in Proc. 13th International Conference on Computer Vision (ICCV), Barcelona, Spain, 2011

A Chains Model for Localizing Participants of Group Activities in Videos

Mohamed R. Amer and Sinisa Todorovic
Oregon State University
amerm@onid.orst.edu, sinisa@eecs.oregonstate.edu

Abstract

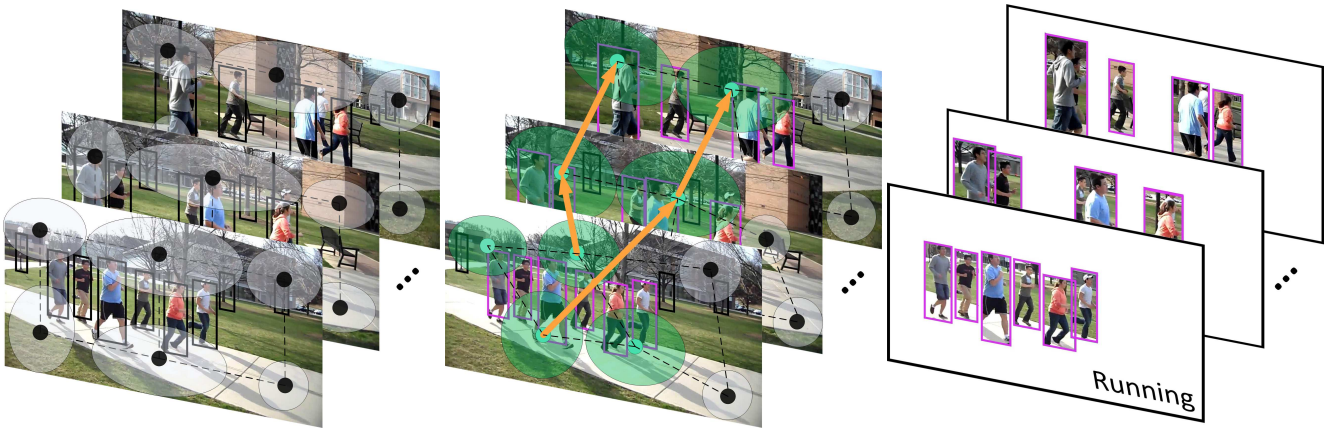
Given a video, we would like to recognize group activities, localize video parts where these activities occur, and detect actors involved in them. This advances prior work that typically focuses only on video classification. We make a number of contributions. First, we specify a new, mid-level, video feature aimed at summarizing local visual cues into bags of the right detections (BORDs). BORDs seek to identify the right people who participate in a target group activity among many noisy people detections. Second, we formulate a new, generative, chains model of group activities. Inference of the chains model identifies a subset of BORDs in the video that belong to occurrences of the activity, and organizes them in an ensemble of temporal chains. The chains extend over, and thus localize, the time intervals occupied by the activity. We formulate a new MAP inference algorithm that iterates two steps: i) Warps the chains of BORDs in space and time to their expected locations, so the transformed BORDs can better summarize local visual cues; and ii) Maximizes the posterior probability of the chains. We outperform the sate of the art on benchmark UT-Human Interaction and Collective Activities datasets, under reasonable running times.

1. Introduction

This paper is about detecting the start and end frames of group activities in videos, and localizing all their participants. This is challenging because group activities are typically characterized by large variations in motions, appearances, and spatiotemporal layouts of actors involved. For example, running in a group can be performed by a varying number of people, in diverse, dynamically changing spatial configurations, in which the runners may partially occlude one another. To address these challenges, we seek answers to the following questions: what video features should be extracted, and how to model group activities for efficient inference and robust learning?

Regarding activity features, we depart from the common practice to extract a set of (arguably good) features, and then conduct inference on this fixed set, without ever revisiting the video. We believe that the complexity of group activities requires a more synergistic interaction between high-level inference algorithms and low-level feature extractors than seen in existing work. We here formalize this interaction as an iteration in which inference guides low-level algorithms in their search for optimal features, while adaptive feature extraction facilitates inference in the face of many competing hypothesis. Instrumental to this is our new, mid-level, video feature, referred to as a bag of the right detections (BORDs). BORDs are aimed at summarizing their space-time neighborhoods by histograms of visual cues that are relevant for recognition of the target activity. Specifically, BORDs are histograms of the right detections of people, who are believed to participate in the target activity, where the detections occur amidst the background clutter and other people who do not participate in the activity. BORDs can be interpreted as spotlights that shed light on certain space-time voxels in the video. If the activity occurs, it has to be in the spotlight of many BORDs so that they can collectively provide a strong support of this hypothesis. There are two main differences from other common mid-level features (e.g., Bag of Words, shape context). First, our low-level features, which are used for computing the histogram of a BORD, are not observable, but hidden variables that must be inferred. The histogram is computed from the right detections, not any detections (as in BoW). Second, BORDs are movable, laying on a deformable space-time grid throughout the video. Their optimal locations are informed top-down in inference. The inference algorithm warps the grid so the BORDs could better summarize relevant visual information for activity recognition. The size of pixel neighborhoods associated with every BORD are adaptively re-computed at new video locations.

Regarding activity representation, we specify a new, chains model aimed at organizing BORDs in an ensemble of temporal chains. The chains may have arbitrary length, ideally, beginning and ending at the end-points of time intervals occupied by the activity. As we will show in this paper, our chains model is particularly suitable for representing non-rigid transformations of various spatial config-



(a) (b) (c)

Figure 1. Our approach: (a) BORDs (illustrated as space-time ellipsoids with black centroids) are placed on a deformable grid in the video to jointly collect the most informative visual evidence for activity recognition. (b) The MAP inference of the chains model identifies a subset of BORDs, organized in a number of temporal chains, which jointly localize the spatiotemporal extent of the recognized activity. (c) The chains are adaptively warped, such that the BORDs can better estimate the right detections of people who participate in the activity.

urations of people conducting the group activity. Another chains model has been recently used for detecting a person's hand in a still image [5]. We here relax their restrictive assumption that the feature chains must start from a known reference point. In our case, the start and end frames of the activity occurrences are hidden variables that must be inferred. We specify a new inference algorithm that efficiently guides the extraction of BORDs.

Related work: Graphical models offer a principled computational framework for representation and inference of many important properties of activities, including motion, appearance, and spatiotemporal layout of people involved in the activity. For example, activities have been modeled by dynamic Bayesian nets [15], prototype trees [9], context-free grammars [4, 3], and CRFs [10]. Most of these approaches, however, do not address group activities. They can typically handle only sanitized environments with static background, where actors are prominently featured (with few exceptions, e.g., [3, 10]). We address more realistic videos. A few methods seek to learn temporal structure of activities from data [10], or relevant contextual relations within a group activity [6]. However, their model structure permits only a fixed number of actors, or a fixed number of primitive actions defining the group activity. We overcome these limitations by enabling inference of arbitrarily large numbers of activity participants and primitives. Spatiotemporal match kernels have been used in a a heuristic voting procedure for localizing group activities in the video [12]. By contrast, we localize activities in a principled manner using the MAP inference of our chains model. Spatiotemporal relations of video features within a group activity are modeled in [1]. However, they can only classify videos, whereas we additionally seek to infer the start and end frames of the activity occurrences, as well as detect all their participants. Our BORD is similar to a mid-level feature, called control point, which has been used for detecting and summarizing the right image contours falling along object boundaries, amidst the clutter of other edges in the background [16].

Overview: Our approach consists of two main steps, as illustrated in Fig. 1. BORDs are placed on a deformable grid in the video to jointly collect visual evidence if any activity of interest is present. The MAP inference of the chains model identifies temporal chains of the BORDs. In inference, the chains are adaptively warped, such that their BORDs can better summarize visual information for recognizing and localizing all occurrences of the activity, and for detecting their participants. Specifically, we specify a new MAP inference algorithm that iterates two steps: i) Warps the chains of BORDs to their expected locations in the video; and ii) Maximizes the posterior probability of the chains. Our experiments demonstrate that the proposed approach is robust against transient occlusions, since BORDs are minimally affected when a particular actor gets occluded. This is important because mutual occlusions of actors are frequent in group activities.

In the sequel, we first specify the BORD, and, then, formalize the chains model and its inference and learning. Next, we present our superior performance on challenging benchmark datasets, including UT Human Interactions [13] and Collective Activities [1], relative to the state of the art.

2. Bags of the Right Detections

The BORD is a descriptor associated with spatiotemporal voxels of the video. In particular, the BORD, $\boldsymbol{h}_i$, is a histogram of human poses detected in a space-time neighborhood centered at point i in the video volume. The his-

togram is not computed from all people that are detected in the neighborhood, but only from those detections that are estimated to take part in the target activity. Also, the right placement of points i in the video, and the size of their neighborhoods are adaptively determined from data, such that the BORDs can summarize most informative human poses for activity recognition and localization. Below, we first explain how to detect people and their poses in the video, and, then, specify the BORD.

Given a video, we first run an efficient people detector. Specifically, we use the approach of [2]. Its parameters are set such that the detector yields high recall, under the average running time of 2s per frame. The high recall ensures that all activity participants are detected in the video. Then, discarding false positives will be delegated to the higher-level inference algorithm. The resulting detections are bounding boxes, which represent our noisy input. In the sequel, we will use the terms detector responses, detections, and bounding boxes, interchangeably.

Next, we identify a characteristic human pose for every detection. In particular, the detector that we use localizes human-body parts (e.g., legs, arms) within each detected bounding box [2]. The spatial layout of these body parts can be used to define a descriptor of human poses, in terms of: i) Part distances and orientations relative to a reference body part; and (ii) Mean optical flow vector within the bounding box, computed by the approach of [14]. We map each human-pose descriptor to a dictionary of codewords, where the words represent characteristic human poses. Our dictionary consists of $d = 300$ characteristic poses, learned by the K-means algorithm on training videos with labeled bounding boxes around people performing various group activities of interest. Note that many other alternative definitions of the human-pose descriptor can be specified, when a particular people detector used does not localize body parts.

Given noisy people detections and their characteristic poses, the BORD at point i seeks to identify the right detections, which belong to the target activity, and compute the histogram $\boldsymbol{h}_i$ of their associated human poses, within a spatiotemporal neighborhood of i. To compute $\boldsymbol{h}_i$, we avoid enumerating exponentially many figure/ground labelings of the detections. Rather, we compute $\boldsymbol{h}_i$ as a linear function:

$$\boldsymbol{h}_i = \boldsymbol{V}_i \boldsymbol{b}, \tag{1}$$

where $\boldsymbol{b} \in \{0,1\}^k$ is a hidden variable vector that must be inferred, and $\boldsymbol{V}_i \in \{0,1\}^{d \times k}$ is an observable matrix defining the spatiotemporal neighborhood of i, as illustrated in Fig. 2. Specifically, $\boldsymbol{b}$ is an indicator vector of all k bounding boxes detected in the video. An element of $\boldsymbol{b}$ is set to 1 if the corresponding bounding box is estimated as foreground, or 0, otherwise. Also, an element $(\boldsymbol{V}_i)_{uv} = 1$ if vth bounding box "falls" within i's neighborhood, and the bounding box is mapped to uth human pose in the dictionary. The shape of the neighborhood is defined as a space-time ellipsoid centered at i, whose scale and principal axes are adaptively estimated to fit the spatiotemporal changes of bounding boxes in the vicinity (this is quite similar to the standard estimation of the scale of interest points in images). In our experiments, we observe ellipsoids that span between 2 to 6 frames.

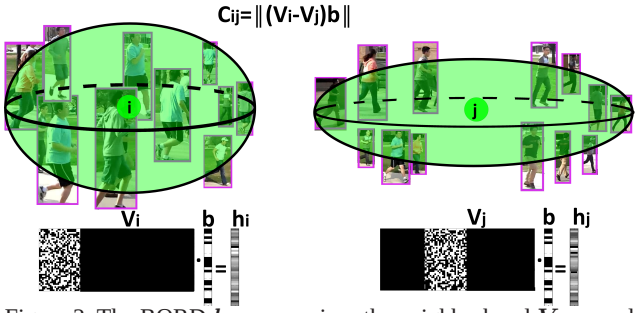


Figure 2. The BORD $\boldsymbol{h}_i$ summarizes the neighborhood $\boldsymbol{V}_i$ around point i in terms of a histogram of the right people detections which are inferred in the indicator vector $\boldsymbol{b}_i$. Dissimilarity between two BORDs, $\boldsymbol{C}_{ij}$, is defined as their Mahalanobis distance.

For activity recognition and localization, we initially sample a number of points, $i = 1, ..., n$, forming a regular spatiotemporal grid. We allow for data-driven, non-rigid deformations of the grid, such that BORDs associated with the grid points may assume optimal locations for collectively providing the accurate information about foreground people detections. Thus, detecting actors in the video, i.e., the inference of $\boldsymbol{b}$, is informed by both local cues in the vicinity of every point i, and contextual cues provided by other points of the grid. The inference of $\boldsymbol{b}$, and the optimal non-rigid transformation of the grid are formalized as the MAP inference of the chains model representing the activity.

3. The Chains Model

The chains model is a generative model of a given activity class. The observable random variables of the chains model include the set of BORDs, $H = \{\boldsymbol{h}_i : i = 1, ..., n\}$, observed in the video. The model accounts that H is partitioned into an ordered, temporal chain of foreground BORDs, $O = (O(1), ..., O(m), ..., O(M))$, $\forall m, 1 \leq O(m) \leq n$, belonging to the activity occurrence, and background BORDs, $H \backslash O$. The hidden variables of the chains model include: (i) the activity's start and end frames, L_S and L_E; (ii) the chain of foreground BORDs O; and (iii) their total number M. The joint probability distribution of all random variables, $P(M, O, L_S, L_E, H)$, is specified as

$$P(M, O, L_S, L_E, H) = P(M)P(O)P(L_S|M, O, H) \cdot \\ \cdot P(L_E|M, O, H) \prod_{m=1}^{M-1} P(\boldsymbol{h}_{O(m+1)}|\boldsymbol{h}_{O(m)}) \prod_{i \in H \backslash O} P_G(\boldsymbol{h}_i). \tag{2}$$

Below, we explain each distribution. M has a Poisson distribution, $P(M = m) = \frac{\lambda_M^m}{m!} e^{-\lambda_M}$. This prevents inference of unrealistically short and very long chains. $P(O)$ is uniform if for all point pairs $(O(m), O(m+1))$ the frame of point $O(m)$ does not happen after the frame of $O(m+1)$, and $P(O) = 0$, otherwise. This prevents the chains to move backwards in time. $P(L_S|M, O, H)$ and $P(L_E|M, O, H)$ are defined as the conditional probability that the chain starts and ends at $1 \leq O(1) \leq n$ and $1 \leq O(M) \leq n$. Specifically, let $t_{O(1)}$ and $t_{O(M)}$ be the frames of the end-points of the chain. Then, we specify that the probability of the start exponentially decreases as $t_{O(1)}$ becomes larger, $P(L_S|M, O, H) = \lambda_S \exp(-\lambda_S \frac{t_{O(1)}}{t_{\max}})$, and, similarly, $P(L_E|M, O, H) = \lambda_E \exp(-\lambda_E \frac{(t_{\max} - t_{O(M)})}{t_{\max}})$. $P(\boldsymbol{h}_{O(m+1)}|\boldsymbol{h}_{O(m)})$ is the transition probability between two consecutive BORDs in the chain. We assume that all legal orderings have a uniform distribution. Thus, in the following, we will simplify notation and write $P(\boldsymbol{h}_j|\boldsymbol{h}_i)$ to denote the probability of transitioning between foreground points. Finally, $P_G(\boldsymbol{h}_i)$ is a uniform distribution that BORD i belongs to the background. In summary, the set of parameters of the chains model is $\Theta = \{\lambda_S, \lambda_E, \lambda_M, P(\boldsymbol{h}_j|\boldsymbol{h}_i)\}$.

4. The MAP Inference

Our MAP inference iterates two steps, as illustrated in Fig. 3. In the first step, we compute the marginal posterior distribution of L_S and L_E over all possible chains, given a set of observed BORDs, $P(L_S, L_E|H)$. In the second step, we re-estimate BORDs by adaptively warping their grid toward maximizing their informativeness. In the following two subsections, we explain each step.

4.1. Maximizing the Marginal Posterior

From (2), the marginal posterior of L_S and L_E is

$$\begin{aligned} P(L_S, L_E|H) &\propto \textstyle\sum_{M,O} P(M, O, L_S, L_E, H), \\ &\propto \textstyle\sum_{M,O} P(M) P(L_S, L_E|M, O, H) \prod_{i,j} P(\boldsymbol{h}_j|\boldsymbol{h}_i). \end{aligned} \tag{3}$$

We compute (3) by organizing all transition probabilities in an $n \times n$ matrix $\boldsymbol{X} = [P(\boldsymbol{h}_j|\boldsymbol{h}_i)]$. Note that each row i of $\boldsymbol{X}$ contains the probabilities of transitioning, in one step, from point i to other points j in the video. The sum of these probabilities along each row of $\boldsymbol{X}$ must be 1, $\boldsymbol{X}\mathbf{1}_n = \mathbf{1}_n$, where $\mathbf{1}_n$ is the n-dimensional vector with all 1's. It follows that the probability of transitioning from i to j in m steps is equal to $(\boldsymbol{X}^m)_{ij}$. In addition, we organize all conditional probabilities $P(L_S, L_E|M, O, H)$, for each control point, into n-dimensional vectors $\boldsymbol{\omega} = [P(L_S{=}1|\cdot), ..., P(L_S{=}n|\cdot)]^{\mathrm{T}}$, and $\boldsymbol{\gamma} = [P(L_E{=}1|\cdot), ..., P(L_E{=}n|\cdot)]^{\mathrm{T}}$. It is straightforward to show that (3) can be computed as

$$P(L_S, L_E|H) \propto \boldsymbol{\omega}^{\mathrm{T}} \left[\textstyle\sum_{m=1}^{n} P(M = m) \boldsymbol{X}^m\right] \boldsymbol{\gamma}, \tag{4}$$

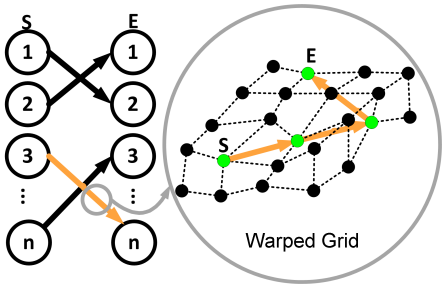


Figure 3. Our MAP inference iteratively finds the start (S) and end (E) frames of the activity through two steps. It: i) Estimates the MAP Markov chains of a given set of BORDs, and ii) Warps the grid of BORDs so they can extract more relevant visual cues for the MAP inference in the next iteration.

Note that the summation in (4) includes all paths of length m – not only simple paths without loops. We resort to this approximation for reducing complexity. Assuming that n is a very large number, $n \to \infty$, the bracketed term in (4) can efficiently be approximated as

$$\sum_{m=1}^{n} P(M = m) \boldsymbol{X}^m \approx e^{-\lambda_M} \sum_{m=0}^{\infty} \frac{\lambda_M^m}{m!} \boldsymbol{X}^m = e^{-\lambda_M} e^{\lambda_M \boldsymbol{X}}. \tag{5}$$

By plugging in the parameters of different activity models, Θ_a, $a = 1, 2, ...$, from (4) and (5), we have that the MAP inference recognizes activity class a^* as

$$a^* = \underset{a=1,2,...}{\operatorname{argmax}} \; \boldsymbol{\omega}_a^{\mathrm{T}} \cdot \exp(\lambda_{M;a} \cdot \boldsymbol{X}_a) \cdot \boldsymbol{\gamma}_a. \tag{6}$$

Also, from (4) and (5), the optimal start and end of activity a^* can be estimated as indices $1 \leq i \leq n$ and $1 \leq j \leq n$ of the maximum product: $(L_S^*, L_E^*) = \operatorname{argmax}_{i,j} \boldsymbol{\omega}_{i;a^*} \cdot \boldsymbol{\gamma}_{j;a^*} \cdot [\exp(\lambda_{M;a^*} \cdot \boldsymbol{X}_{a^*})]_{ij}$. When multiple instances of the activity need to be detected, we choose the second, third, etc. best pair (i, j) for (L_S^*, L_E^*).

From (6), the MAP inference requires $\boldsymbol{X}$. In the next subsection, we explain how to estimate $\boldsymbol{X}$.

4.2. Extracting Optimal Features

Our goal is to estimate optimal locations of BORDs in the video, so their histograms jointly provide the most relevant visual cues for the MAP inference. At different locations, the BORDs include in their histograms different sets of people detections. Thus, when in the optimal layout, the BORDs should jointly include only foreground bounding boxes in their histograms, i.e., correctly provide evidence about people involved in the activity.

Finding optimal locations of BORDs and estimating their transition probabilities $\boldsymbol{X}$ is specified as a constrained optimization, and iteratively solved by alternating two steps. We initially place 16 BORDs in every 5th frame, so they form a regular grid. In the first optimization step,

we deform the grid along the spatial and temporal axes to its expected layout. Then, in the second step, each $\boldsymbol{X}_{ij}$ is estimated as confidence that BORDs i and j are the best one-step transition along the temporal chain representing the target activity. The two steps are iterated until convergence. Below, we explain these two steps in greater detail.

First, given a current estimate of $\boldsymbol{X}$, the expected location of each point i can be found as $\hat{\boldsymbol{q}}_i = \sum_j \boldsymbol{X}_{ij}\boldsymbol{q}_j$. The $\boldsymbol{q}_i$ is defined as a row vector: $\boldsymbol{q}_i = [x_i, y_i, t_i, \cos\varphi_i, \sin\varphi_i]$, where x_i and y_i are coordinates of point i in frame t_i, and φ is the direction of optical flow at i, computed by [14]. Let $\boldsymbol{Q}$ denote an $n \times 5$ matrix whose rows are $\boldsymbol{Q}_{i\cdot} = \boldsymbol{q}_i$, $i = 1, ..., n$. Then, the expected locations of all points can be expressed as $\widehat{\boldsymbol{Q}} = \boldsymbol{X}\boldsymbol{Q}$.

Second, to estimate $\boldsymbol{X}$, we find the best matching pairs of BORDs, such that the matching is invariant to locally affine transformation of the grid. Importantly, as the BORDs change locations in the first optimization step, their histograms "cover" different sets of bounding boxes in the video, and consequently the best matching pairs of BORDs change. In the following, we gradually formalize the matching of BORDs from a simple standard linear assignment problem to the desired complex optimization which allows for non-rigid transformation of the grid.

Using the definition of $\boldsymbol{X}$, given in Sec. 4, the matching of BORDs can be formulated as the linear assignment:

$$\begin{array}{ll}\text{minimize} & \sum_{i=1}^{n}\sum_{j=1}^{n}\boldsymbol{X}_{ij}\boldsymbol{C}_{ij} \\ \text{subject to} & \forall i,j,\ \boldsymbol{X}_{ij} \geq 0,\ \ \boldsymbol{X}\boldsymbol{1}_n = \boldsymbol{1}_n,\end{array} \tag{7}$$

where dissimilarity $\boldsymbol{C}_{ij} = [(\boldsymbol{V}_i - \boldsymbol{V}_j)\boldsymbol{b}]^{\mathrm{T}}\boldsymbol{\Sigma}^{-1}(\boldsymbol{V}_i - \boldsymbol{V}_j)\boldsymbol{b}$. $\boldsymbol{\Sigma}$ is learned for each activity class, and encodes the activity-specific covariance of human poses. We organize all dissimilarities $\boldsymbol{C}_{ij}$ in an $n \times n$ matrix $\boldsymbol{C}_{\boldsymbol{b}}$, where $\boldsymbol{b}$ in subscript indicates that each $\boldsymbol{C}_{ij}$ is a function of $\boldsymbol{b}$. This allows to write the summation in (7) more compactly as $\sum_{i,j}\boldsymbol{X}_{ij}\boldsymbol{C}_{ij} = \mathrm{tr}\{\boldsymbol{C}_{\boldsymbol{b}}^{\mathrm{T}}\boldsymbol{X}\}$.

Since computing $\boldsymbol{C}_{\boldsymbol{b}}$ requires foreground detections, $\boldsymbol{b}$, be estimated, we extend (7) with additional constraints, for conducting minimization with respect to both $\boldsymbol{X}$ and $\boldsymbol{b}$:

$$\begin{array}{ll}\text{minimize} & \mathrm{tr}\{\boldsymbol{C}_{\boldsymbol{b}}^{\mathrm{T}}\boldsymbol{X}\} \\ \text{subject to} & \boldsymbol{X} \geq 0,\ \ \boldsymbol{X}\boldsymbol{1}_n = \boldsymbol{1}_n,\ \ \boldsymbol{b} \geq 0,\ \ \|\boldsymbol{b}\|_2^2 = 1.\end{array} \tag{8}$$

We also wish to constrain points on the grid to preserve their neighbor relations after reallocating to their expected positions $\widehat{\boldsymbol{Q}}$ in the first optimization step. To this end, we define an $n \times n$ adjacency matrix $\boldsymbol{W}$, where each $\boldsymbol{W}_{ij}$ is inversely proportional to the Euclidean distance between $\boldsymbol{q}_i$ and $\boldsymbol{q}_j$, and $\boldsymbol{W}_{ii} = 0$. $\boldsymbol{W}$ is used to extend (8), and additionally minimize expected distances of each point to its neighbors, defined as L1 norm $\|\widehat{\boldsymbol{Q}} - \boldsymbol{W}\widehat{\boldsymbol{Q}}\|_1$, as

$$\begin{array}{ll}\text{minimize} & \mathrm{tr}\{\boldsymbol{C}_{\boldsymbol{b}}^{\mathrm{T}}\boldsymbol{X}\} + \alpha\|(\boldsymbol{I} - \boldsymbol{W})\boldsymbol{X}\boldsymbol{Q}\|_1 \\ \text{subject to} & \boldsymbol{X} \geq 0,\ \ \boldsymbol{X}\boldsymbol{1}_n = \boldsymbol{1}_n,\ \ \boldsymbol{b} \geq 0,\ \ \|\boldsymbol{b}\|_2^2 = 1.\end{array} \tag{9}$$

We also wish to constrain the displacements of points from their original positions, before warping the grid, not only their relative distances to neighbors. This yields our final formulation:

$$\begin{array}{ll}\text{minimize} & \mathrm{tr}\{\boldsymbol{C}_{\boldsymbol{b}}^{\mathrm{T}}\boldsymbol{X}\} + \alpha\|(\boldsymbol{I} - \boldsymbol{W})\boldsymbol{X}\boldsymbol{Q}\|_1 + \beta\|(\boldsymbol{I} - \boldsymbol{X})\boldsymbol{Q}\|_1 \\ \text{subject to} & \boldsymbol{X} \geq 0,\ \ \boldsymbol{X}\boldsymbol{1}_n = \boldsymbol{1}_n,\ \ \boldsymbol{b} \geq 0,\ \ \|\boldsymbol{b}\|_2^2 = 1.\end{array} \tag{10}$$

We here use empirically estimated $\alpha = 0.3$ and $\beta = 0.9$.

To solve (10), we follow the linearization steps presented in [8]. Thus, we introduce auxiliary $n \times 5$ matrices $\boldsymbol{Z}$ and $\boldsymbol{Y}$ to replace the L1-norm constraints in (10) as

$$\begin{array}{ll}\text{minimize} & \mathrm{tr}\{\boldsymbol{C}_{\boldsymbol{b}}^{\mathrm{T}}\boldsymbol{X}\} + \alpha\boldsymbol{1}_{\mathbf{n}}^{\mathrm{T}}\,\mathbf{Z}\,\boldsymbol{1}_{\mathbf{5}} + \beta\boldsymbol{1}_{\mathbf{n}}^{\mathrm{T}}\,\mathbf{Y}\,\boldsymbol{1}_{\mathbf{5}} \\ \text{subject to} & \boldsymbol{X} \geq 0,\ \ \boldsymbol{X}\boldsymbol{1}_n = \boldsymbol{1}_n,\ \ \boldsymbol{b} \geq 0,\ \ \|\boldsymbol{b}\|_2^2 = 1, \\ & \boldsymbol{Z} \geq 0,\ \ \boldsymbol{Y} \geq 0 \\ & (\boldsymbol{I} - \boldsymbol{W})\boldsymbol{X}\boldsymbol{Q} \leq \boldsymbol{Z},\ \ (\boldsymbol{I} - \boldsymbol{W})\boldsymbol{X}\boldsymbol{Q} \geq -\boldsymbol{Z} \\ & (\boldsymbol{I} - \boldsymbol{X})\boldsymbol{Q} \leq \boldsymbol{Y},\ \ (\boldsymbol{I} - \boldsymbol{X})\boldsymbol{Q} \geq -\boldsymbol{Y}\end{array} \tag{11}$$

where minimization is over $\boldsymbol{X}$, $\boldsymbol{b}$, $\boldsymbol{Z}$, and $\boldsymbol{Y}$. From (11), it follows that the matching of BORDs can be efficiently solved as the linear program (LP). An LP with tens of thousands of variables, and thousands of constraints can be solved within seconds on a standard PC using state-of-the-art solvers, such as CVX. The number of variables in our LP model in (11) is proportional to n^2. In our implementation, for 10^3 BORDs, the CVX takes less than 5s on a 2.66GHz, 3.49GB RAM PC.

After finding $\boldsymbol{X}$ and $\boldsymbol{b}$ from (11), the BORDs are moved to their expected locations $\widehat{\boldsymbol{Q}} = \boldsymbol{X}\boldsymbol{Q}$. This, in turn, changes their layout. In the next iteration step, we recompute: i) $\boldsymbol{W}$ to capture new neighbor relations, and ii) histograms $\boldsymbol{h}_i = \boldsymbol{V}_i\boldsymbol{b}$ to account for any new bounding boxes indicated by the new $\boldsymbol{b}$. The new $\boldsymbol{W}$ and $\boldsymbol{C}_{\boldsymbol{b}}$ are then plugged back in (11). The iterations are repeated until changes of the objective of (11) become close to zero. We usually run 5 iterations. Our experiments demonstrate that the approach is relatively insensitive to a specific choice of the initial layout of BORDs, as long as they sufficiently densely populate the video (e.g., our setup of 16 BORDs in every 5th frame).

5. Learning

The model parameters, $\Theta = \{\lambda_S, \lambda_E, \lambda_M, P(\boldsymbol{h}_j|\boldsymbol{h}_i)\}$, are learned from R training videos, provided for each group activity. We assume that the training videos of length $t_{\max}$ are labeled with the start and end frames of the activity occurrences, $\{(t_{S;r}, t_{E;r}) : r = 1, ..., R\}$, and with bounding boxes around participants in regularly sampled frames. Then, we compute $\lambda_S{=}R/\sum_r \frac{t_{S;r}}{t_r}$, $\lambda_E{=}R/\sum_r(1{-}\frac{t_{E;r}}{t_r})$, and $\lambda_M = R/\sum_r \frac{(t_{E;r}-t_{S;r})}{t_r}$. The transition probabilities $P(\boldsymbol{h}_j|\boldsymbol{h}_i)$ depend on the covariance matrix of human poses $\boldsymbol{\Sigma}$. For $\boldsymbol{\Sigma}$, we first run the people detector of [2], and keep only those detections that fall within the labeled bounding

boxes. Then, we map the human-pose descriptors associated with these detections to codewords of the dictionary of characteristic human poses. Finally, each element $\mathbf{\Sigma}_{uu'}$ is computed as the covariance of codewords u and u'.

6. Results

Our approach is evaluated on two benchmark datasets. First, the Collective Activities dataset [1] consists of 75 short videos of crossing, waiting, queuing, walking, talking, running, and dancing. This dataset tests our performance on collective behavior of individuals under realistic conditions, including background clutter, and transient mutual occlusions of actors. For training and testing, we use $2/3$ and $1/3$ of the videos from each class, respectively. The dataset provides labels of every 10th frame, in terms of bounding boxes around people performing the activity, their pose, and activity class. Second, the UT-Interaction dataset [13] consists of 20, 1-minute videos of continuous executions of 6 classes of human interactions: shaking-hands, pointing, hugging, pushing, kicking, and punching. Each video shows at least one instance of every interaction class, where in some cases distinct activities may co-occur. The videos show two scene types: 10 videos are taken on a parking lot, and the other 10 videos are captured in a natural setting by a moving camera. The UT-Interaction dataset presents a number of challenges: the jittery camera motion, simultaneous performance of several activities, activities may begin and end at arbitrary times, presence of people who are not involved in the activity, etc. The dataset provides ground truth labels in terms of time intervals of the activities, and bounding boxes around all actors. Our evaluation setup is the same as that presented in [12]. Specifically, for training, we use 20% of the available manual segmentations of the videos into 60 intervals, each occupied by a unique activity instance. We test on the full (unsegmented) sequences.

We test different aspects of our approach through four variants. **Var1** is our default. It runs the MAP inference on the optimally warped grid of BORDs, which seek to identify the right participants of the activity in the clutter of people detections obtained by the detector of [2]. We set a low threshold of -3 for this detector that gives 93% false positives on the UT-Interaction dataset. **Var2** serves to evaluate using BORDs as mid-level activity features versus directly using lower-level, noisy people detections for activity recognition. Var2 has the same steps as Var1, except that instead of the BORDs we run our MAP inference directly on bounding boxes detected by the people detecter of [2], and described by the pose descriptor. Note that we cannot search for the optimal locations of video features, in Var2, because the features are fixed people detections. Thus, we compute the transition probabilities $\boldsymbol{X}$ in one-shot matching of these detections from (11), where dissimilarities $\boldsymbol{C}_{ij}$ are computed as differences of the pose descriptors. Var2 also evaluates the chains model when its inferences is not boosted by the iterative search for optimal features. **Var3** is designed to evaluate the impact of the people detector of [2] on our performance, by replacing its detections with even more noisy, lower-level features. Given a test video, we compute HOG-HOF descriptors at a dense grid of space-time interest points (STIPs) [7]. As in Var1, we here use K-means to build a dictionary of 300 codewords of HOGHOF descriptors extracted from training videos. STIPs of the test video are mapped to the dictionary of codewords, which enables estimation of our BORDs. As in Var1, we search for the optimal warping of these BORDs capturing the context of HOGHOF-based codewords. Finally, **Var4** evaluates the significance of enabling invariance to non-rigid transformations of the grid of BORDs. Var4 uses all the steps of Var1, except that we set $\alpha = \beta = 0$ in (11).

Quantitative Results: We use three types of metrics for evaluation: i) Activity classification accuracy, ii) Recall and precision of activity detection (a-detection), and iii) Recall and precision of detecting people involved in the activity (p-detection). For evaluating a-detection, we compute a ratio, ρ_a, of the intersection and union of detected and ground-truth time intervals of activity occurrences. If the activity is correctly recognized, and $\rho_a > 0.5$ then the detected interval is declared true positive (a-TP), otherwise it is false positive (a-FP). Note this also evaluates localization of start and end frames of activity occurrences. For testing p-detection, we compute a ratio, ρ_p, of the intersection and union of detected and ground-truth bounding boxes of people participating in activities. If the activity is correctly recognized, and $\rho_p > 0.5$ then the detected person is declared p-TP, otherwise they are p-FP.

Table 1 compares our activity classification accuracy with that of the state of the art approaches [6, 1, 11] on the Collective Activity Dataset. For running and dancing classes, no previous results have been reported. Table 1 also shows the average running times of our different variants. The reported running times include only the MAP inference, and do not include the time it takes to run the people detector, and compute other features and descriptors. As can be seen, Var1 outperforms [6, 1, 11] in reasonable running times. Var2 is the fastest, but also the worst of all our variants, because it does not search for the optimal features, but takes fixed people detections as activity features. Nevertheless, the inference of our chains model on "raw" features, in Var2, compares favorably against the competing methods. From Table 1, searching for the optimal features in Var1 improves performance by 3.9% relative to using "fixed" features in Var2, with reasonable increase in running time. The results for Var3 and Var4 suggest that our approach performs competitively well even with "poorer" STIP features, and that enabling invariance to locally affine transformations of features in Var1 improves performance

Class	Var1	Var2	Var3	Var4	[6]	[1]	[11]
Walk	72.2%	68.2%	68.8%	68.5%	68%	57.9%	25.5%
Cross	69.9%	65.1%	67.3%	66.4%	65%	55.4%	38.9%
Queue	96.8%	96%	96.5%	96.2%	96%	63.3%	25.5%
Wait	74.1%	70.0%	72.0%	71.1%	68%	64.6%	24.4%
Talk	99.8%	99%	99%	99%	99%	83.6%	43.0%
Run	87.6%	80%	82.7%	81.3%	N/A	N/A	N/A
Dance	70.2%	65%	67.2%	67.6%	N/A	N/A	N/A
Avg	81.5%	77.6%	79.0%	78.0%	N/A	N/A	N/A
Time	55s	42s	54s	51s	N/A	N/A	N/A

Table 1. Average activity classification accuracy, and running times of the MAP inference on the Collective Activity Dataset [1]. Our Var1 outperforms [11], [1] and [6] on all classes.

Class	Var3 Accur.	Var3 a-FP Rate	[12] Accur.	[12] a-FP Rate
S.-Hands	78.9%	6.0%	75%	8.8%
Hug	90.4%	5.5%	87.5%	7.5%
Point	66.3%	2.5%	62.5%	2.5%
Punch	63.2%	15.4%	50%	20.13%
Kick	77.5%	10.8%	75%	13.8%
Push	78.2%	10.1%	75%	12.5%
Avg	75.75%	8.3%	70.8%	10.8%

Table 2. Average activity classification accuracy, and false positive rates for a-detection on the UT Interaction Dataset [13].

by 3.5% over Var4.

Table 2 compares our activity classification accuracy, and a-FP rate with those of [12] on the UT-Interaction dataset. We here use Var3 for fair comparison, because [12] does not use any people detector. We outperform [12] in both metrics. Var3's area under ROC curve for a-detection is 0.94 which outperforms 0.91 of [12]. Also, Var3's area under ROC curve for p-detection is 0.87.

Qualitative Results: For generating qualitative results we use our default Var1. Figures 4-5 show our typical results on a few frames from the UT-Interactions and Collective Activity datasets. Figure 6 shows a failure example, where Var1 correctly detects activity *hugging*, but has p-FP for one of the actors who is too close to the true participant.

7. Conclusion

We believe that complexity of group activities requires a more synergistic interaction between high-level inference algorithms and low-level feature extractors than seen in existing work. We have specified this interaction through: i) Defining a new mid-level feature, called BORD; ii) Formulating a new generative chains model for representing group activities; and iii) Deriving a new MAP inference algorithm that guides top-down optimal extraction of BORDs from the video, and organizes them in temporal chains. This has enabled not only recognizing and localizing occurrences of group activities, but also detecting their participants. Our evaluation of benchmark UT-Human Interaction and Collective Activities datasets demonstrates that we outperform the sate of the art in activity recognition and localization, under reasonable running times.

Acknowledgement

The support of the National Science Foundation under grant NSF IIS 1018490 is gratefully acknowledged.

References

[1] W. Choi, K. Shahid, and S. Savarese. What are they doing? : Collective activity classification using spatio-temporal relationship among people. In *ICCV*, 2009. 2, 6, 7, 8

[2] P. Felzenszwalb, R. Girshick, and D. McAllester. Discriminatively trained deformable part models. In *CVPR*, 2008. 3, 5, 6

[3] A. Gupta, P. Srinivasan, J. Shi, and L. Davis. Understanding videos, constructing plots learning a visually grounded storyline model from annotated videos. In *CVPR*, 2009. 2

[4] R. Hamid, S. Maddi, A. Bobick, and I. Essa. Structure from statistics: Unsupervised activity analysis using suffix trees. In *ICCV*, pages 1–8, 2007. 2

[5] L. Karlinsky, M. Dinerstien, D. Harari, and S. Ullman. The chain model for detecting parts by their context. In *CVPR*, 2010. 2

[6] T. Lan, Y. Wang, W. Yang, and G. Mori. Beyond actions: Discriminative models for contextual group activity. In *NIPS*, 2010. 2, 6, 7

[7] I. Laptev and T. Lindeberg. Space-time interest points. In *ICCV*, pages 432–439, 2003. 6

[8] H. Li, E. Kim, X. Huang, and L. He. Object matching with a locally affine-invariant constraint. In *CVPR*, 2010. 5

[9] Z. Lin, Z. Jiang, and L. S. Davis. Recognizing actions by shape-motion prototype trees. In *ICCV*, 2009. 2

[10] J. Niebles, C. Chen, and L. Fei-Fei. Modeling temporal structure of decomposable motion segments for activity classification. In *ECCV*, 2010. 2

[11] J. C. Niebles, H. Wang, and L. Fei-Fei. Unsupervised learning of human action categories using spatial-temporal words. In *IJCV*, 2008. 6, 7

[12] M. S. Ryoo and J. K. Aggarwal. Spatio-temporal relationship match: Video structure comparison for recognition of complex human activities. In *ICCV*, 2009. 2, 6, 7

[13] M. S. Ryoo and J. K. Aggarwal. UT-Interaction Dataset, ICPR contest on Semantic Description of Human Activities (SDHA). http://cvrc.ece.utexas.edu, 2010. 2, 6, 7

[14] D. Sun, S. Roth, and M. J. Black. Secrets of optical flow estimation and their principles. In *CVPR*, 2010. 3, 5

[15] T. Xiang and S. Gong. Beyond tracking: Modelling activity and understanding behaviour. *IJCV*, 67(1):21–51, 2006. 2

[16] Q. Zhu, L. Wang, Y. Wu, and J. Shi. Contour context selection for object detection: A set-to-set contour matching approach. In *ECCV*, 2008. 2

Figure 4. Results of Var1 on example frames from the UT-Interaction Dataset. Our MAP inference correctly: i) Recognizes and detects *pushing* and *kicking* even when these activities co-occur in the video (top); ii) Identifies foreground BORDs (only a few examples shown as colored ellipses, for clarity); and iii) Detects participants of the activity (colored bounding boxes). The BORDs and people detections that are inferred to belong to the same activity are marked with the same color; the other people detections that are not associated with any activity are marked with black bounding boxes. Locations and neighborhood size of BORDs are data-driven. As can be seen, a group of people may be captured by a single BORD, and conversely one person may be covered by several BORDs.

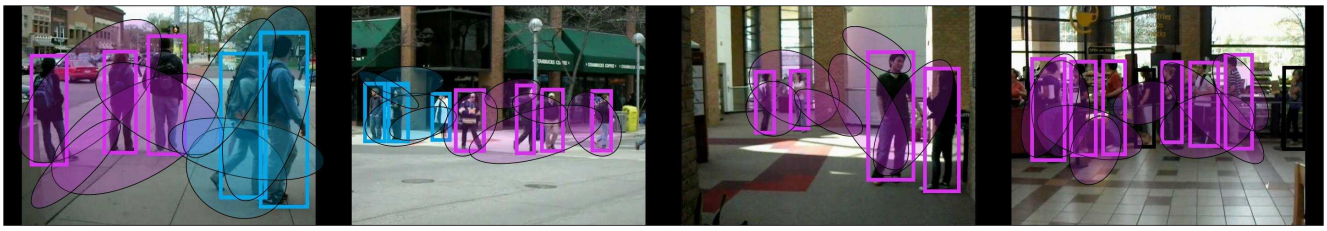


Figure 5. Results of Var1 on example frames from the Collective Activity Dataset [1]: See the caption of Fig. 4. Var1 is able to detect distinct co-occurring activities. Sometimes, Var1 infers two activities instead of one, when the true activity is spatially spread-out (e.g., walking).

Figure 6. A failure example on the UT-Interaction Dataset: See the caption of Fig. 4. Var1 correctly detects the activity *hugging*, but wrongly identifies one bounding box as the actor (red); the other actor is correctly identified (magenta); this error is because the p-FP is very close to the two truly hugging people.