HiRF: Hierarchical Random Field for Collective Activity Recognition in Videos

Mohamed Rabie Amer, Peng Lei, and Sinisa Todorovic

Oregon State University
School of Electrical Engineering and Computer Science, Corvallis, OR, USA
{amerm,leip,todorovics}@onid.orst.edu

Abstract. This paper addresses the problem of recognizing and localizing coherent activities of a group of people, called collective activities, in video. Related work has argued the benefits of capturing long-range and higher-order dependencies among video features for robust recognition. To this end, we formulate a new deep model, called Hierarchical Random Field (HiRF). HiRF models only hierarchical dependencies between model variables. This effectively amounts to modeling higher-order temporal dependencies of video features. We specify an efficient inference of HiRF that iterates in each step linear programming for estimating latent variables. Learning of HiRF parameters is specified within the max-margin framework. Our evaluation on the benchmark New Collective Activity and Collective Activity datasets, demonstrates that HiRF yields superior recognition and localization as compared to the state of the art.

Keywords: Activity recognition, hierarchical graphical models.

1 Introduction

This paper presents a new deep model for representing and recognizing collective activities in videos. A collective activity is characterized by coherent behavior of a group of people both in time and space. Coherence, for example, can be a result of all individuals in the group simultaneously performing the same action (e.g., joint jogging), or coordinated space-time interactions among people in the group (e.g., people assembling by approaching one another, or standing and periodically moving in a line). In addition to localizing time intervals where collective activities occur, we are also interested in localizing individuals that participate in the activities. While prior work has addressed recognition of collective activities, their localization has received scant attention.

Localizing collective activities in videos is challenging. It requires reasoning across a wide range of spatiotemporal scales about individual actions and trajectories, along with people interactions within various groupings. Moreover, as a group of people typically occupy a relatively large percentage of the field of view, capturing their collective activity often requires the camera to move, so recognition and localization have to be performed under camera motion.

Initial work focused on designing a heuristic descriptor of the entire video aimed at capturing coherence of a group's behavior over relatively large spatiotemporal extents [7,8,20]. Recent work specified a variety of graphical models for modeling collective activities. For example, Hierarchical Conditional Random Field (HCRF) used

D. Fleet et al. (Eds.): ECCV 2014, Part VI, LNCS 8694, pp. 572–585, 2014.

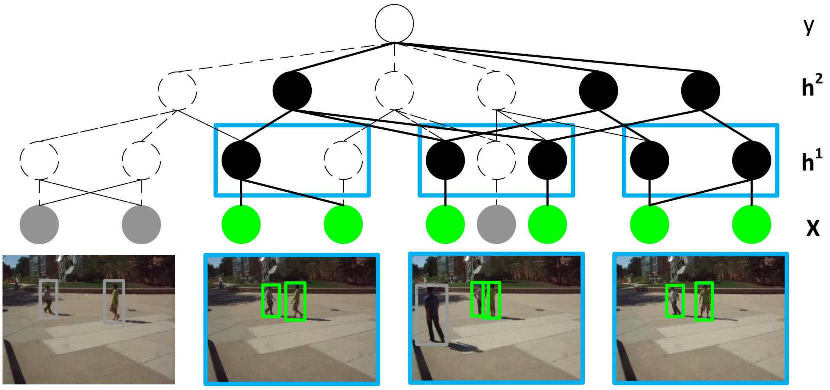

Fig. 1. Hierarchical Random Field (HiRF) for detecting and localizing collective activities. (Top) HiRF encodes the following variables: activity label y; latent temporal-connectivity variables $\boldsymbol{h}^2$; latent frame-wise connectivity variables $\boldsymbol{h}^1$; and observable video features $\boldsymbol{x}$, namely, noisy person detections. (Bottom) Our results on the New Collective Activity Dataset [6] for the activity "Talking". HiRF identifies relevant actors per frame (green), and groups person detections into temporal segments (blue) relevant for recognizing "Talking". Detections estimated as background are marked gray, and latent groupings of background video parts are marked with dashed lines.

in [17] is capable of encoding only short-term dependencies of video features, and thus may poorly discriminate between distinct activities with similar short-term but different long-range properties (e.g., group assembling vs. group walking). More advanced approaches seek to integrate people tracking with reasoning about people actions using hierarchical models [22], AND-OR graphs [2], factor graphs [6,13], or flow models [14]. However, as the complexity of these models increases, finding efficient and sufficiently accurate approximations of their intractable inference becomes more challenging.

In this paper, we advance existing work by specifying a new graphical model of collective activities, called Hierarchical Random Field (HiRF). HiRF is aimed at efficiently capturing long-range and higher-order spatiotemporal dependencies of video features, which have been shown by prior work as critical for characterizing collective activities. HiRF aggregates input features into mid-level video representations, which in turn enable robust recognition and localization. This is because the multiscale aggregation identifies groupings of foreground features, and discards features estimated as belonging to background clutter. In this way, HiRF localizes foreground in the video.

Similar to models used by recent work [2,6,13,14], HiRF also seeks to capture long-range temporal dependencies of visual cues. However, the key difference is that HiRF avoids the standard strategy to establish lateral temporal connections between model variables. Instead, HiRF encodes temporal dependencies among video features through strictly hierarchical ("vertical") connections via two hierarchical levels of latent variables, as illustrated in Fig. 1. At the leaf level, HiRF is grounded onto video features, extracted by applying a person detector in each frame. The next level of HiRF consists of latent variables, which serve to spatially group foreground video features into subactivities relevant for recognition. Since this feature grouping is latent, the identified latent subactivities may not have any semantic meaning. The next higher level of HiRF also consists of latent variables. They are aimed at identifying long-range temporal dependencies among latent subactivities. The result of this long-range feature grouping is used for inferring the activity class at the root node of HiRF.

We specify an efficient bottom-up/top-down inference of HiRF. In particular, our inference is iterative, where each step solves a linear program (LP) for estimating one set of latent variables at a time. This is more efficient than the common quadratic programming used for inference of existing graphical models with lateral connections. Learning of HiRF parameters is specified within the max-margin framework.

HiRF does not require explicit encoding of higher-order potentials as the factor graphs of [6,13]. Yet, our evaluation on the benchmark New Collective Activity dataset [6] demonstrates that HiRF yields superior recognition accuracy by 4.3% relative to the factor graph of [6], and outperforms the approaches of [13] and [6] by 20% and 12% on the Collective Activity dataset [7]. To the best of our knowledge, we present the first evaluation of localizing people that participate in collective activities on the New Collective Activity dataset.

In the following, Sec. 2 reviews related work; Sec. 3 formulates HiRF; Sec. 4 specifies inference; Sec. 5 explains learning; and Sec. 6 presents our results.

2 Related Work

There is a large volume of literature on capturing spatiotemporal dependencies among visual cues for activity recognition [1,25,5]. Representative models include Dynamic Bayesian Networks [28,27], Hidden Conditional Random Fields (HCRFs) [24,17], hierarchical graphical models[17,16,15,22], AND-OR graphs [21,2], and Logic Networks [19,4]. In these models, higher-order dependencies are typically captured by latent variables. This generally leads to NP-hard inference. Intractable inference is usually addressed in a heuristic manner by, for example, restricting the connectivity of variables in the model to a tree [24]. Our HiRF is not restricted to have a tree structure, while its strictly hierarchical connectivity enables efficient inference and learning.

HiRF is also related to Shape Boltzmann Machines [10] used for object segmentation in images. They locally constrain the allowed extent of dependencies between their model variables so as to respect image segmentation. We also locally constrain the connectivity between our latent variables over certain temporal windows along the video, in order to identify latent subactivities relevant for recognition of the collective activity. HiRF is also related to Conditional Random Fields of [18,12] which encode higher-order potentials of image features using Restricted Boltzmann Machines. Similarly, HiRF uses a hierarchy of latent variables to identify latent groupings of video features, which amounts to encoding their higher-order dependencies in time and space.

3 The Model

This section, first, introduces some notation and definitions which will be used for specifying our HiRF model, then reviews closely related models used for representing collective activities, and finally defines HiRF.

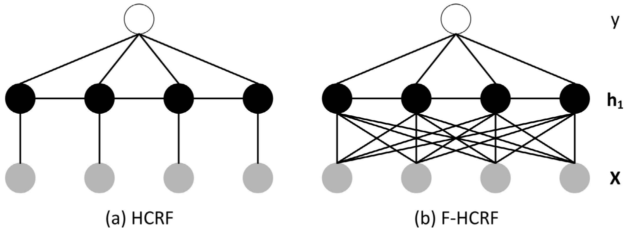

Fig. 2. Closely related existing models: (a) HCRF [23,24] contains a hidden layer $\boldsymbol{h}^1$ that is temporally connected with lateral edges, while every hidden node h_i^1 is connected to only one observable node $\boldsymbol{x}_i$. (b) F-HCRF [17] extends HCRF such that every hidden node $\boldsymbol{h}_i^1$ is connected to many observable nodes $\boldsymbol{x}$, capturing long-range temporal dependencies.

HiRF is a graphical model defined over a graph, $\mathcal{G} = (\mathcal{V}, \mathcal{E})$. Nodes $\mathcal{V}$ represent: observable video features $\boldsymbol{x} = \{\boldsymbol{x}_i : \boldsymbol{x}_i \in \mathbb{R}^d\}$, integer latent variables $\boldsymbol{h} = \{h_i : h_i \in \mathcal{H}\}$, and a class label $y \in \mathcal{Y}$. Edges $\mathcal{E}$ encode dependencies between the nodes in $\mathcal{V}$. HiRF is characterized by a posterior distribution, $P(y, \boldsymbol{h}|\boldsymbol{x}; \boldsymbol{w})$, where $\boldsymbol{w}$ are model parameters. The posterior distribution has the Gibbs form: $P(\cdot) = \exp(-E(\cdot))/Z$, where $E(\cdot)$ is the energy, and Z is the partition function.

In the following, we explain our novelty by defining $E(\cdot)$ for a progression of closely related models, shown in Fig. 2.

3.1 Review of Hidden Conditional Random Field

HCRF [23,24] extends the expressiveness of standard CRF by introducing a *single* layer of hidden variables $\boldsymbol{h}^1$ between $\boldsymbol{x}$ and y, where each h_i is connected to a single $\boldsymbol{x}_i$. Each h_i may take a value from a set of integers, called "topics". In this way, video features can be grouped into latent "topics", and thus capture their dependencies. The energy of HCRF is defined as

$$E_{\text{HCRF}}(y, \boldsymbol{h}|\boldsymbol{x}) = -\Big[\sum_i \boldsymbol{w}^0 \cdot \boldsymbol{\phi}^0(y, h_i^1) + \sum_i \boldsymbol{w}^1 \cdot \boldsymbol{\phi}^1(h_i^1, \boldsymbol{x}_i) + \sum_{i,j} \boldsymbol{w}^2 \cdot \boldsymbol{\phi}^2(y, h_i^1, h_j^1)\Big], \tag{1}$$

where "$\cdot$" denotes scalar multiplication of two vectors; $\boldsymbol{\phi}(y, \boldsymbol{h}, \boldsymbol{x}) = (\boldsymbol{\phi}^0, \boldsymbol{\phi}^1, \boldsymbol{\phi}^2)$ are feature vectors with all elements equal to zero except for a single segment of non-zero elements indexed by the states of latent variables; and $\boldsymbol{w} = (\boldsymbol{w}^0, \boldsymbol{w}^1, \boldsymbol{w}^2)$ are model parameters.

The one-to-one connectivity between $\boldsymbol{h}^1$ and $\boldsymbol{x}$ in HCRF, however, poorly captures long-range dependancies. To overcome this issue, HCRF has been extended to the feature-level HCRF (F-HCRF) [17] by establishing a full connectivity between the hidden nodes and observable nodes. The energy of F-HCRF is defined as

$$E_{\text{F-HCRF}}(y, \boldsymbol{h}|\boldsymbol{x}) = -\Big[\sum_i \boldsymbol{w}^0 \cdot \boldsymbol{\phi}^0(y, h_i^1) + \sum_{i,j} \boldsymbol{w}^1 \cdot \boldsymbol{\phi}^1(h_i^1, \boldsymbol{x}_j) + \sum_{i,j} \boldsymbol{w}^2 \cdot \boldsymbol{\phi}^2(y, h_i^1, h_j^1)\Big]. \tag{2}$$

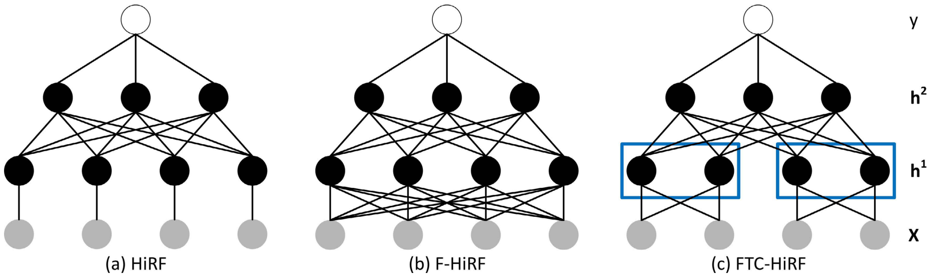

Fig. 3. Variants of HiRF. (a) HiRF introduces an additional hidden layer $\boldsymbol{h}^2$ to HCRF and removes all lateral connections between the hidden nodes; (b) F-HiRF extends HiRF by establishing the full connectivity between nodes of the hidden layer $\boldsymbol{h}^1$ and the leaf nodes $\boldsymbol{x}$; and (c) FTC-HiRF extends F-HiRF by introducing temporal constraints on the connectivity of hidden nodes $\boldsymbol{h}^1$ to the leaf nodes for reasoning about subactivities of the collective activity.

3.2 Formulation of HiRF

In this section, we formulate HiRF by: (1) Adding another layer of hidden variables to HCRF and F-HCRF reviewed in Sec. 3.1; (2) Removing the lateral temporal connections between all hidden variables; and (3) Enforcing local constraints on temporal connections between the hidden variables. The extensions (1) and (2) are aimed at more efficiently capturing higher-order and long-range temporal dependencies of video features. The extension (3) is aimed at automatically capturing domain knowledge that complex collective activities are typically temporally structured into subactivities, which bound the extent of long-range temporal dependencies of video features. Since these subactivities may not have a particular semantic meaning, but may be relevant for recognition, we use (3) to model subactivities as latent groupings of the first layer of hidden variables $\boldsymbol{h}^1$, as further explained below.

We first define two variants of our model — namely, HiRF which extends HCRF, and F-HiRF which extends F-HCRF by introducing a new layer of hidden variables, $\boldsymbol{h}^2$, between $\boldsymbol{h}^1$ and y, as illustrated in Fig. 3. Their energy functions are defined as

$$\begin{aligned} E_{\text{HiRF}}(y,\boldsymbol{h}|\boldsymbol{x}) &= -\Big[\sum_i \boldsymbol{w}^0\cdot\boldsymbol{\phi}^0(y,h_i^2) + \sum_i \boldsymbol{w}^1\cdot\boldsymbol{\phi}^1(h_i^1,\boldsymbol{x}_i) + \sum_{i,j} \boldsymbol{w}^2\cdot\boldsymbol{\phi}^2(h_i^1,h_j^2)\Big], \\ E_{\text{F-HiRF}}(y,\boldsymbol{h}|\boldsymbol{x}) &= -\Big[\sum_i \boldsymbol{w}^0\cdot\boldsymbol{\phi}^0(y,h_i^2) + \sum_{i,j} \boldsymbol{w}^1\cdot\boldsymbol{\phi}^1(h_i^1,\boldsymbol{x}_j) + \sum_{i,j} \boldsymbol{w}^2\cdot\boldsymbol{\phi}^2(h_i^1,h_j^2)\Big], \end{aligned} \tag{3}$$

where we use the same notation for feature vectors, $\boldsymbol{\phi}(y,\boldsymbol{h},\boldsymbol{x}) = (\boldsymbol{\phi}^0,\boldsymbol{\phi}^1,\boldsymbol{\phi}^2)$, and model parameters, $\boldsymbol{w} = (\boldsymbol{w}^0,\boldsymbol{w}^1,\boldsymbol{w}^2)$, as defined for (1) and (2).

In (3), y encodes the activity class label, and latent integer variables $\boldsymbol{h}^2$ and $\boldsymbol{h}^2$ are aimed at identifying and grouping foreground video features relevant for recognizing the activity. Specifically, every node h_i^2 may take binary values, $h_i^2 \in \{0,1\}$, indicating figure-ground assignment of video features. Every node h_i^1 may take integer values, $h_i^1 \in \mathcal{H} = \{0,\ldots,|\mathcal{H}|\}$, indicating latent groupings of video features into "topics".

As shown in Fig. 3, both HiRF and F-HiRF have only hierarchical edges between variables. The newly introduced hidden layer $\boldsymbol{h}^2$ serves to replace the lateral connections of HCRF and F-HCRF. At the same time, $\boldsymbol{h}^2$ enables long-range temporal connectivity between the hidden nodes without introducing higher order potentials. From (3), the key difference between HiRF and F-HiRF is that we allow only one-to-one connectivity between the hidden layer $\boldsymbol{h}^1$ and observable nodes $\boldsymbol{x}$ in HiRF, whereas this connectivity is extended to be full in F-HiRF. In this way, our F-HiRF is expected to have the same advantages of F-HCRF over HCRF, mentioned in Sec. 3.1.

We next extend F-HiRF to Temporally Constrained F-HiRF, called FTC-HiRF. FTC-HiRF enforces local constraints on temporal dependencies of the hidden variables. Similar to Shape Boltzmann Machines [10], we partition the first hidden layer $\boldsymbol{h}^1$ such that every partition has access only to a particular temporal segment of the video. While the partitions of $\boldsymbol{h}^1$ cannot directly connect to all video segments, their long-range dependencies are captured through connections to the second hidden layer $\boldsymbol{h}^2$.

Specifically, in FTC-HiRF, the first hidden layer $\boldsymbol{h}^1$ is divided into subsets, $\boldsymbol{h}^1 = \{\boldsymbol{h}^1_t : t = 1, \ldots, T\}$, where each $\boldsymbol{h}^1_t$ can be connected only to the corresponding temporal window of video features $\boldsymbol{x}_t$. The energy of FTC-HiRF is defined as

$$\begin{aligned} E_{\text{FTC-HiRF}}(y, \boldsymbol{h}|\boldsymbol{x}) = -\Big[\sum_i \boldsymbol{w}^0 \cdot \boldsymbol{\phi}^0(y, h^2_i) + \sum_t \sum_{(i,j)\in t} \boldsymbol{w}^1 \cdot \boldsymbol{\phi}^1(h^1_{it}, \boldsymbol{x}_{jt}) \\ + \sum_{i,j} \boldsymbol{w}^2 \cdot \boldsymbol{\phi}^1(h^2_i, h^1_j) \Big], \end{aligned} \tag{4}$$

where the third term includes all the hidden variables $\boldsymbol{h}^1$ from all temporal partitions.

3.3 Definitions of the Potential Functions

The section defines the three types of potential functions of HiRF, specified in (3).

The potential $[\boldsymbol{w}^0 \cdot \boldsymbol{\phi}^0(y, h^2_i)]$ models compatibility between the class label $y \in \mathcal{Y} = \{a : a = 1, 2, \ldots\}$, and the particular figure-ground indicator $h^2_i \in \{b : b = 0, 1\}$. The parameters $\boldsymbol{w}^0 = [w^0_{ab}]$ are indexed by the activity class labels of y, and binary states of h^2_i. We define this potential as

$$\boldsymbol{w}^0 \cdot \boldsymbol{\phi}^0(y, h^2_i) = \textstyle\sum_{a\in\mathcal{Y}} \sum_{b\in\{0,1\}} w^0_{ab} \mathbb{1}(y = a)\mathbb{1}(h^2_i = b). \tag{5}$$

The potential $[\boldsymbol{w}^1 \cdot \boldsymbol{\phi}^1(h^1_i, \boldsymbol{x}_j)]$ models compatibility between the "topic" assigned to $h^1_i \in \mathcal{H} = \{c : c = 1, \ldots, |\mathcal{H}|\}$, and the d-dimensional video feature vector $\boldsymbol{x}_j$, when nodes $\boldsymbol{x}_j$ and h^1_i are connected in the graphical model. The parameters $\boldsymbol{w}^1 = [\boldsymbol{w}^1_c]$ are indexed by the "topics" of h^1_i. We define this potential as

$$\boldsymbol{w}^1 \cdot \boldsymbol{\phi}^1(h^1_i, \boldsymbol{x}_j) = \textstyle\sum_{c\in\mathcal{H}} \boldsymbol{w}^1_c \cdot \boldsymbol{x}_j \mathbb{1}(h^1_i = c). \tag{6}$$

The potential $[\boldsymbol{w}^2 \cdot \boldsymbol{\phi}^2(h^1_i, h^2_j)]$ models compatibility between the figure-ground assignment of $h^2_j \in \{b : b = 0, 1\}$, and the "topic" assigned to $h^1_i \in \mathcal{H} = \{c : c = 1, \ldots, |\mathcal{H}|\}$ when nodes h^2_j and h^1_i are connected in the graphical model. The parameters $\boldsymbol{w}^2 = [w^2_{bc}]$ are indexed by the binary states of h^2_j and the "topics" of h^1_i. We define this potential as

$$\boldsymbol{w}^2 \cdot \boldsymbol{\phi}^2(h^1_i, h^2_j) = \textstyle\sum_{b\in\{0,1\}} \sum_{c\in\mathcal{H}} w^2_{bc} \mathbb{1}(h^2_j = b)\mathbb{1}(h^1_i = c). \tag{7}$$

4 Bottom-up/Top-down Inference Using Linear Programming

Given a video $\boldsymbol{x}$ and model parameters $\boldsymbol{w}$, the goal of inference is to predict y and $\boldsymbol{h}$ as

$$\{\hat{y}, \hat{\boldsymbol{h}}\} = \arg\max_{y,\boldsymbol{h}} \ \boldsymbol{w}\cdot\boldsymbol{\phi}(y, \boldsymbol{h}, \boldsymbol{x}). \tag{8}$$

We solve this inference problem by iterating the following bottom-up and top-down computational steps.

Bottom-up pass. In the bottom-up pass of iteration τ, we first use the observable variables $\boldsymbol{x}$ and $\hat{\boldsymbol{h}}^2(\tau-1)$ to estimate $\hat{\boldsymbol{h}}^1(\tau)$, then, from $\hat{\boldsymbol{h}}^1(\tau)$ and $\hat{y}(\tau-1)$ we compute $\hat{\boldsymbol{h}}^2(\tau)$, and finally, we use $\hat{\boldsymbol{h}}^2(\tau)$ to estimate $\hat{y}(\tau)$. To this end, we reformulate the potentials, given by (5)–(7), as linear functions of the corresponding unknown variables, using auxiliary binary vectors $\boldsymbol{z}_i^1 \in \{0,1\}^{|\mathcal{H}|}$, $\boldsymbol{z}_i^2 \in \{0,1\}^2$, and $\boldsymbol{z}^y \in \{0,1\}^{|\mathcal{Y}|}$, where $z_{i,c}^1 = 1$ if $h_i^1 = c \in \mathcal{H}$, and $z_{i,b}^2 = 1$ if $h_i^2 = b \in \{0,1\}$, and $z_a^y = 1$ if $y = a \in \mathcal{Y}$. Thus, from (3), (6), and (7), we derive the following LPs for each node i of our model:

$$\boldsymbol{z}_i^1(\tau) = \arg\max_{\boldsymbol{z}_i^1} \ \boldsymbol{z}_i^1 \cdot \Big[\sum_j \boldsymbol{w}^1\cdot\boldsymbol{x}_j + \sum_j \boldsymbol{w}^2 \cdot \boldsymbol{z}_j^2(\tau-1)\Big], \text{ s.t.} \sum_{c\in\mathcal{H}} z_{i,c}^1 = 1, \tag{9}$$

$$\boldsymbol{z}_i^2(\tau) = \arg\max_{\boldsymbol{z}_i^2} \ \boldsymbol{z}_i^2 \cdot \Big[\sum_j \boldsymbol{w}^2 \cdot \boldsymbol{z}_j^{1(\tau)} + \boldsymbol{w}^0 \cdot \boldsymbol{z}^y(\tau-1)\Big], \text{ s.t.} \sum_{b\in\{0,1\}} z_{i,b}^2 = 1, \tag{10}$$

$$\boldsymbol{z}^y(\tau) = \arg\max_{\boldsymbol{z}^y} \ \boldsymbol{z}^y \cdot \Big[\sum_i \boldsymbol{w}^0 \cdot \boldsymbol{z}_i^2(\tau-1)\Big], \text{ s.t.} \sum_{a\in\mathcal{Y}} z_a^y = 1 \tag{11}$$

Top-down pass. In the top-down pass, we solve the above LPs in the reverse order.

In our experiments, we observed convergence for $\tau_{\max} = 10$. After $\tau_{\max}$ iterations, the LP solutions $\boldsymbol{z}_i^1(\tau_{\max})$, $\boldsymbol{z}_i^2(\tau_{\max})$, and $\boldsymbol{z}^y(\tau_{\max})$ uniquely identify $\hat{\boldsymbol{h}}$ and $\hat{y}$. Due to the LP formulations in (9)–(11), our inference is more efficient than the quadratic-optimization based inference algorithms of recent approaches presented in [17,6,13].

5 Max-Margin Learning

We use the max-margin framework for learning HiRF parameters, as was done in [24] for learning HCRF parameters. In particular, we use the latent-SVM to learn $\boldsymbol{w}$ on labeled training examples $\mathcal{D} = \{(\boldsymbol{x}^{(l)}, y^{(l)}) : l = 1, 2, \dots\}$, by solving the following optimization problem:

$$\min_{\boldsymbol{w}} \underbrace{\Big[\frac{C}{2}\|\boldsymbol{w}\|^2 + \sum_l \boldsymbol{w}\cdot\boldsymbol{\phi}(\hat{y}^{(l)}, \hat{\boldsymbol{h}}^{(l)}, \boldsymbol{x}^{(l)}) + \Delta(\hat{y}, y^{(l)})\Big]}_{f(\boldsymbol{w})} - \underbrace{\Big[\sum_l \boldsymbol{w}\cdot\boldsymbol{\phi}(y^{(l)}, \boldsymbol{h}^{*(l)}, \boldsymbol{x}^{(l)})\Big]}_{g(\boldsymbol{w})} \tag{12}$$

where $\Delta(\hat{y}, y^{(l)})$ is the 0-1 loss, $\boldsymbol{w} \cdot \boldsymbol{\phi}(\hat{y}^{(l)}, \hat{\boldsymbol{h}}^{(l)}, \boldsymbol{x}^{(l)}) = \max_{y,\boldsymbol{h}} \boldsymbol{w} \cdot \boldsymbol{\phi}(y, \boldsymbol{h}, \boldsymbol{x}^{(l)})$, and $\boldsymbol{w} \cdot \boldsymbol{\phi}(y^{(l)}, \boldsymbol{h}^{*(l)}, \boldsymbol{x}^{(l)}) = \max_{\boldsymbol{h}} \boldsymbol{w} \cdot \boldsymbol{\phi}(y^{(l)}, \boldsymbol{h}, \boldsymbol{x}^{(l)})$. The presence of hidden variables $\boldsymbol{h}$ in (12) make the overall optimization problem non-convex. The problem in (12) can be expressed as a difference of two convex terms $f(\boldsymbol{w})$ and $g(\boldsymbol{w})$, and thus can be solved using the CCCP algorithm [26]. Our learning iterates two steps: (i) Given $\boldsymbol{w}$, each $\hat{y}^{(l)}$, $\hat{\boldsymbol{h}}^{(l)}$, and $\boldsymbol{h}^{*(l)}$ can be efficiently estimated using our bottom-up/top-down inference explained in Sec. 4; (ii) Given the 0-1 loss $\Delta(\hat{y}, y^{(l)})$ and all features $\boldsymbol{\phi}(\hat{y}^{(l)}, \hat{\boldsymbol{h}}^{(l)}, \boldsymbol{x}^{(l)})$ and $\boldsymbol{\phi}(y^{(l)}, \boldsymbol{h}^{*(l)}, \boldsymbol{x}^{(l)})$, $\boldsymbol{w}$ can be estimated by the CCCP algorithm.

6 Results

This section specifies our evaluation datasets, implementation details, evaluation metrics, baselines and comparisons with the state of the art.

We evaluate our approach on the Collective Activity Dataset (CAD) [7], and New Collective Activity Dataset (New-CAD) [6]. CAD consists of 44 videos showing 5 collective activities: crossing, waiting, queuing, walking, and talking. For training and testing, we use the standard split of $3/4$ and $1/4$ of the videos from each class. In every 10th frame, CAD provides annotations of bounding boxes around people performing the activity, their pose, and activity class. We follow the same experimental setup as described in [17]. New-CAD consists of 32 videos showing 6 collective activities – namely, gathering, talking, dismissal, walking together, chasing, queuing – and 9 interactions – specifically, approaching, walking-in-opposite-direction, facing-each-other, standing-in-a-row, walking-side-by-side, walking-one-after-the-other, running-side-by-side, running-one-after-the-other, and no-interaction – and 3 individual actions called walking, standing still, and running. The annotations include 8 poses. As in [6], we divide New-CAD into 3 subsets, and run 3-fold training and testing.

Implementation Details. We first run the person detector of [11] that uses HOG features [9]. The detector is learned to yield high recall. On CAD and New-CAD, we get the average false positive rates of 15.6% and 18.1%, respectively. Each person detection corresponds to one leaf node in HiRF, and is assigned an action descriptor, $\boldsymbol{x}_i$, similar to the descriptor used in [17]. We compute the action descriptor by concatenating a person descriptor that captures the person's pose and action, and another contextual descriptor that captures the poses and actions of nearby persons. The person descriptor is a $|\mathcal{Y}|\cdot 8$-dimensional vector that consists of confidences of two classifiers which use HOG features of the person's detection bounding box – namely, confidences of SVM over $|\mathcal{Y}|$ action classes, and confidences of the 8-way pose detector presented in [7]. The contextual descriptor is a $|\mathcal{Y}|\cdot 8$-dimensional vector that computes the maximum confidences over all person descriptors associated with the neighboring person detections. We use 10 nodes at the $\boldsymbol{h}^1$ level, and 20 nodes at the $\boldsymbol{h}^2$ level, empirically estimated as providing an optimal trade-off between accuracy and model complexity. We establish the fully connectivity between all nodes of levels $\boldsymbol{h}^1$ and $\boldsymbol{h}^2$. To enforce temporal constraints in FTC-HiRF, we split the video into T time intervals, and allow connections between $\boldsymbol{h}^1_{it}$ nodes and leaf nodes $\boldsymbol{x}_{jt}$ only within their respective intervals $t = 1, \ldots, T$. The optimal $T = 10$ is empirically evaluated. Training takes about 6 hours, on a 3.40GHz PC with 8GB RAM.

Baselines. Our baselines include HCRF and F-HCRF specified in Sec. 3.1 and illustrated in Fig. 2. The comparison between our HiRF and HCRF evaluates the effect of replacing the temporal lateral connections in the HCRF with strictly hierarchical connections in HiRF. F-HiRF and FTC-HiRF are variants of HiRF. F-HiRF fully connects all nodes of the hidden layer $\boldsymbol{h}^1$ with all observable variables $\boldsymbol{x}$. FTC-HiRF splits the observable nodes into T disjoint sets $\boldsymbol{x}_t$, $t = 1, \ldots, T$, corresponding to T time intervals in the video, and connects each node of the hidden layer $\boldsymbol{h}^1$ only with the corresponding set of observable nodes $\boldsymbol{x}_t$. The comparison between F-HiRF and FTC-HiRF evaluates the effect of temporally constraining the video domain modeled by each node of the hidden layer $\boldsymbol{h}^1$. In the following, we will assume that our default model is FTC-HiRF.

Comparison. We compare FTC-HiRF with the state-of-the-art temporal approaches of [2,6,13]. These approaches apply a people tracker, and thus additionally use tracking information for inferring their Factor Graphs (FG) [6,13] and spatiotemporal And-Or graphs [2]. FTC-HiRF *does not* use any tracking information. For a fair comparison with non-temporal approaches of [3,17,7] that conduct per-frame reasoning about activities using SVM [7], F-HCRF [17], and spatial And-Or graph [3], we define a special variant of our model, called HiRFnt. HiRFnt has the same formulation as HiRF except that it uses observables (i.e., person detections) only from a single frame. Thus, inference of HiRFnt is performed for every individual video frame. Also, note that for evaluation on New-CAD all prior work uses a higher level of supervision for training their hidden variables – namely, the available interaction labels – whereas, we do not use these labels in our training.

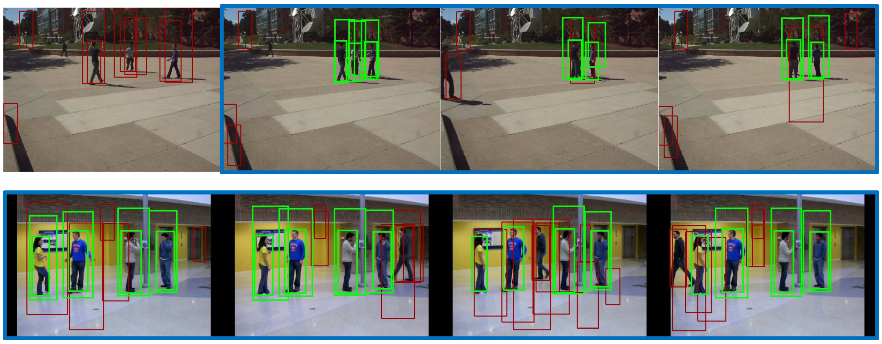

Fig. 4. Person detections using the detector of [11] set to give high recall, and our results using FTC-HiRF on: (top) the New-CAD dataset, and (bottom) the CAD dataset, for the activity "talking". The estimated foreground and background are marked green and red, respectively. The blue frames indicate our localization of the activity's temporal extent. Note that more than one detection falling on the same person may be estimated as foreground.

Evaluation Metrics. We evaluate classification accuracy in (%), and precision and recall of localizing foreground video parts. A true positive is declared if the intersection of the estimated foreground and ground truth is larger than 50% of their union.

Experiments. We first evaluate individual components of our model by comparing the performance of our model variants. Then, we compare FTC-HiRF against the state of the art. Fig. 4 shows our example results on CAD and New-CAD. As can be seen, in these examples, FTC-HiRF successfully detected the activity and localized foreground associated with the recognized activity. In general, FTC-HiRF successfully estimates foreground-background separation, as long as the background activities do not have similar spatiotemporal layout and people poses as the foreground activity.

Fig. 5 shows the sensitivity of FTC-HiRF to the input parameters on the CAD and New-CAD datasets. As can be seen, our model is relatively insensitive to a range of parameter values, but starts overfitting for higher values.

Tab. 1 and Tab. 2 show the comparison of FTC-HiRF and other variants of our approach with the baselines on CAD and New-CAD. As can be seen, in comparison with HCRF, HiRF reduces the running time of inference, and achieves a higher average classification accuracy. These results demonstrate the advantages of using strictly hierarchical connections in our approach. F-HiRF slightly improves the results of HiRF, but has a longer running time. The longer running time is due to the full connectivity between the $\boldsymbol{h}^1$ level and the leaf level. Finally, FTC-HiRF improves the classification accuracy of F-HiRF, and at the same time runs faster than F-HiRF. The training and test times of FTC-HiRF are smaller than those of F-HiRF, since convergence in inference is achieved faster due to a more constrained graph connectivity in FTC-HiRF.

Tab. 3, and Tab. 4 show the comparison of FTC-HiRF with the state of the art on CAD and New-CAD. As can be seen, we are able to achieve higher classification accuracy, under faster running times. Our non-temporal variant HiRFnt outperforms HCRF [17] by 6.2% and spatiotemporal And-Or graph [2] by 3% on both datasets.

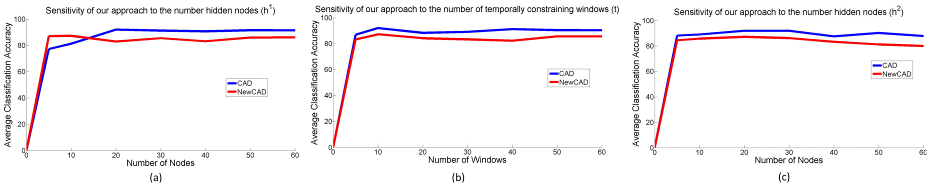

Fig. 5. Sensitivity of FTC-HiRF to the input parameters on the CAD (blue) and New-CAD (red) datasets. Average classification accuracy when using different numbers of: (a) Nodes at the $\boldsymbol{h}^1$ level; (b) Temporal intervals T; (c) Nodes at the $\boldsymbol{h}^2$ level.

Table 1. CAD: Average classification accuracy in [%], and run time in seconds

Class	HCRF [24]	F-HCRF [17]	FTC-HCRF [17]	HiRF	F-HiRF	FTC-HiRF
Walk	83.3	83.9	87.6	84.1	86.2	89.7
Cross	71.2	71.7	78.7	76.8	78.1	86.5
Queue	79.2	80.5	82.2	81.1	83.4	98.2
Wait	71.8	73.6	75.8	74.3	75.1	85.9
Talk	99.1	99.3	99.4	99.3	99.4	99.6
Avg	80.9	81.8	84.7	83.1	84.4	92.0
Time	400	440	300	100	150	120

Table 2. New-CAD: Average accuracy in [%], and run time in seconds

Class	HCRF [24]	F-HCRF [17]	FTC-HCRF [17]	HiRF	F-HiRF	FTC-HiRF
Gathering	45.3	47.1	52.3	49.2	52.1	54.9
Talking	84.1	84.5	83.9	84.7	86.2	89.3
Dismissal	78.2	79.6	80.6	78.1	82.8	87.6
Walking	89.3	89.1	89.0	89.4	91.2	94.3
Chasing	93.5	93.5	93.7	94.1	96.4	98.2
Queuing	93.0	93.1	94.3	93.8	95.6	99.2
Avg	80.6	81.1	82.3	81.5	84.0	87.2
Time	400	440	300	100	150	120

Table 3. Average classification accuracy in [%], and run time in seconds on CAD. FTC-HiRF is used to compare against temporal approaches of [2,6,13], while HiRFnt is used to compare against non-temporal approaches of [3,17,7].

Class	FTC-HiRF	ST-AOG [2]	FG [6]	FG [13]	HiRFnt	AOG [3]	HCRF [17]	SVM [7]
Walk	89.7	83.4	65.1	61.5	77.3	74.7	80	58.6
Cross	86.5	81.1	61.3	67.2	81.2	77.2	68	59.4
Queue	98.2	97.5	95.4	81.1	96.2	95.4	76	80.6
Wait	85.9	83.9	82.9	56.8	78.4	78.3	69	81.9
Talk	99.6	98.8	94.9	93.3	99.6	98.4	99	86.0
Avg	92.0	88.9	80.0	72.0	86.6	84.8	78.4	72.5
Time	120	180	N/A	N/A	80	160	N/A	N/A

Table 4. Average classification accuracy in [%], and run time in seconds on New-CAD. FTC-HiRF is used to compare against temporal approaches of [2,6], while HiRFnt is used to compare against non-temporal approaches of [3,7].

Class	FTC-HiRF	ST-AOG[2]	FG [6]	HiRFnt	AOG[3]	SVM [7]
Gathering	54.9	48.9	43.5	51.2	44.2	50.0
Talking	89.3	86.5	82.2	83.1	76.9	72.2
Dismissal	87.6	84.1	77.0	79.2	50.1	49.2
Walking	94.3	92.5	87.4	88.1	84.3	83.2
Chasing	98.2	96.5	91.9	92.6	91.2	95.2
Queuing	99.2	97.2	93.4	92.1	92.2	95.9
Avg	87.3	84.2	83.0	81.0	74.8	77.4
Time	120	180	N/A	80	160	N/A

Tab. 5 shows our precision and false positive rates for localizing the activities on CAD. As can be seen, HiRFnt successfully localizes foreground, and outperforms the spatial And-Or graph of [3] by 3.7% in precision.

Tab. 6 shows our precision and false positive rates for localizing the activities on New-CAD. To the best of our knowledge, we are the first to report localization results on New-CAD.

Table 5. Average precision and false positive rates in (%) on CAD

Class	FTC-HiRF Precision	FTC-HiRF FP	HiRFnt Precision	HiRFnt FP	S-AOG [3] Precision	S-AOG [3] FP
Walk	70.0	7.6	68.1	8.0	65.3	8.2
Cross	78.3	8.1	75.0	8.4	69.6	8.7
Queue	79.1	5.0	78.7	5.1	76.2	5.2
Wait	76.7	7.0	74.1	7.4	68.3	7.7
Talk	87.9	5.7	84.4	6.0	82.1	6.2
Avg	78.4	6.7	76.0	7.0	72.3	7.2

Table 6. Average precision and false positive rates in (%) on New-CAD

Class	FTC-HiRF Precision	FTC-HiRF FP	HiRFnt Precision	HiRFnt FP
Gathering	77.8	15.6	77.5	18.5
Talking	85.1	6.4	80.9	6.5
Dismissal	72.2	11.1	68.2	14.1
Walking	74.5	3.1	72.2	3.5
Chasing	68.0	7.7	65.2	10.5
Queuing	92.7	6.5	88.1	7.2
Avg	77.7	8.4	75.2	10.0

7 Conclusion

We have presented a new deep model, called Hierarchical Random Field (HiRF), for modeling, recognizing and localizing collective activities in videos. HiRF extends recent work that models activities with HCRF by: 1) Adding another layer of hidden variables to HCRF, 2) Removing the lateral temporal connections between all hidden variables, and 3) Enforcing local constraints on temporal connections between the hidden variables for capturing latent subactivities. We have also specified new inference of HiRF. Our inference iterates bottom-up/top-down computational steps until convergence, where each step efficiently estimates the latent variables using a linear program. Efficiency comes from our formulation of the potentials of HiRF as linear functions in each set of the hidden variables, given current estimates of other variables. This advances prior work which requires more complex quadratic programing in inference. Our empirical evaluation on the benchmark Collective Activity Dataset [7] and New Collective Activity Dataset [6] demonstrates the advantages of using strictly hierarchical connections in our approach. Our model is relatively insensitive to a range of input parameter values, but starts overfitting for higher values. In comparison with HCRF, HiRF reduces the running time of inference, and achieves a higher average classification accuracy. Also, HiRF outperforms the state-of-the-art approaches, including Factor Graphs [6,13] and spatiotemporal And-Or graphs [2], in terms of classification accuracy, precision, and recall.

Acknowledgment. This work was supported in part by grant NSF RI 1302700 and DARPA MSEE FA 8650-11-1-7149.

References

1. Aggarwal, J., Ryoo, M.: Human activity analysis: A review. ACM Comput. Surv. 43, 16:1–16:43 (2011)
2. Amer, M., Todorovic, S., Fern, A., Zhu, S.: Monte carlo tree search for scheduling activity recognition. In: ICCV (2013)
3. Amer, M.R., Xie, D., Zhao, M., Todorovic, S., Zhu, S.-C.: Cost-sensitive top-down/Bottom-up inference for multiscale activity recognition. In: Fitzgibbon, A., Lazebnik, S., Perona, P., Sato, Y., Schmid, C. (eds.) ECCV 2012, Part IV. LNCS, vol. 7575, pp. 187–200. Springer, Heidelberg (2012)
4. Brendel, W., Fern, A., Todorovic, S.: Probabilistic event logic for interval-based event recognition. In: CVPR (2011)
5. Chaquet, J.M., Carmona, E.J., Fernández-Caballero, A.: A survey of video datasets for human action and activity recognition. CVIU 117(6), 633–659 (2013)
6. Choi, W., Savarese, S.: A unified framework for multi-target tracking and collective activity recognition. In: Fitzgibbon, A., Lazebnik, S., Perona, P., Sato, Y., Schmid, C. (eds.) ECCV 2012, Part IV. LNCS, vol. 7575, pp. 215–230. Springer, Heidelberg (2012)
7. Choi, W., Shahid, K., Savarese, S.: What are they doing?: Collective activity classification using spatio-temporal relationship among people. In: ICCV (2009)
8. Choi, W., Shahid, K., Savarese, S.: Learning context for collective activity recognition. In: CVPR (2011)
9. Dalal, N., Triggs, B.: Histograms of oriented gradients for human detection. In: CVPR (2005)
10. Eslami, S.M.A., Heess, N., Williams, C.K.I., Winn, J.: The shape boltzmann machine: a strong model of object shape. IJCV (2013)
11. Felzenszwalb, P., McAllester, D., Ramanan, D.: A discriminatively trained, multiscale, deformable part model. In: CVPR (2008)
12. Kae, A., Sohn, K., Lee, H., Learned-Miller, E.: Augmenting crfs with boltzmann machine shape priors for image labeling. In: CVPR (2013)
13. Khamis, S., Morariu, V.I., Davis, L.S.: Combining per-frame and per-track cues for multi-person action recognition. In: Fitzgibbon, A., Lazebnik, S., Perona, P., Sato, Y., Schmid, C. (eds.) ECCV 2012, Part I. LNCS, vol. 7572, pp. 116–129. Springer, Heidelberg (2012)
14. Khamis, S., Morariu, V., Davis, L.: A flow model for joint action recognition and identity maintenance. In: CVPR (2012)
15. Lan, T., Sigal, L., Mori, G.: Social roles in hierarchical models for human activity recognition. In: CVPR (2012)
16. Lan, T., Wang, Y., Mori, G.: Discriminative figure-centric models for joint action localization and recognition. In: ICCV (2011)
17. Lan, T., Wang, Y., Yang, W., Robinovitch, S.N., Mori, G.: Discriminative latent models for recognizing contextual group activities. TPAMI (2012)
18. Li, Y., Tarlow, D., Zemel, R.: Exploring complositional high order pattern potentials for structured output learning. In: CVPR (2013)
19. Morariu, V.I., Davis, L.S.: Multi-agent event recognition in structured scenarios. In: Computer Vision and Pattern Recognition (CVPR) (2011)
20. Odashima, S., Shimosaka, M., Kaneko, T., Fukui, R., Sato, T.: Collective activity localization with contextual spatial pyramid. In: Fusiello, A., Murino, V., Cucchiara, R. (eds.) ECCV 2012 Ws/Demos, Part III. LNCS, vol. 7585, pp. 243–252. Springer, Heidelberg (2012)

21. Pei, M., Jia, Y., Zhu, S.C.: Parsing video events with goal inference and intent prediction. In: ICCV (2011)
22. Ryoo, M.S., Aggarwal, J.K.: Stochastic Representation and Recognition of High-level Group Activities. IJCV (2011)
23. Wang, S.B., Quattoni, A., Morency, L.P., Demirdjian, D., Darrell, T.: Hidden conditional random fields for gesture recognition. In: CVPR (2006)
24. Wang, Y., Mori, G.: Hidden part models for human action recognition: Probabilistic versus max margin. TPAMI (2011)
25. Weinland, D., Ronfard, R., Boyer, E.: A survey of vision-based methods for action representation, segmentation and recognition. CVIU 115, 224–241 (2011)
26. Yuille, A.L., Rangarajan, A.: The concave-convex procedure. Neural Comput. 15(4), 915–936 (2003)
27. Zeng, Z., Ji, Q.: Knowledge based activity recognition with Dynamic Bayesian Network. In: Daniilidis, K., Maragos, P., Paragios, N. (eds.) ECCV 2010, Part VI. LNCS, vol. 6316, pp. 532–546. Springer, Heidelberg (2010)
28. Zhu, Y., Nayak, N.M., Roy-Chowdhury, A.K.: Context-aware modeling and recognition of activities in video. In: CVPR (2013)