. In *International Conference on Computer Vision (ICCV)*. 22, 26, 27

[Hess and Fern, 2009] Hess, R. and Fern, A. (2009). Discriminatively trained particle filters for complex multi-object tracking. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 240–247. 18

[Hofmann, 1999] Hofmann, T. (1999). Probabilistic latent semantic indexing. In *Proceedings of the 22nd annual international ACM SIGIR conference on Research and development in information retrieval*, pages 50–57. ACM. 93

[Hu et al., 2008] Hu, M., Ali, S., and Shah, M. (2008). Detecting global motion patterns in complex videos. In *IEEE International Conference on Pattern Recognition*, pages 1–5. 31

[Hu et al., 2012] Hu, W., Li, X., Luo, W., Zhang, X., Maybank, S., and Zhang, Z. (2012). Single and multiple object tracking using log-euclidean riemannian subspace and block-division appearance model. *IEEE Transactions on Pattern Analysis and Machine Intelligence (TPAMI)*. 18, 20, 22, 26, 27, 35

[Huang et al., 2008] Huang, C., Wu, B., and Nevatia, R. (2008). Robust object tracking by hierarchical association of detection responses. In *European Conference on Computer Vision (ECCV)*, pages 788–801. 16, 18

[Izadinia et al., 2012a] Izadinia, H., Saleemi, I., Li, W., and Shah, M. (2012a). (mp)2t: Multiple people multiple parts tracker. In *European Conference on Computer Vision (ECCV)*. 22

[Izadinia et al., 2012b] Izadinia, H., Saleemi, I., Li, W., and Shah, M. (2012b). Mp2t: Multiple people multiple parts tracker. In *European Conference on Computer Vision (ECCV)*, pages 100–114. Springer. 25, 26, 29, 35, 100, 114, 115, 116

[Ji and Ye, 2009] Ji, S. and Ye, J. (2009). An accelerated gradient method for trace norm minimization. In *Proceedings of the 26th Annual International Conference on Machine Learning (ICML)*. ACM. 45

[Jiang et al., 2007] Jiang, H., Fels, S., and Little, J. J. (2007). A linear programming approach for multiple object tracking. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 1–8. 18

[Jin and Mokhtarian, 2007] Jin, Y. and Mokhtarian, F. (2007). Variational particle filter for multi-object tracking. In *International Conference on Computer Vision (ICCV)*, pages 1–8. 18

[Kalal et al., 2012] Kalal, Z., Mikolajczyk, K., and Matas, J. (2012). Tracking-learning-detection. *IEEE Transactions on Pattern Analysis and Machine Intelligence (TPAMI)*, 34(7):1409–1422. 57, 74, 76, 79, 80

[KC and De Vleeschouwer, 2013] KC, A. K. and De Vleeschouwer, C. (2013). Discriminative label propagation for multi-object tracking with sporadic appearance features. In *International Conference on Computer Vision (ICCV)*, pages 2000–2007. 34

[Keni and Rainer, 2008] Keni, B. and Rainer, S. (2008). Evaluating multiple object tracking performance: the clear mot metrics. *EURASIP Journal on Image and Video Processing*. 37, 81

[Khan et al., 2004] Khan, Z., Balch, T., and Dellaert, F. (2004). An mcmc-based particle filter for tracking multiple interacting targets. In *European Conference on Computer Vision (ECCV)*, pages 279–290. 18

[Khan et al., 2005] Khan, Z., Balch, T., and Dellaert, F. (2005). Mcmc-based particle filtering for tracking a variable number of interacting targets. *IEEE Transactions on Pattern Analysis and Machine Intelligence (TPAMI)*, 27(11):1805–1819. 18

[Khan et al., 2006] Khan, Z., Balch, T., and Dellaert, F. (2006). Mcmc data association and sparse factorization updating for real time multitarget tracking with merged and multiple

measurements. *IEEE Transactions on Pattern Analysis and Machine Intelligence (TPAMI)*, 28(12):1960–1972. 18

[Kim and Xing, 2010] Kim, S. and Xing, E. P. (2010). Tree-guided group lasso for multi-task regression with structured sparsity. In *Proceedings of the 27th International Conference on Machine Learning (ICML)*, pages 543–550. 45

[Kratz and Nishino, 2010] Kratz, L. and Nishino, K. (2010). Tracking with local spatio-temporal motion patterns in extremely crowded scenes. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. 17, 21, 22, 27, 33

[Kratz and Nishino, 2012] Kratz, L. and Nishino, K. (2012). Tracking pedestrians using local spatio-temporal motion patterns in extremely crowded scenes. *IEEE Transactions on Pattern Analysis and Machine Intelligence (TPAMI)*, 34(5):987–1002. 33

[Kuo et al., 2010] Kuo, C., Huang, C., and Nevatia, R. (2010). Multi-target tracking by on-line learned discriminative appearance models. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. 22, 26, 28, 30, 98

[Kuo and Nevatia, 2011] Kuo, C.-H. and Nevatia, R. (2011). How does person identity recognition help multi-person tracking? In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 1217–1224. 19, 30

[Lazebnik et al., 2006] Lazebnik, S., Schmid, C., and Ponce, J. (2006). Beyond bags of features: Spatial pyramid matching for recognizing natural scene categories. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 2169–2178. 27

[Leal-Taixé et al., 2014] Leal-Taixé, L., Fenzi, M., Kuznetsova, A., Rosenhahn, B., and Savarese, S. (2014). Learning an image-based motion context for multiple people tracking. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. 89, 108, 110, 111, 113

[Leibe et al., 2008] Leibe, B., Schindler, K., Cornelis, N., and Van Gool, L. (2008). Coupled object detection and tracking from static cameras and moving vehicles. *PAMI*, 30(10):1683–1698. 65

[Li et al., 2009] Li, Y., Huang, C., and Nevatia, R. (2009). Learning to associate: Hybrid-boosted multi-target tracker for crowded scene. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. 28, 37, 57, 81

[Liu et al., 2009] Liu, J., Ji, S., and Ye, J. (2009). Multi-task feature learning via efficient l 2, 1-norm minimization. In *Proceedings of the Twenty-Fifth Conference on Uncertainty in Artificial Intelligence (UAI)*. AUAI Press. 45

[Liu et al., 2005] Liu, X., Tu, P. H., Rittscher, J., Perera, A., and Krahnstoever, N. (2005). Detecting and counting people in surveillance applications. In *IEEE Conference on Advanced Video and Signal Based Surveillance (AVSS)*, pages 306–311. IEEE. 6

[Liu et al., 2012] Liu, Y., Li, H., and Chen, Y. Q. (2012). Automatic tracking of a large number of moving targets in 3d. In *European Conference on Computer Vision (ECCV)*, pages 730–742. 17, 18, 28

[Lowe, 1999] Lowe, D. G. (1999). Object recognition from local scale-invariant features. In *International Conference on Computer Vision (ICCV)*, volume 2, pages 1150–1157. 6

[Lowe, 2004] Lowe, D. G. (2004). Distinctive image features from scale-invariant keypoints. *International Journal of Computer Vision (IJCV)*, 60(2):91–110. 25, 27

[Luo and Kim, 2013] Luo, W. and Kim, T.-K. (2013). Generic object crowd tracking by multi-task learning. In *British Machine Vision Conference (BMVC)*. xxiii, 11, 12, 24, 63, 65, 74, 76, 80, 81, 90, 106, 107, 108

[Luo et al., 2014a] Luo, W., Kim, T.-K., Stenger, B., Zhao, X., and Cipolla, R. (2014a). Bi-label propagation for generic multiple object tracking. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. xxiii, 12, 24, 89, 90, 106, 107, 108, 110

[Luo et al., 2015] Luo, W., Stenger, B., Zhao, X., and Kim, T.-K. (2015). Automatic topic discovery for multi-object tracking. In *Proc. of the Association for the Advancement of Artificial Intelligence (AAAI)*. 12

[Luo et al., 2014b] Luo, W., Zhao, X., and Kim, T.-K. (2014b). Multiple object tracking: A review. *arXiv:1409.7618*. 23

[Luo et al., 2013] Luo, Y., Tao, D., Geng, B., Xu, C., and Maybank, S. (2013). Manifold regularized multi-task learning for semi-supervised multi-label image classification. *IEEE Transactions on Image Processing (TIP)*. 51

[Mikolajczyk et al., 2006] Mikolajczyk, K., Leibe, B., and Schiele, B. (2006). Multiple object class detection with a generative model. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. 43

[Milan et al., 2014] Milan, A., Roth, S., and Schindler, K. (2014). Continuous energy minimization for multitarget tracking. *IEEE Transactions on Pattern Analysis and Machine Intelligence (TPAMI)*, 36(1):58–72. 29

[Milan et al., 2013a] Milan, A., Schindler, K., and Roth, S. (2013a). Detection- and trajectory-level exclusion in multiple object tracking. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 3682–3689. 34

[Milan et al., 2013b] Milan, A., Schindler, K., and Roth, S. (2013b). Detection-and trajectory-level exclusion in multiple object tracking. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. 23, 89, 90, 108, 110, 111, 113

[Mitzel et al., 2010] Mitzel, D., Horbert, E., Ess, A., and Leibe, B. (2010). Multi-person tracking with sparse detection and continuous segmentation. In *European Conference on Computer Vision (ECCV)*, pages 397–410. 25, 26, 28, 36

[Mitzel and Leibe, 2011] Mitzel, D. and Leibe, B. (2011). Real-time multi-person tracking with detector assisted structure propagation. In *International Conference on Computer Vision Workshops (ICCV Workshops)*, pages 974–981. 17, 25

[Moeslund et al., 2006] Moeslund, T. B., Hilton, A., and Krüger, V. (2006). A survey of advances in vision-based human motion capture and analysis. *Computer Vision and Image Understanding*, 104(2):90–126. 6

[Nesterov, 2007] Nesterov, Y. (2007). Gradient methods for minimizing composite objective function. core discussion papers 2007076, université catholique de louvain. *Center for Operations Research and Econometrics (CORE)*. 52

[Nillius et al., 2006] Nillius, P., Sullivan, J., and Carlsson, S. (2006). Multi-target tracking-linking identities using bayesian network inference. In *Proc. IEEE Int. Conf. Comput. Vis. Pattern Recognit.*, pages 2187–2194. 30

[Obozinski et al., 2010] Obozinski, G., Taskar, B., and Jordan, M. I. (2010). Joint covariate selection and joint subspace selection for multiple classification problems. *Statistics and Computing*, 20(2):231–252. 45

[Okuma et al., 2004] Okuma, K., Taleghani, A., de Freitas, N., Little, J., and Lowe, D. (2004). A boosted particle filter: Multitarget detection and tracking. In *European Conference on Computer Vision (ECCV)*, pages 28–39. xxii, 25, 56, 59, 74, 81

[Pellegrini et al., 2009] Pellegrini, S., Ess, A., Schindler, K., and Van Gool, L. (2009). You'll never walk alone: Modeling social behavior for multi-target tracking. In *International Conference on Computer Vision (ICCV)*, pages 261–268. IEEE. 22, 32, 89, 108, 110, 111, 112, 113

[Perera et al., 2006] Perera, A. A., Srinivas, C., Hoogs, A., Brooksby, G., and Hu, W. (2006). Multi-object tracking through simultaneous long occlusions and split-merge conditions. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 666–673. 18

[Pirsiavash et al., 2011] Pirsiavash, H., Ramanan, D., and Fowlkes, C. C. (2011). Globally-optimal greedy algorithms for tracking a variable number of objects. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 1201–1208. 18, 89, 108, 110

[Porikli et al., 2006] Porikli, F., Tuzel, O., and Meer, P. (2006). Covariance tracking using model update based on lie algebra. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 728–735. 26

[Qin and Shelton, 2012] Qin, Z. and Shelton, C. (2012). Improving multi-target tracking via social grouping. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. 18, 22, 23, 30, 32

[Reid, 1979] Reid, D. B. (1979). An algorithm for tracking multiple targets. *IEEE Trans. Autom. Control*, 24(6):843–854. 17

[Reilly et al., 2010] Reilly, V., Idrees, H., and Shah, M. (2010). Detection and tracking of large number of targets in wide area surveillance. In *European Conference on Computer Vision (ECCV)*. 8, 18, 73

[Rodriguez et al., 2009] Rodriguez, M., Ali, S., and Kanade, T. (2009). Tracking in unstructured crowded scenes. In *International Conference on Computer Vision (ICCV)*, pages 1389–1396. 26, 33

[Rodriguez et al., 2011] Rodriguez, M., Sivic, J., Laptev, I., and Audibert, J.-Y. (2011). Data-driven crowd analysis in videos. In *International Conference on Computer Vision (ICCV)*, pages 1235–1242. 17, 26, 33

[Ryoo and Aggarwal, 2008] Ryoo, M. S. and Aggarwal, J. K. (2008). Observe-and-explain: A new approach for multiple hypotheses tracking of humans and objects. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 1–8. 36

[Scovanner and Tappen, 2009] Scovanner, P. and Tappen, M. F. (2009). Learning pedestrian dynamics from the real world. In *International Conference on Computer Vision (ICCV)*, pages 381–388. 32

[Shafique et al., 2008] Shafique, K., Lee, M. W., and Haering, N. (2008). A rank constrained continuous formulation of multi-frame multi-target tracking problem. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 1–8. 18, 29

[Shi and Tomasi, 1994] Shi, J. and Tomasi, C. (1994). Good features to track. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 593–600. 25

[Shu et al., 2012] Shu, G., Dehghan, A., Oreifej, O., Hand, E., and Shah, M. (2012). Part-based multiple-person tracking with partial occlusion handling. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. 18, 22, 23, 35, 90, 100, 114, 115, 116

[Song et al., 2010] Song, B., Jeng, T., Staudt, E., and Roy-Chowdhury, A. (2010). A stochastic graph evolution framework for robust multi-target tracking. In *European Conference on Computer Vision (ECCV)*. 19, 22, 23, 25, 28

[Sugimura et al., 2009] Sugimura, D., Kitani, K., Okabe, T., Sato, Y., and Sugimoto, A. (2009). Using individuality to track individuals: Clustering individual trajectories in crowds using local appearance and frequency trait. In *International Conference on Computer Vision (ICCV)*. 21, 22, 25, 27

[Sun et al., 2006] Sun, Z., Bebis, G., and Miller, R. (2006). On-road vehicle detection: A review. *IEEE Transactions on Pattern Analysis and Machine Intelligence (TPAMI)*, 28(5):694–711. 20

[Tang et al., 2013] Tang, S., Andriluka, M., Milan, A., Schindler, K., Roth, S., and Schiele, B. (2013). Learning people detectors for tracking in crowded scenes. In *International Conference on Computer Vision (ICCV)*, pages 1049–1056. 36

[Tang et al., 2014] Tang, S., Andriluka, M., and Schiele, B. (2014). Detection and tracking of occluded people. *International Journal of Computer Vision (IJCV)*, 110(1):58–69. 36

[Teh, 2010] Teh, Y. W. (2010). Dirichlet process. In *Encyclopedia of machine learning*, pages 280–287. Springer. 94

[Teh et al., 2006] Teh, Y. W., Jordan, M. I., Beal, M. J., and Blei, D. M. (2006). Hierarchical Dirichlet processes. *Journal of the American Statistical Association*, 101(476):1566–1581. 94

[Topkaya et al., 2013] Topkaya, I. S., Erdogan, H., and Porikli, F. (2013). Detecting and tracking unknown number of objects with dirichlet process mixture models and markov random fields. In *Advances in Visual Computing*, pages 178–188. 94

[Torralba et al., 2007] Torralba, A., Murphy, K., and Freeman, W. (2007). Sharing visual features for multiclass and multiview object detection. *IEEE Transactions on Pattern Analysis and Machine Intelligence (TPAMI)*, 29(5):854–869. 43, 46

[Tuzel et al., 2006] Tuzel, O., Porikli, F., and Meer, P. (2006). Region covariance: A fast descriptor for detection and classification. In *European Conference on Computer Vision (ECCV)*, pages 589–600. 26

[Verbeek and Triggs, 2007] Verbeek, J. and Triggs, B. (2007). Region classification with markov field aspect models. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 1–8. IEEE. 94

[Viola and Jones, 2004] Viola, P. and Jones, M. (2004). Robust real-time face detection. *International Journal of Computer Vision (IJCV)*, 57(2):137–154. 46

[Wang et al., 2009a] Wang, C., Blei, D., and Li, F.-F. (2009a). Simultaneous image classification and annotation. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 1903–1910. 94

[Wang and Grimson, 2008] Wang, X. and Grimson, E. (2008). Spatial latent dirichlet allocation. In *Advances in Neural Information Processing Systems (NIPS)*, pages 1577–1584. 94

[Wang et al., 2009b] Wang, X., Han, T., and Yan, S. (2009b). An hog-lbp human detector with partial occlusion handling. In *International Conference on Computer Vision (ICCV)*. 55

[Wang et al., 2011] Wang, X., Ma, K. T., Ng, G.-W., and Grimson, W. E. L. (2011). Trajectory analysis and semantic region modeling using nonparametric hierarchical bayesian models. *International Journal of Computer Vision (IJCV)*, 95(3):287–312. 94

[Wolf et al., 1989] Wolf, J. K., Viterbi, A. M., and Dixon, G. S. (1989). Finding the best set of k paths through a trellis with application to multitarget tracking. *IEEE Trans. Aerosp. Electron. Syst.*, 25(2):287–296. 18

[Wu and Nevatia, 2007a] Wu, B. and Nevatia, R. (2007a). Cluster boosted tree classifier for multi-view, multi-pose object detection. In *International Conference on Computer Vision (ICCV)*. 43

[Wu and Nevatia, 2007b] Wu, B. and Nevatia, R. (2007b). Detection and tracking of multiple, partially occluded humans by bayesian combination of edgelet based part detectors. *International Journal of Computer Vision (IJCV)*, 75(2):247–266. 18

[Wu et al., 2012] Wu, Z., Thangali, A., Sclaroff, S., and Betke, M. (2012). Coupling detection and data association for multiple object tracking. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. 18, 23, 27, 65

[Xing et al., 2009] Xing, J., Ai, H., and Lao, S. (2009). Multi-object tracking through occlusions by local tracklets filtering and global tracklets association with detection responses. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 1200–1207. 18, 24, 30, 89, 108

[Yamaguchi et al., 2011] Yamaguchi, K., Berg, A., Ortiz, L., and Berg, T. (2011). Who are you with and where are you going? In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. 22, 25, 27, 32

[Yang et al., 2011] Yang, B., Huang, C., and Nevatia, R. (2011). Learning affinities and dependencies for multi-target tracking using a crf model. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. 22, 30

[Yang and Nevatia, 2012a] Yang, B. and Nevatia, R. (2012a). Multi-target tracking by online learning of non-linear motion patterns and robust appearance models. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. xv, 22, 23, 30, 31

[Yang and Nevatia, 2012b] Yang, B. and Nevatia, R. (2012b). An online learned crf model for multi-target tracking. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. 22, 23, 30

[Yang and Nevatia, 2012c] Yang, B. and Nevatia, R. (2012c). Online learned discriminative part-based appearance models for multi-human tracking. In *European Conference on Computer Vision (ECCV)*, pages 484–498. xxi, 19, 21, 28, 35

[Yang et al., 2005] Yang, C., Duraiswami, R., and Davis, L. (2005). Fast multiple object tracking via a hierarchical particle filter. In *International Conference on Computer Vision (ICCV)*, pages 212–219. 18

[Yang et al., 2009] Yang, M., Lv, F., Xu, W., and Gong, Y. (2009). Detection driven adaptive multi-cue integration for multiple human tracking. In *International Conference on Computer Vision (ICCV)*. 17, 18, 21, 22, 25, 27

[Yang et al., 2007] Yang, M., Yu, T., and Wu, Y. (2007). Game-theoretic multiple target tracking. In *International Conference on Computer Vision (ICCV)*, pages 1–8. 20

[Yang et al., 2013] Yang, Y., Shu, G., and Shah, M. (2013). Semi-supervised learning of feature hierarchies for object detection in a video. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. 65

[Yilmaz et al., 2006] Yilmaz, A., Javed, O., and Shah, M. (2006). Object tracking: A survey. *ACM Computer Survey*, 38(4):13. 19

[Yilmaz et al., 2004] Yilmaz, A., Li, X., and Shah, M. (2004). Contour-based object tracking with occlusion handling in video acquired using mobile cameras. *IEEE Transactions on Pattern Analysis and Machine Intelligence (TPAMI)*, 26(11):1531–1536. 6

[Yu et al., 2007] Yu, Q., Medioni, G., and Cohen, I. (2007). Multiple target tracking using spatio-temporal markov chain monte carlo data association. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 1–8. 29

[Yu et al., 2008] Yu, T., Wu, Y., Krahnstoever, N. O., and Tu, P. H. (2008). Distributed data association and filtering for multiple target tracking. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 1–8. 26

[Zhan et al., 2008] Zhan, B., Monekosso, D. N., Remagnino, P., Velastin, S. A., and Xu, L.-Q. (2008). Crowd analysis: a survey. *Machine Vision Application*, 19(5):345–357. 33

[Zhang et al., 2008] Zhang, L., Li, Y., and Nevatia, R. (2008). Global data association for multi-object tracking using network flows. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 1–8. 18, 23, 36, 89, 108, 110, 111, 113

[Zhang and van der Maaten, 2013a] Zhang, L. and van der Maaten, L. (2013a). Preserving structure in model-free tracking. *IEEE Transactions on Pattern Analysis and Machine Intelligence (TPAMI)*. 20

[Zhang and van der Maaten, 2013b] Zhang, L. and van der Maaten, L. (2013b). Structure preserving object tracking. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. 20, 65, 74, 81

[Zhang et al., 2007] Zhang, L., Wu, B., and Nevatia, R. (2007). Detection and tracking of multiple humans with extensive pose articulation. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 1–8. 6

[Zhang et al., 2012] Zhang, T., Ghanem, B., Liu, S., and Ahuja, N. (2012). Robust visual tracking via multi-task sparse learning. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. 46, 53

[Zhang et al., 2014] Zhang, X., Li, C., Hu, W., Tong, X., Maybank, S., and Zhang, Y. (2014). Human pose estimation and tracking via parsing a tree structure based human model. *IEEE Transactions on Systems, Man, and Cybernetics: System*, 44(5):580–592. 6

[Zhao et al., 2012] Zhao, X., Gong, D., and Medioni, G. (2012). Tracking using motion patterns for very crowded scenes. In *European Conference on Computer Vision (ECCV)*, pages 315–328. Springer. 25, 38, 57, 107

[Zhou et al., 2011a] Zhou, B., Wang, X., and Tang, X. (2011a). Random field topic model for semantic region analysis in crowded scenes from tracklets. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 3441–3448. 6

[Zhou et al., 2011b] Zhou, J., Chen, J., and Ye, J. (2011b). Clustered multi-task learning via alternating structure optimization. *Advances in Neural Information Processing Systems (NIPS)*. 45, 65, 69, 70, 121, 122, 123

[Zhou et al., 2011c] Zhou, J., Chen, J., and Ye, J. (2011c). *MALSAR: Multi-tAsk Learning via StructurAl Regularization*. Arizona State University. 54

[Zhu et al., 2005] Zhu, X., Ghahramani, Z., and Lafferty, J. (2005). Time-sensitive dirichlet process mixture models. Technical report, DTIC Document. 103