Discovering Human Interactions in Videos with Limited Data Labeling

by

Mehran Khodabandeh
B.Sc. Sharif University of Technology, 2008

Thesis Submitted in Partial Fulfillment of the Requirements for the Degree of

Master of Science

in the
School of Computing Science
Faculty of Applied Sciences

© Mehran Khodabandeh 2015
SIMON FRASER UNIVERSITY
Spring 2015

All rights reserved. However, in accordance with the Copyright Act of Canada, this work may be reproduced without authorization under the conditions for “Fair Dealing”. Therefore, limited reproduction of this work for the purposes of private study, research, criticism, review and news reporting is likely to be in accordance with the law, particularly if cited appropriately.
APPROVAL

Name: Mehran Khodabandeh
Degree: Master of Science
Title: Discovering Human Interactions in Videos with Limited Data Labeling

Examining Committee: Chair: Dr. Ghassan Hamarneh
Simon Fraser University
Professor

Dr. Greg Mori
Senior Supervisor
Simon Fraser University
Associate Professor

Dr. Ze-Nian Li
Co-Supervisor
Simon Fraser University
Professor

Dr. Ping Tan
Internal Examiner
Simon Fraser University
Assistant Professor

Date Approved: January 7, 2015
Partial Copyright Licence

The author, whose copyright is declared on the title page of this work, has granted to Simon Fraser University the non-exclusive, royalty-free right to include a digital copy of this thesis, project or extended essay[s] and associated supplemental files (“Work”) (title[s] below) in Summit, the Institutional Research Repository at SFU. SFU may also make copies of the Work for purposes of a scholarly or research nature; for users of the SFU Library; or in response to a request from another library, or educational institution, on SFU’s own behalf or for one of its users. Distribution may be in any form.

The author has further agreed that SFU may keep more than one copy of the Work for purposes of back-up and security; and that SFU may, without changing the content, translate, if technically possible, the Work to any medium or format for the purpose of preserving the Work and facilitating the exercise of SFU’s rights under this licence.

It is understood that copying, publication, or public performance of the Work for commercial purposes shall not be allowed without the author's written permission.

While granting the above uses to SFU, the author retains copyright ownership and moral rights in the Work, and may deal with the copyright in the Work in any way consistent with the terms of this licence, including the right to change the Work for subsequent purposes, including editing and publishing the Work in whole or in part, and licensing the content to other parties as the author may desire.

The author represents and warrants that he/she has the right to grant the rights contained in this licence and that the Work does not, to the best of the author’s knowledge, infringe upon anyone’s copyright. The author has obtained written copyright permission, where required, for the use of any third-party copyrighted material contained in the Work. The author represents and warrants that the Work is his/her own original work and that he/she has not previously assigned or relinquished the rights conferred in this licence.

Simon Fraser University Library
Burnaby, British Columbia, Canada

revised Fall 2013
Abstract

We present a novel approach for discovering human interactions in videos. Activity understanding techniques usually require a large number of labeled examples, which are not available in many practical cases. Here, we focus on recovering semantically meaningful clusters of human-human and human-object interaction in an unsupervised fashion. A new iterative solution is introduced based on Maximum Margin Clustering (MMC), which also accepts user feedback to refine clusters. This is achieved by formulating the whole process as a unified constrained latent max-margin clustering problem. Extensive experiments have been carried out over three challenging datasets, Collective Activity, VIRAT, and UT-interaction. Empirical results demonstrate that the proposed algorithm can efficiently discover perfect semantic clusters of human interactions with only a small amount of labeling effort.

Keywords: Clustering; Human interactions; Video; Maximum Margin; User feedback
I would like to take this opportunity to first and foremost thank God for being my strength and guide in completing my Masters degree. Without Him, I would not have had the wisdom or the physical ability to do so.

I would like to thank all the people who contributed in some way to the work described in this thesis. I have to express my gratitude towards my supervisor, Dr. Greg Mori. His leadership, support, attention to details, and hard work have set an example I hope to match some day. I attribute the level of my Masters degree to his encouragement and effort and without him this thesis, too, would not have been completed or written. One simply could not wish for a better or friendlier supervisor. I also would like to thank my second supervisor, Professor Ze-Nian Li, a talented teacher and passionate adviser. I thank him for appreciating my research strengths and patiently encouraging me to improve in my weaker areas. I would like to thank the members of my examining committee, Dr. Ghassan Hamarneh and Dr. Ping Tan, not only for their time and extreme patience, but for their intellectual contributions to my development.

Every result described in this thesis was accomplished with the help and support of fellow lab-mates and collaborators. Arash Vahdat, Hossein Hajimirsadeghi, Mehrsan Javan, Guang-Tong Zhu, Stephen Se, and I worked together on several different phases of this project, and without their efforts my job would have undoubtedly been more difficult. I greatly benefited from Hossein's keen insight and invaluable advices. Out of all people in the Vision and Media Lab, I will miss hanging out with Ali Madooei the most. I can only hope that my future lab-mates will be as kind and hilarious as him. I would also like to thank my dear friend, Mohammad Asi, for always encouraging me in all of my pursuits and inspiring me to follow my dreams.

I thank my parents for always being supportive of my education. Although they are so far away there has never been a time when I have felt alone, whether in the writing of this
thesis or in the general attainment of my education. I would also like to thank my younger brother, Mehrdad, who is not only a kind and passionate brother but also my best friend in my life.
Contents

Approval ii
Partial Copyright License iii
Abstract iv
Acknowledgements v
Contents vii
List of Tables ix
List of Figures x

1 Introduction 1

2 Previous Work 4
  2.1 Supervised Activity Recognition 4
  2.2 Unsupervised Activity Recognition 7
  2.3 Clustering Methods 9

3 Clustering Human Interactions 10
  3.1 Maximum Margin Clustering 11
  3.2 Max-Margin Clustering with User Feedback 11
    3.2.1 Formulation 12
    3.2.2 Optimization 15
  3.3 Relaxed Latent Maximum Margin Clustering 15
<table>
<thead>
<tr>
<th>Section</th>
<th>Title</th>
<th>page</th>
</tr>
</thead>
<tbody>
<tr>
<td>4</td>
<td>Features and Implementation Details</td>
<td>18</td>
</tr>
<tr>
<td>4.1</td>
<td>Proximity Features</td>
<td>19</td>
</tr>
<tr>
<td>4.2</td>
<td>Appearance Features</td>
<td>20</td>
</tr>
<tr>
<td>4.3</td>
<td>Latent Variables for Temporal Alignment</td>
<td>20</td>
</tr>
<tr>
<td>5</td>
<td>Experiments</td>
<td>23</td>
</tr>
<tr>
<td>5.1</td>
<td>Datasets</td>
<td>23</td>
</tr>
<tr>
<td>5.1.1</td>
<td>UT-Interaction Dataset:</td>
<td>23</td>
</tr>
<tr>
<td>5.1.2</td>
<td>Collective Activity:</td>
<td>25</td>
</tr>
<tr>
<td>5.1.3</td>
<td>VIRAT Dataset:</td>
<td>25</td>
</tr>
<tr>
<td>5.2</td>
<td>Results</td>
<td>26</td>
</tr>
<tr>
<td>5.2.1</td>
<td>Fully unsupervised (iteration-0):</td>
<td>26</td>
</tr>
<tr>
<td>5.2.2</td>
<td>User feedback:</td>
<td>27</td>
</tr>
<tr>
<td>5.2.3</td>
<td>Relaxed Latent Max-Margin Clustering:</td>
<td>30</td>
</tr>
<tr>
<td>6</td>
<td>Conclusion and Future Work</td>
<td>34</td>
</tr>
<tr>
<td>6.1</td>
<td>Limitations</td>
<td>34</td>
</tr>
<tr>
<td>6.2</td>
<td>Future Directions</td>
<td>35</td>
</tr>
<tr>
<td>Bibliography</td>
<td></td>
<td>36</td>
</tr>
</tbody>
</table>
List of Tables

5.1 Summary of VIRAT clustering user study results. For each scene, the mean percentage of user responses choosing a video in the same cluster, deeming both to be similar, choosing the out of cluster video, or neither is reported. Standard deviations over the four users is reported. . . . . . . . . . . . . 30
List of Figures

1.1 Given a set of videos, we aim to extract clusters corresponding to different types of human interactions. The top shows a set of sequences of human-human interactions. In the first step, an unsupervised clustering approach is used to create an initial set of clusters. Next, user feedback is obtained iteratively to refine the clusters. ................................................................. 2

2.1 On the right side the graphical model is represented and on the left side an activity is illustrated in time and space [50] ©2013 IEEE. Reprinted, with permission, from Zhu et al., Context-Aware Modeling and Recognition of Activities in Video, Computer Vision and Pattern Recognition (CVPR), 2013. 5

2.2 The pipeline of Bandla and Grauman’s method [2] ©2013 IEEE. Reprinted, with permission, from Bandla and Grauman, Active Learning of an Action Detector from Untrimmed Videos, Computer Vision and Pattern Recognition (CVPR), 2013. ............................................................... 7

2.3 Overview of Neibels et al. [26] method, used with kind permission from Springer Science and Business Media. ................................................................. 8

3.1 Overview of our iterative algorithm. (a) Unsupervised step: Data are clustered into groups with large margins. Colored shape on the bottom left corner of each sample image indicates the ground truth label, which is unknown for the algorithm. (b) User feedback: The user specifies the must-link (green lines) and cannot-link (red lines) constraints. (c) Cluster refinement: Clusters consistent with must-link and cannot-link constraints are regenerated. ................................................................. 13
4.1 Latent variables for aligning tracks. Two example trajectories are shown in time-lapse form top and bottom. Each contains a person approaching a vehicle. However, there are differences in the behavior of each person over the course of the track, with substantial variation in the amount of time the person lingers in front of the vehicle, and what he does afterwards. Latent temporal regions (in green) can be chosen to align these trajectories and extract more similar feature representations.

5.1 Performance of our clustering method compared to other baseline methods on different datasets. For all datasets, our proposed method generates clusters of high purity without any supervision on activity labels.

5.2 Confusion matrices of clusters generated at iteration-0 for UT Interactions Set 1 (a), Set 2 (b), Collective Activity dataset (c), and VIRAT(d).

5.3 Average purity of the proposed clustering model for UT-Interaction Set 1 (a), Set 2 (b), Collective Activity (c), and VIRAT (d). Our method constructs 100% pure clusters after a few iterations of obtaining user-feedback. Our performance is significantly better than the manually labeled baseline that uses the feedback for correcting misclustered interactions that are generated in iteration-0 with zero supervision.

5.4 Examples of VIRAT clustering results. Each rectangle shows a set of three sequences from a single cluster.

5.5 Examples of VIRAT clustering results. Each rectangle shows a set of three sequences from a single cluster.
Chapter 1

Introduction

We are living in the age of information. Visual data is one of the fastest growing kind of information. Every minute or hour of videos are being captured from different sources such as surveillance, movies, hand-held cameras, sports, robots, and etc. A problem that has raised attention is discovering contents of these videos, which is still an open problem. Manual approaches for analyzing videos such as brute force is infeasible, since the immense data and videos are growing at a tremendous rate. This is where computer vision plays an effective role.

Automatic discovery of human activities and event detection in videos is a topic in computer vision which has been explored extensively [29]. It is a challenging problem with a variety of applications such as surveillance, robotics, entertainments, and etc. In addition, it is still immature in some specific sub-problems such as discovery of human interactions. The term “interaction” refers to any kind of interaction between humans, and humans and objects that are present in the scene, such as vehicles, rather than activities which are performed by a single subject.

Automated analysis of videos of human activity can take many forms – answering questions about the presence of specific types of activities through to the discovery of what has happened in a scene. In this paper we focus on the latter and present an algorithm to label human interactions in videos. The algorithm works in a clustering paradigm, starting with an unsupervised step that forms groups of similar human interactions. These clusters are refined based on user feedback, and the process is iterated, as shown in Fig. 1.1.

Different strategies can be followed in order to label how people are interacting in a set of input videos. Brute-force labeling approaches involving manual labour are costly,
Figure 1.1: Given a set of videos, we aim to extract clusters corresponding to different types of human interactions. The top shows a set of sequences of human-human interactions. In the first step, an unsupervised clustering approach is used to create an initial set of clusters. Next, user feedback is obtained iteratively to refine the clusters.

since input videos often cover a long period of time. Hence, a common approach is to use supervised learning and focus on detecting a set of pre-specified activities of interest (e.g. [22, 50, 35]). For instance, an algorithm can be pre-trained to detect instances of people getting into vehicles, and then find all instances of that specific event. To obtain high accuracy, those approaches often require lots of labeled training data, which is not easy to obtain in many cases.

Extensions based on active learning can be used to build up a collection of labeled data, while being efficient with human labeling effort (e.g. [44, 2, 38]). Impressive results have been obtained by these supervised methods, however, these remain limited to pre-specified categories of events.

On the other hand, unsupervised analysis techniques aim to obtain clusters of human activities or perform novelty/outlier detection to find rare events. This paradigm is attractive since it requires neither a priori specification of events nor human labeling effort. Effective methods in this vein have been developed previously (e.g. [11, 48, 39, 21, 23, 32, 24]). In general, those methods focus on either creating one (or a few) big clusters or a large number of clusters of common activities. In the former, those clusters do not necessarily
represent activities of the same labels and in the latter, there are many clusters that are representing the same type of activity. Our work follows in this line, but is focused on discovering and labeling common human interactions, utilizing a clustering approach to create meaningful activity/interaction groups and accepting user feedback to improve accuracy.

In this paper we propose a novel algorithm for discovering human interactions in video sequences. The algorithm performs iterated clustering and incorporation of user feedback. The contributions include a principled formulation of this process as a constrained latent max-margin clustering problem. We demonstrate that this algorithm can be very effective, obtaining state of the art clustering results from no labeled data, and obtaining perfect clustering after a small amount of user feedback.
Chapter 2

Previous Work

Human activity and interaction understanding is an active research area. Recent surveys such as Poppe [29] and Weinland et al. [41] provide an overview of the literature. We emphasize that the objective of this work is to describe human interactions rather than individual activities performed by a single subject.

2.1 Supervised Activity Recognition

There is an extensive literature on recognizing interactions or analyzing the behaviours of groups of people. Much of this work involves supervised learning, either in the form of specific classes of interactions to detect or templates/rules for detecting interactions of interest. The early works in this vein includes Medioni et al. [22], who analyzed vehicle trajectories, for instance detecting vehicles approaching or avoiding road checkpoints. Their method has two steps. In the first step, the motion of camera in 3D world is estimated and also the moving objects in the scene are detected and tracked. This process could be very noisy. Then in the second step, the behavior of moving objects obtained by the first step are recognized. In this step, contextual information on the trajectories which was provided by the user is used. Intille and Bobick [13] developed probabilistic graphical models for interpreting football plays based on player trajectories. In this work, a system is designed for recognizing collaborative activity using a multiperson action description. A modular probabilistic network is built on a collection of visual features. In addition, this work demonstrates that collaborative actions can be recognized using a large number of binary and simple comparisons.
Ryoo and Aggarwal [35] use stochastic grammars to incorporate sub-events and the actions of individuals into larger events. They use Context Free Grammar to represent the group activities in temporal, spatial, and logical order. On the other hand, they recognize actions of individuals in the scene. Then they use Markov Chain Monte Carlo to model the probability distribution of the group activity. Next, they perform a stochastic search to find individuals with maximum posterior probability in the group activity representation. Zhu et al. [50] develop a method for detecting specified human-vehicle interactions based on spatio-temporal contextual models. Assuming the idea that related activities are dependent in time and space, they build a hierarchical model on motion and contextual features to recognize the activities of interest. They perform motion segmentation and then use a max-margin framework to learn the spatio-temporal structure of detected action segments (Fig. 2.1). The mentioned structure consists of context patterns, motion, and duration of activities. Amer et al. [1] model activities at varying levels of detail, formulating AND-OR graph representations that permit efficient inference. They address the problem of localizing human activities whether performed by individuals or a group of people. The group activities are modeled using an AND-OR graph, where bottom layers relate to primitive actions or objects and higher levels model the activities. Next, a cost-sensitive inference algorithm is proposed on top of an explore-exploit method which uses the hierarchical structure of the graph. In addition, they released a new dataset for activity recognition in a
courtyard of UCLA campus. Choi and Savarese [5] develop a unified framework for track-
ing and inferring the actions/activities of a group of people. They model the activities in a
hierarchical way and then perform a bottom-up inference. The bottom layer of their model
represents atomic activities. Their proposed tracking algorithm works here and the results
are used in this layer. In the upper layer interactions are inferred and the top layer predicts
the collective activity. A max-margin framework is used to learn the model weights. Khamis
et al. [16] include temporal analysis of the actions of individuals and develop efficient in-
ference techniques for analyzing collective activities. To do so, they first train a linear SVM
to recognize actions of individuals. The classifier is trained on Action Context descriptors.
Then they leverage the idea that individual actions are in harmony with activity of the group
in a scene. Therefore, they use the SVM scores to obtain the action likelihood. In addition,
they maintain the identity of individuals across the frames by training an identity association
model on appearance and action cues. The main contribution is to formulate this task
in a unified model. Lan et al. [20] model interactions between individuals in a scene and
their relations to an over-arching scene-level activity label. Patron-Perez et al. [28] detect
human-human interactions in television shows using a structural SVM approach. Our work
builds on these methods for analyzing interactions, but aims for unsupervised learning or
discovery of interactions rather than the supervised approach common to these methods.

Active learning: In this approach, human labeling is integrated with a learning algo-
rithm, typically presenting the most uncertain or most helpful unlabeled data to a user to
acquire additional labels. This type of learning has been mainly deployed in the object/ac-
on the information measured on untrimmed videos. Then they ask a user to annotate
the videos with most valuable information according to the scores of the previous step.
Their method iteratively performs two major steps. In the first step, a Hough-based action
detector is used to detect the intervals in videos that actions of interest could occur with
high probability. Then they choose intervals that decrease the total uncertainty among all
videos. The next and final step is to request the user to annotate the candidate intervals.
The overall pipeline of this method is illustrated in Fig. 2.2. Kovashka et al [17] propose a
method that selects the most influential queries for the user based on an entropy reduction
criterion that they have defined. In their method, annotations include both object cate-
gories and object attributes. They model the relationship between object and attributes
or attributes and attributes using a discriminative latent model. Duan et al [9] develop an effective algorithm that localizes attributes in the images, which are both detectable and understandable by machine and human. Their method works iteratively involving a user who annotates candidates recommended by their algorithm as semantically meaningful attributes. In their proposed method, attributes are split into two categories, considered as most confusable. A latent conditional random field approach is used to do this task. Then the user is requested to annotate them.

Our approach shares similarities, though is focused on interaction discovery, within a clustering paradigm rather than supervised recognition approach.

2.2 Unsupervised Activity Recognition

A diverse set of unsupervised methods has been developed for activity analysis, ranging from pixel-level flow models to the clustering of person trajectories. In general, holistic scene models are deemed to have the advantage of being more robust compared to tracking-based approaches, because of the challenges in tracking individual people. But, they are typically limited in the level of semantic detail that can be modeled. Examples of work in this area include Zhong et al. [48], who performed novelty detection in a clustering framework based on long videos represented using spatio-temporal derivatives. Mehran
et al. [23] developed a social force model for interpreting the behaviour of crowds of people observed from a long distance.

Other related methods typically model small patches of scenes. Hospedales et al. [11] and Wang et al. [39] build novel topic models for the actions of people or vehicles in surveillance scenes. Kuettel et al. [18] model temporal evolution of discovered topics or activities, for instance discovering different phases of activity.

More closely related to our approach are those that try to form clusters of human activities using unlabeled data. Niebles et al. [26] use a topic model over a bag-of-words representation from local features around a person. They used probabilistic Latent Semantic Analysis (pLSA) and Latent Dirichlet Allocation (LDA) to model human action categories. By detecting spatio-temporal interest points they are able to localize different human actions simultaneously in a video sequence. The overview of their method is illustrated in Fig. 2.3. Wang et al. [40] cluster images using shape features to discover action classes. Calderara et al. [4] track individuals and reason about scenes to find anomalous trajectories. In this work we develop a clustering framework that examines trajectories in an unsupervised fashion, but reasons about interactions between these trajectories of individuals and objects.
2.3 Clustering Methods

Clustering is a widely-studied problem; many standard clustering methods have been created, such as k-means, spectral clustering [25], topic models [3], and a variety of mixture models. In addition to the aforementioned methods, maximum margin clustering (MMC) [42] emphasizes the separation between classes. MMC is an extension of max-margin supervised learning (i.e. SVMs). Given a set of observations, MMC performs clustering by finding the hyperplanes with maximum margin through the data. Experimental results have shown that this method often outperforms competing clustering methods.

Supervised large margin methods usually lead to convex optimization problems, while solving unsupervised versions require untangling a non-convex integer program. Therefore, recent research tackled the problem of reducing the computational complexity of MMC [47, 45]. Zhang et al. directly optimize the non-convex problem by changing the loss function to Laplacian loss, instead of optimizing the problem as a non-convex semidefinite program (SDP) [45]. Zhao et al. accelerated the convergence of MMC via a series of tighter relaxed MMC instances [47]. Another line of work is incorporating further information and constraints into MMC. Hu et al. [12] added slack variables for soft pairwise constraints. Zhou et al. developed a maximum margin framework that handles unobserved knowledge in data using latent variables [49]. We build on this line of work, developing a variant of this approach and a novel model for unsupervised discovery of human interactions.
Chapter 3

Clustering Human Interactions

We assume that an object detection and tracking algorithm exists and a set of trajectories are available\(^1\). Therefore, the goal is to cluster human trajectories based on their interactions with surrounding humans or vehicles. Each cluster should contain a semantically similar set of interactions. A common approach to this problem is to feed features extracted on each person to a standard clustering algorithm.

However, clustering interactions using a standard approach may not necessarily result in clusters of semantically similar interactions. Two key reasons are:

- **Feature representation**: The underlying features should represent the desired semantic similarity. Otherwise, grouping similar interactions in the space of low-level features cannot guarantee the formation of coherent high-level clusters.

- **Lack of supervision**: A purely unsupervised clustering algorithm is still prone to mistakes due to intra-class variation in high-level semantic classes.

The proposed algorithm can handle those issues effectively. An overview of the algorithm is presented in Fig. 3.1. We show that by injecting a small amount of user-provided feedback, errors in unsupervised learning can be corrected. Leveraging latent variable representations can address the feature representation issues. We formulate those ideas in a novel variant of max-margin clustering.

\(^1\)In chapter 5 we provide dataset-specific details on these algorithms.
3.1 Maximum Margin Clustering

MMC extends the principle of maximum margin in supervised learning (e.g. SVM) to unsupervised clustering, where the labels of data are unobserved. Given a set of examples $X = \{x_1, x_2, \ldots, x_N\}$, the goal of the algorithm is to find a set of binary labels $Y = \{y_{it}\} (i \in \{1, \ldots, N\}, t \in \{1, \ldots, K\})$. MMC groups the data into $K$ clusters in such a way that the margin between classes is maximal.

We can formulate it as the following optimization problem [49]:

\[
\min_{W, Y, \xi \geq 0} \frac{1}{2} \sum_{t=1}^{K} ||w_t||^2 + \frac{C}{K} \sum_{i=1}^{N} \sum_{r=1}^{K} \xi_{ir}
\]

s.t. \quad \forall i, r \quad \sum_{t=1}^{K} y_{it} w_t^\top \phi(x_i) - w_r^\top \phi(x_i) \geq 1 - y_{ir} - \xi_{ir},

\forall i, t \quad y_{it} \in \{0, 1\}, \quad \forall t \quad \sum_{t=1}^{K} y_{it} = 1

\forall t \quad L \leq \sum_{i=1}^{N} y_{it} \leq U

where $y_{it} = 1$ denotes that the example $x_i$ belongs to the cluster $t$, and is $y_{it} = 0$ otherwise. The feature vector for example $x_i$ is denoted as $\phi(x_i)$, i.e. the concatenation of the velocity, distance, and appearance features. In this formulation $\mathcal{W} = \{w_t\}_{t=1}^{K}$ contains the parameters of the model including bias term. And in order to have soft margin, we have the slack variables $\xi = \{\xi_{ir}\}, i \in \{1, \ldots, N\}, t \in \{1, \ldots, K\}$, and have constant $C$ to control effect of the slack variables.

Zhou et al. [49] extend this formulation to include latent variables which can modulate the feature representation for each data sample. We make use of this formulation, and further modify the so-called cluster balance constraints (Eq. 3.4) that prevent degenerate solutions.

3.2 Max-Margin Clustering with User Feedback

We propose a novel iterative clustering approach that improves the quality of clusters by iterations of obtaining user feedback on automatically-generated clusters. The basic idea is that a small amount of feedback in each iteration not only fixes mistakes in the clusters, but also can be generalized to other incorrectly clustered examples. This feedback will
reduce mistakes in clustering, cases where interactions are assigned to clusters whose dominant interaction type is semantically different (c.f. cluster purity measurements).

Assume that we have a set of clusters formed from a video dataset (Fig. 3.1(a)). A user can be asked to view the generated clusters and to mark a few examples, such as those corresponding to the dominant interaction in each cluster, or misplaced examples (Fig. 3.1(b)).

Some user-marked samples represent correctly clustered interactions that are semantically similar. Thus, in further clustering they must be grouped together. We represent these interactions in each cluster as must-link constraints.

Interactions that are in incorrect clusters can be moved by a user to their corresponding correct clusters. This implies that these samples and the ones in the must-link groups of the incorrect cluster should never be grouped together. This can be represented as cannot-link constraints formed between every pair of incorrectly clustered samples and samples in the must-link groups. Second, a must-link constraint should be formed with the samples in the correct group.

In summary, the user-provided feedback indicates a few samples that are correctly clustered and a few samples that should be moved to another cluster in order to improve the clustering quality. This feedback is collected iteratively and the clusters are re-generated (Fig. 3.1(c)), resulting in pure clusters after a few iterations.

### 3.2.1 Formulation

We modify the recently proposed latent max-margin clustering (MMC) [49] to formulate our clustering idea. The formulation of MMC is explained in Sec. 3.1. This formulation is extended to include latent variables which can modulate the feature representation for each data sample. In this case, the features for each example are altered by the notion of latent variables such that the separation between clusters is maximized. However, neither MMC nor latent MMC is capable of incorporating user feedback while discovering clusters of similar interactions. Here, we propose a novel extension of the latent max-margin clustering framework that is able to collect feedback from a user on a set of clusters in order to improve their quality iteratively.

The must-link and cannot-link constraints respectively indicate a set of points that must and must not be grouped together. The set of all must-link constraints is represented using
Figure 3.1: Overview of our iterative algorithm. (a) Unsupervised step: Data are clustered into groups with large margins. Colored shape on the bottom left corner of each sample image indicates the ground truth label, which is unknown for the algorithm. (b) User feedback: The user specifies the must-link (green lines) and cannot-link (red lines) constraints. (c) Cluster refinement: Clusters consistent with must-link and cannot-link constraints are regenerated.
$G = \{g_m\}_{m=1}^M$ where $g_m \subset \{1, 2, ..., N\}$ indicates the indices of samples that must be assigned to the same cluster as indicated by user. Similarly, the cannot-link constraints are represented using a set of pairs $C = \{(p, q)\}$ where $(p, q) \in \{1, 2, ..., N\}$ indicate indices of examples that must not be assigned to the same cluster. In addition to the cluster labels $\mathcal{Y} = \{y_{it}\}$, a set of new binary variables $\mathcal{E} = \{e_{mt}\}$ for each group and cluster is defined.

Our proposed clustering framework is defined as the optimization:

$$\min_{\mathcal{W}, \mathcal{Y}, \mathcal{E}, \xi_{ir} \geq 0} \lambda \frac{1}{2} \sum_{t=1}^{K} ||w_t||^2 + \frac{1}{K} \sum_{i=1}^{N} \sum_{r=1}^{K} \xi_{ir}$$  

$$\text{s.t.} \sum_{t=1}^{K} y_{it} f(x_i; w_t) - f(x_i; w_r) \geq 1 - y_{ir} - \xi_{ir} \quad \forall i, r$$  

$$\sum_{t=1}^{K} y_{it} = 1 \quad \forall i, \quad \sum_{t=1}^{K} e_{mt} = 1 \quad \forall m$$  

$$y_{it} \in \{0, 1\} \quad \forall i, t, \quad e_{mt} \in \{0, 1\} \quad \forall m, t$$  

$$L \leq \sum_{i=1}^{N} y_{it} \leq U \quad \forall t$$  

$$y_{pt} + y_{qt} \leq 1 \quad \forall (p, q), t$$

**Objective Function:** In this formulation $\mathcal{W} = \{w_t\}_{t=1}^{K}$ contains the parameters of the model. The slack variables $\xi = \{\xi_{ir}\}, i \in \{1, \ldots, N\}, t \in \{1, \ldots, K\}$ allow a soft margin, and constant $\lambda$ controls the trade-off between the slack variables and the margin. The objective function (Eq. 3.5) and the constraint in Eq. 3.6 optimizes the parameters of the clustering model $f(x_i; w_t)$, and the cluster assignment variables $\mathcal{Y}$ and $\mathcal{E}$ such that the margin between the score of the assigned cluster for each sample and its score for any other cluster is maximum. Here, $f(x_i; w_t) = \max_h [w_t^T \phi(x_i, h)]$ represents the score of assigning the example $x_i$ to the cluster $t$, which is computed using the best configuration of latent variables. The feature vector for example $x_i$ with a latent variable configuration $h$ is denoted by $\phi(x_i, h)$. $y_{it} = 1$ denotes that the example $x_i$ belongs to the cluster $t$, $y_{it} = 0$ otherwise. Similarly $e_{mt} = 1$ denotes that the must link group $g_m$ belongs to the cluster $t$, $e_{mt} = 0$ otherwise.

**Assignment Constraints:** The constraints in Eqs. 3.7 and 3.8 enforce the instances (or a whole must-link group) to necessarily be assigned to a cluster and only one cluster.
**Cluster Balance:** The constraint in Eq. 3.9 avoids a degenerate solution to the optimization problem, where all the data points are grouped into one cluster that has infinite margin with other clusters. This constraint sets upper ($U$) and lower ($L$) bounds on the size of the clusters and can further enforce balanced clusters.

**Must-Link Constraints:** The constraint in Eq. 3.10 ensures that all instances in a must-link group have the same cluster label. Note that here the same must-link group assignment variable $e_{ml}$ is shared between all instances of a group.

**Cannot-Link Constraints:** The constraint in Eq. 3.11 enforces that two cannot-link instances are not assigned to the same cluster. Assuming $(p, q)$ represents two cannot-link samples, if they were assigned to the same cluster, we would have $y_{pt} + y_{qt} = 2$ for at least one cluster.

### 3.2.2 Optimization

We use an alternating descent algorithm to solve the optimization problem in Eq. 3.5 considering the constraints defined in Eqs. 3.6-3.11. This minimization involves solving for unknown latent variables $h$ and cluster assignments $y_{it}$, and then revising estimates of parameters $w$. We use the non-convex regularized bundle method (NRBM) [8]. Details of the initialization strategies are described in the experimental results.

We can obtain the set of must-link and cannot-link constraints iteratively from a user. In the first iteration, a clustering of interactions is generated with no supervision, i.e. without considering any constraint of this type. The initial clustering is presented to a user to obtain his/her feedback. The feedback is modeled as additional constraints, as described above. Then, the samples are clustered again in the next iteration to generate new groups of human interactions that reflect the cumulative user-provided feedback in all previous iterations. By iteratively clustering and obtaining feedback one can construct a pure clustering of data with no incorrectly clustered samples. In the experiments section we will show that this can be achieved with a small amount of user feedback.

### 3.3 Relaxed Latent Maximum Margin Clustering

We also developed a method to discover different groups of data by modifying the assignment constraint of latent max-margin clustering. This objective of this method is to
find different clusters with large margin and represent a few samples from each. The methods work well when the number of data is large but the goal is to only discover different types of clusters, which in our case is to find different types of human interactions.

By Eq. 3.2, which is the first constraint, the clusters are enforced to have a lower bound on the margins between them. In other words, the score of an instance $x_i$ in any cluster is lower than the score of $x_i$ in the assigned cluster. Equation 3.3 enforces the instances to necessarily be assigned to a cluster and only one cluster. We later on relax this constraint to have some unlabeled data, which can be helpful in the case of outlier data points or inhomogeneous clusters.

Without the last constraint, Eq. 3.4, the optimization problem has a degenerate solution, where all the data points are grouped into one cluster that has infinite margin with other clusters. One of the ways to avoid this is to set upper and lower bound on the size of the clusters. This further enforces balanced clusters, which is not good in all cases.

As we mentioned above, by Equation 3.3 each instance must be assigned to one of the $K$ clusters. However, we relax the necessity of assignment of each instance to a cluster. In other words, each example is either assigned into one cluster or none. The main idea is to find the small portions of data, as clusters, which have the maximum margin. These data are far away from each other and are likely to be outliers. Since the number of outliers is small compared to the size of data, we set a small lower bound for the size of clusters. We can omit the upper bound constraint in this case.

We formulate the final optimization, including latent variables, as:

$$\min_{W,Y,\xi \geq 0} \frac{1}{2} \sum_{t=1}^{K} ||w_t||^2 + \frac{C}{K} \sum_{i=1}^{N} \sum_{r=1}^{K} \xi_{ir}$$

s.t. $\forall i, r \sum_{t=1}^{K} y_{it} \max_h \left[ w_t^T \phi(x_i, h) \right] - \max_h \left[ w_r^T \phi(x_i, h) \right] \geq 1 - y_{ir} - \xi_{ir}$, \hspace{1cm} (3.13)

$\forall i, t \ y_{it} \in \{0, 1\}$, $\forall i \sum_{t=1}^{K} y_{it} \leq 1$ \hspace{1cm} (3.14)

$\forall t \ L \leq \sum_{i=1}^{N} y_{it}$ \hspace{1cm} (3.15)

In the above formulation, there is no constraint that penalizes the existence of instances with no cluster assignment. Therefore, by optimizing the problem we try to set instances out of the clusters as much as possible. The only limitation is Eq. 3.15, which is the minimum
size of each cluster. So the solution is $K$ clusters of size $L$. To find the outliers we set $L$ to be a very small portion of the size of data (e.g. $0.05N$). On the other hand we don’t need the upper bound for the size of the clusters. We also set $K$ to be a small number (e.g. $K = 3$ or $K = 2$) because eventually I’ll label $KL$ instances as outliers and remove them from data. So if we set $KL$ to a large number we might lose some meaningful data.
Chapter 4

Features and Implementation Details

We develop methods for clustering human actions according to their interactions. The framework outlined in Chap. 3 is a general-purpose approach that could be used in a variety of settings for analyzing human interactions. For concreteness, we evaluate our algorithm for human interaction clustering on three standard datasets – UT-Interaction [34], Collective Activity [6], and VIRAT [27]. UT-Interaction and Collective Activity are standard datasets, providing well-defined sets of activity classes for measuring clustering performance. VIRAT contains a larger, more diverse set of potential interactions between humans or between humans and vehicles. It provides an excellent domain on which to evaluate algorithms’ abilities to discover classes of interactions that are not defined a priori.

We utilize feature representations appropriate to each dataset. For the Collective Activity Dataset, we analyze the human detections in a frame, and cluster video frames according to the group activity present. We describe each frame using an existing method that represents the appearance of person in a scene using HOG features [20]. These HOG features are classified into categories of pose/action, the values of which are treated as latent variables in the clustering model.

For clustering human trajectories in the VIRAT and UT-Interaction datasets, we develop a set of features including relative position/velocity and appearance. These are augmented with a latent variable representation that handles temporal alignment. Details of these features are provided next.
4.1 Proximity Features

Given a set of trajectories of people in a scene, we wish to build a representation for their interactions. We assume we have trajectories for the people and objects of interest (e.g. vehicles) in a scene. Different classes of interaction will likely have stereotypical patterns of proximity. For instance, a crowd of people might stand together, engaged in a conversation. Two people might walk together across a scene. A solitary person might approach a parked vehicle. We build a representation that captures the relative positions and velocities of people in a scene in order to differentiate between these types of categories of interaction.

We build a representation for each person in a scene. Focusing on one person, we examine his positions and movements with respect to other people and vehicles in the scene. We use a representation that only examines the focal person and the one person and one vehicle that is closest to that focal person over the course of a trajectory. For that one person or vehicle, we build a histogram representation that captures the relative position and velocity of the focal person with respect to the other.

The histogram representation requires choosing a quantization with respect to relative velocity and distance. In order to reduce dependence on an a priori specification of these bin edges, we use an unsupervised approach. We collect sets of samples of relative velocities and distances across a dataset, and then build either a mixture of Gaussians model or a percentile-based representation in order to construct the histogram representation. Each sample point from a respective trajectory of person or vehicle is encoded according to its responsibility under each component of the mixture of Gaussians or its membership in a percentile range.

More precisely, for a person trajectory $x$, we define features for velocity $\phi_v(x)$ and relative distance $\phi_d(x)$. We omit further differentiation of person/vehicle for clarity, but in experiments we add copies of these features for both where applicable. We define:

$$\phi_v(x) = [\gamma_1(v(x)), \gamma_2(v(x)), \ldots, \gamma_K(v(x))]$$

(4.1)

where $v(x)$ is the magnitude of the velocity for person $x$, estimated via finite differences between the start and end locations of the track. These velocities are fed into a mixture of Gaussian model. $\gamma_i(v) = P(z = i|v)$ is the responsibility, the probability that observation $v$ is generated by the $i^{th}$ component of a mixture of Gaussian model with latent mixture
variable $z$. These responsibilities are concatenated into a vector representing the velocity feature.

We use a similar representation for distance, building a separate histogram model over the relative distances. The bin edges are set as percentile ranks of relative distances between people and/or vehicles in the dataset, in order to capture “close”, “far” type relations (precise details appear in Chap. 5). We define:

$$\phi_d(x) = \frac{1}{|T|} \sum_{t \in T} [bin_1(d_{x,t}), bin_2(d_{x,t}), \ldots, bin_K(d_{x,t})]$$ (4.2)

where $d_{x,t}$ is the relative distance between person $x$ and its nearest person trajectory at time $t$. The set $T$ is the set of times for which the trajectory $x$ exists. The functions $bin_k(\cdot)$ are indicators for whether the distance falls into the corresponding percentile range. These distance bin occupancies are summed over time, and concatenated into a feature vector.

### 4.2 Appearance Features

Beyond relative positions and movements, the appearance of a person can capture information about the type of interaction occurring. We augment the track-level proximity features with appearance features based on histogram of oriented gradients (HOG) and histogram of oriented flow (HOF) features. In the UT-Interaction dataset, we use a mixture of Gaussians model to represent appearance. We concatenate the HOG and HOF features into a single feature vector and then train a mixture of Gaussians model. Again, each frame is represented by its responsibilities under this model, and summed over time to create a representation for the trajectory.

### 4.3 Latent Variables for Temporal Alignment

The aforementioned features describe a trajectory via a combination of distance, velocity, and appearance features. However, a challenge when attempting to cluster person trajectories is alignment between different tracks. Global histogram-type features of this type can be used to represent trajectories. Yet this type of representation will suffer from a lack of alignment between features for different tracks. For instance, a person might spend a portion of a trajectory standing still, before engaging in an interaction. The precise
start or end points of this period of motion are variable, and can be modeled with a latent variable.

Consider the examples in Fig. 4.1. Trajectories of people can exist for different lengths of time – due to entry/exit of people from a scene or difficulties in tracking. Further, the relevant activities of a person may occur at different times along different trajectories. For instance, a person might spend a portion of a trajectory standing still, engaged in a conversation, before walking away. The precise start or end points of this period of motion is variable, and can be modeled with a latent variable.

In order to account for these differences, we modulate the track features defined above with latent variables that can be used to align the features of different trajectories. We include latent variables to offset the temporal range on which relative distance and velocity features are defined. For instance, we define the relative distance feature as:

\[
\phi_d(x, h) = \frac{1}{|T(h)|} \sum_{t \in T(h)} [\text{bin}_1(d_{x,t}), \text{bin}_2(d_{x,t}), \ldots, \text{bin}_K(d_{x,t})]
\]  \hspace{1cm} (4.3)

where \( T(h) \) now denotes a time-shifted temporal window controlled by the value of the latent variable \( h \). The distance feature now accumulates relative distances in a time range that can be offset. Similar modifications are performed on other features.
Figure 4.1: Latent variables for aligning tracks. Two example trajectories are shown in time-lapse form top and bottom. Each contains a person approaching a vehicle. However, there are differences in the behavior of each person over the course of the track, with substantial variation in the amount of time the person lingers in front of the vehicle, and what he does afterwards. Latent temporal regions (in green) can be chosen to align these trajectories and extract more similar feature representations.
Chapter 5

Experiments

**Performance measure:** We measure clustering performance using *purity*, a standard measure which evaluates accuracy of the most frequent class in each cluster. In each cluster if we assume the points that have the same label as the most frequent class are *correctly* labeled, then the purity is the ratio of all correctly labeled points to the total number of points. Note that purity is analogous to classification accuracy in a setting where the number of clusters equals the number of ground truth classes.

**Initialization:** For the first iteration, which is fully unsupervised, we initialize our clustering algorithm with a weight vector with all weights set to 1. This produces a set of clusters, then we obtain feedback from the user and add the constraints to our clustering algorithm. For the next iteration, we initialize the algorithm with the weight vector that we obtained from the previous iteration. We do this iteratively until we reach 100% purity.

5.1 Datasets

5.1.1 UT-Interaction Dataset:

The UT-Interaction Dataset [34] contains 2 sets of videos containing pairs of people interacting with each other. Set 1 is captured in a parking lot with a stationary background. Set 2 is captured on a grassy lawn with slight background movement and some camera jitter. Each set contains 10 video sequences with at least one occurrence of each of 6 categories of interaction: *shake hands, hug, kick, point, punch, and push*. We use the
classification version of the dataset, and run automated human detection [7] and track-
ing [33] to obtain trajectories of the two people involved in each interaction. Set 1 exhibits
scale variation, and the scale of the humans in each sequence is automatically estimated
from human detection results. We compute velocity, distance, and appearance features for
each person (Chap. 4).

We use two different latent variables in our experiments on UT-Interaction. The first is
a temporal alignment latent variable that chooses the best 20 frame long temporal window
from a track. The second latent variable models who is playing which role in an interaction –
for example in a pushing interaction, one person is the pusher, and the other the “pushee.”
A latent variable is used to swap the roles of the two people in the feature vector. Since
the UT-Interaction dataset is cleanly structured, with each interaction coming from one of 6
categories, we cluster the tracks into 6 groups. We conduct experiments using a variety of
values for parameter $\lambda$ in the set of $\{10^{-3}, 10^{-2}, ..., 10^2, 10^3\}$, and the best purity is selected.
We set lower bound ($L$) and upper bound ($U$) of clusters to 0.9 and 1.1 of average cluster
size respectively.

Features: We compute velocity, distance, and appearance features for each person.
For velocity features, we split a track in half, and compute the velocity in each half of the
track via finite differences between start and end position. We fit a 3-component Gaussian
mixture model to all the velocities in all tracks, an represent the 2 (first half, second half)
velocities of each track using the responsibilities under each of the 3 components. This
results in a 6-dimensional representation for velocity for each person.

We perform similar modeling for distance features, computing a histogram of relative
distances between the two people in an interaction. We use 3 bins for this histogram. We
quantize distances into “very close”, “near”, and “far” bins based on thresholds that are the
$50^{th}$ and $80^{th}$ percentile of all distances. These counts are aggregated over time for each
interaction to form a histogram.

For appearance features we use HOG and HOF features for each person. Both HOG
and HOF are computed in non-overlapping cells of 8-by-8 pixels. We fit a Gaussian mixture
model with 50 components, and again use the responsibilities under each component as a
representation for each frame, and aggregate these into a feature vector by averaging over
each trajectory.
Latent Variables: We use two different latent variables in our experiments on UT-Interaction. The first is a temporal alignment latent variable that selects a temporal sub-window of the trajectory, and computes features on it. This latent variable can model different temporal offsets (when does an interaction occur). The latent variable chooses the best 20 frame long temporal window from a track. The second latent variable models who is playing which role in an interaction – for example in a pushing interaction, one person is the pusher, and the other the "pushee." A latent variable is used to swap the roles of the two people in the feature vector.

5.1.2 Collective Activity:

This dataset contains a total of 44 short video clips recorded by consumer camcorders. In each video, people are annotated every ten frames, and labeled as one of the following five categories: crossing, waiting, queuing, walking, and talking. The label of each frame is assigned according to the dominant activity of people in that frame. The features are obtained from [20]: from each activity category one third of the videos are taken to be clustered using our model, and the rest used to for the joint action/pose classifiers that are used as features.

Each person can have one of the following eight pose categories: right, front-right, front, front-left, left, back-left, back and back-right. We assign an action label to each person according to his/her pose and activity. Therefore, there are forty different action labels (e.g. crossing front-left). These action labels are latent variables and our algorithm automatically assigns them to people. We cluster the scenes into $K = 6$ clusters. In our experiment we tried a wide range of values for $\lambda$ in the set of $\{10^{-3}, 10^{-2}, ..., 10^{2}, 10^{3}\}$ for both the first iteration of our algorithm and MMC. We used the best purity for comparison. Lower bound ($L$) and upper bound ($U$) are set to 0.6 and 1.4 of average cluster size, respectively.

5.1.3 VIRAT Dataset:

The release 2 of VIRAT Ground dataset [27] contains more than 8 hours of videos captured by surveillance cameras from 12 different scenes. The ground truth annotation contains rare human-vehicle interactions designed for detection tasks in surveillance settings. However, in this work we are interested in discovering other types of interactions
such as human-human interactions in addition to human-vehicle interactions. Therefore, we defined a new set of labels and manually labeled a portion of the dataset. The label set contains: *talking to a person*, *interacting with a car*, *walking alone*, *walking with a person*, and *standing alone*.

In this dataset, we focus on scene 0001, viewing a parking lot. We used a state-of-the-art tracking algorithm [46] to automatically extract 90 human/vehicle tracklets of length 12 seconds (80 humans and 10 vehicles) from manual initializations. We formed the ground truth by labeling the human tracklets based on their interaction with other people or vehicles.

We compute distance and velocity features over each quarter of each tracklet, i.e. temporally binning features with 4 replicates. The dataset contains scenes with multiple people and vehicles present at once. For each focal person, we find the one person and one vehicle with shortest median distance to the focal person, which are considered as the closest person and vehicle to the focal person, respectively. Distance features are computed with respect to this vehicle and person pair. Latent variables for temporal alignment are used in a sliding window fashion, choosing a 6 second long sub-region within the tracklet. We set the number of clusters $K = 5$, lower bound $L = 0.4$, and upper bound $U = 1.6$. We use the best purity $\lambda$ in the set $\{10^{-3}, 10^{-2}, ..., 10^2, 10^3\}$ for each method.

### 5.2 Results

#### 5.2.1 Fully unsupervised (iteration-0):

In the first round of the process, we leverage the features and latent model to cluster data into groups with large margin. Other than the number of clusters, there is no supervision in this step. And the information that we have on the number of clusters is not considered as supervision. In clustering algorithms, user has the option to choose the number of clusters and see different results. Our experiments show that the clustering results of this step are better than baseline clustering methods. We compare our method with *K-means*, *Spectral Clustering*, and *Max-Margin Clustering*. Fig. 5.1 shows the results in terms of purity. Our method works significantly better than the common baselines. For
instance, on the Collective Activity dataset, among baselines MMC achieved highest purity, 76.84%. While our method produces clusters with 80.59% purity.

Fig. 5.2 shows confusion matrices, which illustrates how our method distinguished between different types of interaction. The confusion matrix of VIRAT, Fig. 5.2(d), shows that in the first unsupervised step, our algorithm failed to distinguish between two classes of "people standing alone" and "people walking alone". As a result, most of the samples of these two classes are assigned to one cluster. This might have happened because of the less variation of these two type of activities. But in the next step, our method separates these two classes into two different clusters. This can be done using the cannot-link constraints provided by the user.

Figure 5.1: Performance of our clustering method compared to other baseline methods on different datasets. For all datasets, our proposed method generates clusters of high purity without any supervision on activity labels.

5.2.2 User feedback:

After the first iteration, a user is asked to look through the clusters and choose a small group of dominant interactions from each cluster. These form the must-link constraints. Then we ask the user to select a few mis-clustered interactions from each cluster and put them in their corresponding groups that are chosen in the previous step. These form the cannot-link constraints. Note that if the user doesn’t provide misclustered points for some clusters, we consider them as pure clusters and don’t break them in the next iteration.
Figure 5.2: Confusion matrices of clusters generated at iteration-0 for UT Interactions Set 1 (a), Set 2 (b), Collective Activity dataset (c), and VIRAT(d).
We used the ground truth labels to mimic the user feedback. In each iteration, \( m \) interactions are selected uniformly at random from the dominant interactions of each cluster. Those form the \textit{must-link} constraints. We chose \( m = 5 \) for Collective Activity and UT and \( m = 8 \) for VIRAT, which can practically be done by a real user. Then, we randomly select up to \( c \) interactions from the misclustered interactions, form \textit{cannot-link} constraints, then add them to their corresponding groups. We set \( c = 5 \) for Collective Activity and \( c = 2 \) for UT and VIRAT.

Correcting the label of misclustered interactions will increase \textit{purity}, since the total number of correctly clustered points will be increased. In order to demonstrate how our method is capable of generalizing the user-feedback to incorrectly labeled interactions, a baseline method called \textit{Manually labeled} is also defined that represents the purity of clusters after correcting the misclustered interactions solely based on the feedback.

Fig. 5.3 shows the average performance of our method over 10 runs with different random samplings at each iteration. The results show that our proposed method generates pure clusters after a few iterations of obtaining user-feedback on clusters that were originally generated with zero supervision (i.e. iteration-0). The comparison of our method with the manually labeled baseline also demonstrates how our method can generalize the user-feedback to mis-clustered samples. Error bars show the standard deviation over the 10 runs.

\textbf{Running time:} Our proposed clustering algorithm takes only a few seconds to cluster data given user feedback. The average clustering time per feedback iteration for each dataset is as follows: VIRAT: 2 seconds, Collective Activity: 7s, and UT-Interaction: 18s on a Intel Core i7 CPU (@ 3.40GHz) in a MATLAB implementation.

\textbf{User effort:} The number of data points corrected by the user in each iteration is small. On average the number of misclustered points that are labeled by the user in each iteration is \( 9.5 \pm 3.2 \) for VIRAT, \( 9.6 \pm 6.4 \) for Collective Activity, and \( 4.5 \pm 4.9 \) for UT. Note that, overall, this corresponds to a small amount of labeling compared to the dataset size. For instance, for the Collective Activity dataset the total number of these annotations over all iterations is only 15\% of the whole dataset on average.
Table 5.1: Summary of VIRAT clustering user study results. For each scene, the mean percentage of user responses choosing a video in the same cluster, deeming both to be similar, choosing the out of cluster video, or neither is reported. Standard deviations over the four users is reported.

<table>
<thead>
<tr>
<th>Scenes:</th>
<th>In cluster (%)</th>
<th>Both (%)</th>
<th>Out of cluster (%)</th>
<th>Neither (%)</th>
</tr>
</thead>
<tbody>
<tr>
<td>0401</td>
<td>67.71 ± 2.08</td>
<td>27.08 ± 7.22</td>
<td>4.17 ± 3.40</td>
<td>1.04 ± 2.08</td>
</tr>
<tr>
<td>Average</td>
<td>54.24</td>
<td>24.58</td>
<td>15.83</td>
<td>5.35</td>
</tr>
</tbody>
</table>

5.2.3 Relaxed Latent Max-Margin Clustering:

We conducted experiments on VIRAT dataset to justify our method. We cluster trajectories from each scene independently, into $K = 5$ or $K = 6$ clusters depending on the number of trajectories in the scene. We utilize the relaxed version of latent MMC, and set the lower bound such that each cluster has at least 6 members, and fix parameter $\lambda = 10^3$.

Qualitative results of clusters obtained appear in Fig. 5.4 and Fig. 5.5 (videos in supplementary material). We are able to discover clusters that seem to correspond to categories such as: people walking alone far from vehicles or other people, people walking with another person but far from other vehicles, people approaching a car, people standing close to another person, people standing alone, and people standing close to a vehicle and another person.

In order to quantitatively measure performance, we conducted a small user study. Four users were shown one interaction (A) and asked which of two other interactions (B, C) was most similar. Options to select “both” or “neither” were also presented. The interaction A was from one of the obtained clusters, as was one of B and C. The other of B and C was randomly chosen from among all other videos not in the same cluster as A.

The user responses can be used as an estimate of the purity and recall of the clustering method. Responses choosing the option from the same cluster, or “both” reinforce purity of the discovered clusters, while choices strictly of the same cluster represent high purity and recall.

A summary of the user responses is shown in Table 5.1. A clear tendency to prefer videos in the same clusters our algorithm chooses is exhibited in the user responses, with roughly 80% of videos deemed to belong to the same cluster as the reference video.
Figure 5.3: Average purity of the proposed clustering model for UT-Interaction Set 1 (a), Set 2 (b), Collective Activity (c), and VIRAT (d). Our method constructs 100% pure clusters after a few iterations of obtaining user-feedback. Our performance is significantly better than the manually labeled baseline that uses the feedback for correcting misclustered interactions that are generated in iteration-0 with zero supervision.
Figure 5.4: Examples of VIRAT clustering results. Each rectangle shows a set of three sequences from a single cluster.
Figure 5.5: Examples of VIRAT clustering results. Each rectangle shows a set of three sequences from a single cluster.
Chapter 6

Conclusion and Future Work

We proposed a method for discovering human interactions in video sequences based on unsupervised learning combined with user feedback. The method operates on trajectories of people, and reasons about their interactions with other people and/or vehicles present in a set of videos. We use feature representations that allow the model to account for alignment of trajectories extracted from different parts of a video and the actions of individual people. A novel variant of latent max-margin clustering was developed to discover clusters in an iterative fashion, including user feedback at each iteration.

The method shows promise for automatically discovering the types of interactions that occur in a scene. On the standard UT-Interaction and Collective Activity datasets, the purely unsupervised approach obtains cluster purity that is close to methods based on supervised classification. On the large VIRAT corpus, a varied set of human-human and human-vehicle interactions were discovered. A small number of iterations of limited user feedback results in perfectly pure clusters of human interactions, demonstrating a promising alternative to supervised approaches for human interaction analysis. In the rest of this chapter, we will briefly explain some of the limitations that we encountered and also some directions for future work.

6.1 Limitations

In this work we demonstrated the effectiveness of our method for discovering human interactions in videos, providing quantitative results. However, there are some limitations to this method. For instance, In Sec 3.2.2 we explained that we use an alternating descent
algorithm to solve the optimization problem. The process works in an iterative fashion that performs two steps in each iteration. First, it solves the minimization that leads to find latent variables and cluster assignments. Then, parameters $w$ is revised based on the results of previous step. The first step is a NP problem, which can be solved using Integer Linear Programming methods. This is still NP and is not trivial. Our approach to address this problem is to use LP optimization and then round the results to get integer results. This method works most of the time but it is possible that labeling assignments after "rounding" process violate some of the constraints.

Feature representation is a prominent factor in obtaining good results. In this work we used different types of features to represent the interactions. Different features produce different results depending on the correlation of features to the model. Since we design the model we have some clues on which features we should choose. But there is no absolute way or deterministic method to choose the best features that produce best results.

6.2 Future Directions

In Sec. 3.2 we explained how user feedback is incorporated into clustering method to refine the clusters. One of the steps in obtaining the user feedback is to show the clustering results to user and ask him/her to provide the requested information. However, it still requires a lot of user effort to look through the results and select the mis-clustered data. There is a possible line of work here which is related to the area of active learning. In active learning, methods works interactively with the user to obtain better results. In the process, the algorithm query the user to provide some information such as labeling. Queries can have different types, for example, labeling most uncertain data, labeling points that would most change the model, labeling points that would minimize the variance, and etc. As mentioned in the introduction, a lot of work has been done related to this area but in supervised methods. However, in unsupervised approaches there is still room for exploring such methods. In our case, instead of asking user to look through all the data and select the desired points, we can possibly choose the data that would most affect the results.
Bibliography


36


