Human Interaction Understanding With Joint Graph Decomposition and Node Labeling

Zhenhua Wang, Jinchao Ge, Dongyan Guo, Jianhua Zhang, *Senior Member, IEEE*, Yanjing Lei, and Shengyong Chen, *Senior Member, IEEE*

***Abstract*— The task of human interaction understanding involves both recognizing the action of each individual in the scene and decoding the interaction relationship among people, which is useful to a series of vision applications such as camera surveillance, video-based sports analysis and event retrieval. This paper divides the task into two problems including grouping people into clusters and assigning labels to each of them, and presents an approach to solving these problems in a joint manner. Our method does not assume the number of groups is known beforehand as this will substantially restrict its application. With the observation that the two challenges are highly correlated, the key idea is to model the pairwise interacting relations among people via a complete graph and its associated energy function such that the labeling and grouping problems are translated into the minimization of the energy function. We implement this joint framework by fusing both deep features and rich contextual cues, and learn the fusion parameters from data. An alternating search algorithm is developed in order to efficiently solve the associated inference problem. By combining the grouping and labeling results obtained with our method, we are able to achieve the semantic-level understanding of human interactions. Extensive experiments are performed to qualitatively and quantitatively evaluate the effectiveness of our approach, which outperforms state-of-the-art methods on several important benchmarks. An ablation study is also performed to verify the effectiveness of different modules within our approach.**

***Index Terms*— Human interaction understanding, joint grouping and labeling, graph decomposition.**

I. Introduction

HUMAN interaction might change across time. For example, non-interacting people at a time-stamp become interacting later. Also there are scenarios including multiple concurrent human activities (see Figure 1). Consequently, it is inaccurate to predict for each video or image just one category of human activities as that done by human action recognition [1], [2]. To address this, we consider the human interaction understanding (HIU) task [3], [4] in this paper, which aims at assigning each individual in a time-stamp (video frame) an action label, meanwhile decoding the interactive relations (*i.e.*, who is interacting with whom) among people in the scene. Specifically, given an input video which might contain multiple individuals, we tackle HIU in a frame-level, and decompose it into two complementary sub-tasks. One task is the atomic prediction of the action category of each individual. The second task is to group people into different clusters such that people within the same cluster are interacting while people belong to distinct clusters have no interaction with each other. Figure 1 shows an example HIU. Although this task could be viewed as solving two independent classification problems (one for action classification and the second for interaction classification) such that various deep models [1], [2], [5]–[10] are applicable to them, as an effective fine-grained post-processing step, however, structured models [3], [4], [11], [12], [12] have shown to be able to boost HIU performance mainly because they exploit both deep content-based features and human interaction context.

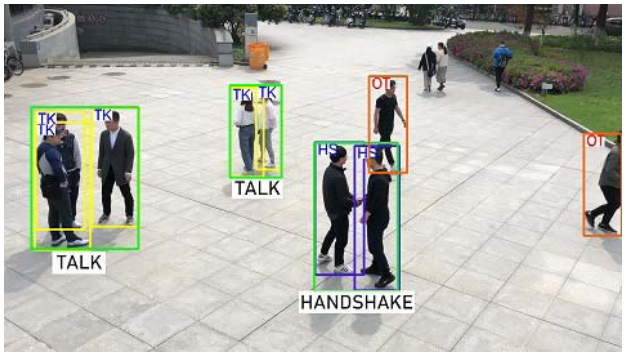

Fig. 1. Given an input image and the detected human bounding boxes, HIU requires to decode both the person-wise action labels and their interaction configuration, that is, which individuals are interacting with each other. Here multiple interacting individuals are grouped into identical clusters represented by green rectangles. TK, HS and OT are action labels denoting *talk*, *handshake* and *others*, respectively.

A canonical structured-model for HIU typically involves two consecutive steps [4], [13]. The first step decodes the pairwise interaction relations by training a binary classifier, either throng shallow or deep models. The second step takes as input the interaction recognition results, the deep features

Manuscript received October 9, 2020; revised March 18, 2021 and May 11, 2021; accepted June 22, 2021. Date of publication July 5, 2021; date of current version July 13, 2021. This work was supported in part by the National Key Research and Development Program of China under Grant 2018YFB1305200 and in part by the National Natural Science Foundation of China under Grant 61802348, Grant 62002325, Grant 61876167, and Grant 62020106004. The associate editor coordinating the review of this manuscript and approving it for publication was Prof. Guo-Jun Qi. *(Corresponding author: Shengyong Chen.)*

Zhenhua Wang, Jinchao Ge, Dongyan Guo, and Yanjing Lei are with the College of Computer Science and Technology, Zhejiang University of Technology, Hangzhou 310023, China (e-mail: zhhwang@zjut.edu.cn; guodongyan@zjut.edu.cn; leiyj@zjut.edu.cn).

Jianhua Zhang is with the College of Computer Science and Technology, Zhejiang University of Technology, Hangzhou 310023, China, and also with the School of Computer Science and Engineering, Tianjin University of Technology, Tianjin 300222, China (e-mail: zjh@zjut.edu.cn).

Shengyong Chen is with the School of Computer Science and Engineering, Tianjin University of Technology, Tianjin 300222, China (e-mail: sy@ieee.org).

Digital Object Identifier 10.1109/TIP.2021.3093383

1941-0042 © 2021 IEEE. Personal use is permitted, but republication/redistribution requires IEEE permission.
See https://www.ieee.org/publications/rights/index.html for more information.

Authorized licensed use limited to: West Virginia University. Downloaded on February 06,2022 at 13:41:12 UTC from IEEE Xplore. Restrictions apply.

of atomic actions, and the contextual features to construct a conditional random field (CRF) to model the relations within human interactions. Let n be the number of people in the scene, and let $\mathbf{x}$ be a frame. The CRF graph $G = (V, E)$ owns a node set $V = \{1, \ldots, n\}$ and an edge set E, where each node $i \in V$ and the associated discrete variable $y_i \in \mathcal{Y}$ represent the action category of person i, and each edge $\{u, v\} \in E \subset V^2$ indicates that the associated people have interaction with each other. Edges E is typically set up according to the interaction classification results, specifically, by adding edges between all pairs of interacting individuals according to the classification. The core component of CRF models is an energy function evaluating the cost to pay when assigning a particular $\mathbf{y} = [y_i]_{i=1}^n$ to the nodes V. The energy function decomposes according to the graph structure into a series of node and edge energies. Specifically, for each $i \in V$, its node energy can be defined as $\theta_i(y_i) = \langle \mathbf{w}_1, \phi_i(\mathbf{x}, y_i) \rangle$, which is a dot product between the parameter vector $\mathbf{w}_1$ and the local feature $\phi_i(\mathbf{x}, y_i)$. This feature is usually a softmax-score output by a network when the label takes y_i. Likewise, for each $(u, v) \in E$, edge energy can be defined as $\theta_{u,v}(y_u, y_v) = \langle \mathbf{w}_2, \psi_{u,v}(\mathbf{x}, y_u, y_v) \rangle$, which is again a dot product between the parameter vector $\mathbf{w}_2$ and the edge feature $\psi_{u,v}(\mathbf{x}, y_u, y_v)$. Here $\psi_{u,v}(\mathbf{x}, y_u, y_v)$ denotes a vector embedding of the contextual features prepared for the interacting pair (u, v) and a particular assignment of their action labels. Usually, the edge feature takes information like content-based convolutional neural network (CNN) descriptors, the 2D distance between people, their head orientations and the compatibility between the associated action labels. The model parameters $\mathbf{w} = [\mathbf{w}_1, \mathbf{w}_2]$ could be learned from data using structured SVM [14]. In order to predict $\mathbf{y}$, one can minimize the total energy defined as:

$$\min_{\mathbf{y}} \sum_{i \in V} \theta_i(y_i) + \sum_{\{u,v\} \in E} \theta_{u,v}(y_u, y_v). \tag{1}$$

The CRF defined by Equation (1) can be interpreted as a probabilistic distribution $\mathbb{P}(\mathbf{y} \mid \mathbf{w}, \mathbf{x}) \propto \exp(\sum_i -\theta_i(y_i) - \sum_{\{u,v\}} \theta_{u,v}(y_u, y_v))$. To find the best $\mathbf{y}$, one can use loopy belief propagation (LBP) algorithm to solve the optimization (1) efficiently and approximately.

An important observation is that the labeling of actions of people in a scene and their interaction configuration are highly correlated. Hence an interesting question is how to model this correlation to improve the HIU performance. To this end, a number of approaches have been proposed for the joint estimation of the atomic actions $\mathbf{y}$ and the pairwise interaction configuration $\mathbf{z}$ [3], [12], [15], [16]. An effective formulation proposed in [15][1] is

$$\min_{\mathbf{y},\mathbf{z}} \sum \theta_i(y_i) + \sum \theta_{u,v}(y_u, y_v)(1 - z_{u,v}) + \rho_{u,v}(z_{u,v}), \tag{2}$$

where $\mathcal{E} = N^2 \setminus \{\{i, i\}\}_{i=1}^n$ contains all possible pairwise interactions (edges). Each $\{u, v\} \in \mathcal{E}$ is associated with a binary variable $z_{u,v} \in \{0, 1\}$, where $z_{u,v} = 0$ indicates the related people are interacting, and $z_{u,v} = 1$ means that they are not interacting. Here $\rho_{u,v}(z_{u,v}) = \langle \mathbf{w}_3, \mu_{u,v}(\mathbf{x}, z_{u,v}) \rangle$, where the feature vector $\mu_{u,v}(\mathbf{x}, z_{u,v})$ embeds the binary label and a series of features, which typically include the 2D distance between people, their head orientations and a softmax score (obtained via a CNN) measuring the confidence when the label is predicted as $z_{u,v}$. Note when $\mathbf{z}$ is fixed, the optimization (2) degenerates to the CRF model (1). Likewise, this joint model of $\mathbf{y}$ and $\mathbf{z}$ can be interpreted in a probabilistic way by constructing a joint distribution of them with the objective function of (2). In [15], the optimization problem (2) is solved approximately with an alternating search algorithm, and all model parameters $[\mathbf{w}_1, \mathbf{w}_2, \mathbf{w}_3]$ are trained with stochastic gradient descent.

[1]In this paper, we reformulate the original formulation of [15] for our convenience of presentation.

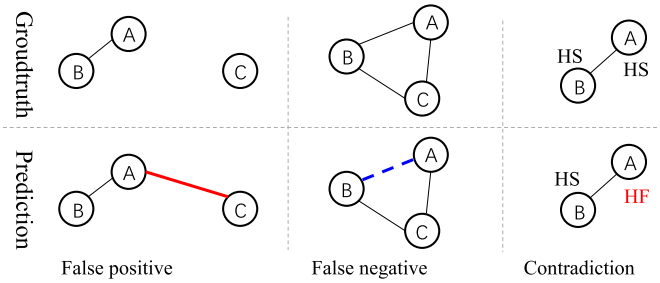

Fig. 2. Three types of issues raised by the joint formulation in Equation (2). The first row shows the groudtruth of three types with edges representing the interacting people, and the second row lists the corresponding predictions of the three types. HS means *handshake* and HF means *highfive handshake*. See text for details.

Nevertheless, the above joint model has neither constraints on the interaction-relation-predictions across different person-pairs, nor constraints on the consistency between action and interaction predictions. Consequently, the prediction probably contains issues as that illustrated in Figure 2 (each column shows an issue), given the imperfectness of the feature representation and the sub-optimality of the inference solver. The left-most column shows a false alarm of interaction between (A, C), ignoring the predictions that (A, B) are interacting and (B, C) have no interaction. In the middle, the model incorrectly predicts that A and B are not interacting even when both interaction predictions between (A, C) and (B, C) are positive. In the rightmost, the prediction says that (A, B) are interacting, which clearly contradicts with the action prediction results (A performs handshake while B performs highfive). To address these issues, we propose to incorporate two pieces of human knowledge in the joint model. The first piece is the transitivity of the pairwise interaction relations, and the second piece enforces the consistency between the atomic action labels of two people who are interacting. Leveraging such prior knowledge, we formulate HIU as a joint problem of decomposing a dense graph and labeling its nodes. We demonstrate such formulation is able to absorb various deep features as well as contextual cues to enhance the representation, propose an alternating search algorithm to solve the related inference problem efficiently, and we conduct ablation study to explore the effectiveness of key components in our model. Finally, experimental results on multiple benchmarks demonstrate our approach achieves the new state-of-the-art.

Authorized licensed use limited to: West Virginia University. Downloaded on February 06,2022 at 13:41:12 UTC from IEEE Xplore. Restrictions apply.

II. Related Work

A. Closely Related Tasks

As a task of video analysis, HIU has several siblings including human action recognition (HAR) [1], [17], collective activity recognition (CAR) [8], [13], action detection (AD) [18] and human-object interaction recognition (HOIR) [19], [20]. While these tasks share the common requirement of extracting robust motion-appearance features to describe human actions, HIU differs from other tasks saliently. HAR and CAR are of classification tasks as they simply require assigning a unique label to each input video, while HIU is much more challenging as it requires decoding the interaction structure as well as assigning action labels to all participants. Different from HIU, AD pays particular attention to detect spatial-temporal locations of human actions without considering the interactions among people. For HOIR, the focus is the recognition of interactions between human and non-human objects, which is different from human-human interactions considered by HIU. A survey of HIU is provided in [21].

B. HIU With Fixed Interaction Prediction

Previous work [4] tackles HIU in two separate steps. The first step is to determine the pairwise interaction relations via a trained binary classifier. Then its second step predicts the action labels for all people by building a spatial-temporal CRF, where the CRF graph is created according to the interaction classification result of the first step. In this sense, errors of interaction classification in the first step could harm subsequent action predictions.

C. HIU With Joint Interaction and Action Prediction

The idea of treating person-person interaction configuration (if they have interaction or not) as unknown and inferring it jointly with person-wise action labels has been studied by previous works [3], [12], [15], [16]. The chief observation is that action labels and interaction configuration among people preserve strong correlations and the correct identification of one of them should be helpful to the recognition of the other. Also the joint models are able to fuse various shallow representations (usually HoG and HoF [22]), deep CNN representations (typically VGGNet [5], ResNet [6], CapsNet [23], 3D ConvNet, Two-stream ConvNets [1] and Temporal Segment Networks [2]) as well as high-level contextual information (popular ingredients are head orientations, 2D distances and the co-occurrence of action labels) of human interactions to improve the performance of HIU. The mentioned joint models concentrate on obtaining effective formulations of human interactions and how to solve the related inference problem efficiently, while they neglect the issues illustrated by Figure 2. Our recent work [24] proposed a novel formulation which tackles the false positive and false negative issues within the prediction of interaction configuration. In this paper, we extend this preliminary work by 1) enhancing the original energy function by introducing new energy terms, 2) providing more evaluations for the proposed approaches, and 3) showing that our new solution achieves the new state-of-the-art of HIU.

D. HIU and Deep Graphical Representation

A recent advance is to integrate deep CNN (and/or RNN) and CRF to jointly learn deep representations and contextual features for CAR [8], [25]–[28]. However, these methods are designed for CAR, which assume that only one category of human activities exists in a sample. Hence these graphical representations cannot deal with multiple concurrent human interactions (a vital requirement of HIU).

E. Joint Grouping and Labeling

It has been shown that the joint grouping/clustering and labeling of nodes in graphs avails a range of applications, *e.g.* pedestrian tracking [29], pose estimation [30] and unsupervised graph network learning [31]. This work is inspired by the subgraph decomposition work proposed in [29], [32] and the joint graph decomposition and labeling technique proposed in [30]. In order to track multiple targets across time, work [29] leverages a graphical representation where nodes encode human body detection and edges represent if two bounding boxes belong to the same track of an object. Consequently, tracking translates to a minimum-cost-cut of such graphs into separate components, and each component corresponds to one track. Work [30] extends the graph model of [29] by adding node variables and modifying the cost function to facilitate the joint estimation of cuts and node labels. One typical application of this joint framework is pose estimation, where node variables represent the joint classes and edge variables denote if two joints belong to the same articulation. In this paper, we reformulate the objective function of the joint framework [30] to enable the simultaneous training of all model parameters using a unique structured model, and propose a new inference algorithm to solve the related inference problem efficiently (see Figure 3 for an overview of our approach). To our best knowledge, this is the first work that models the HIU task as the joint graph decomposition and node labeling. In conjunction with deep action representations and contextual features of human interactions, the proposed joint framework achieves the new state-of-the-art on multiple benchmarks of HIU.

III. Formulation

Given any frame $\mathbf{x}$ in a video and the associated n detected human bodies indexed by a set $N = \{1, 2, \ldots, n\}$, in order to understand the human interactions among these n people, we need to solve two sub-tasks as illustrated by Figure 3. The first task is to decompose all targets into $x \in N$ mutually exclusive subsets $\mathcal{C} = \{C_s\}_{s=1}^{x}$, where $C_s \subseteq N, \bigcup_s (C_s) = N$, $C_s \bigcap C_t = \emptyset \ \forall s \neq t, s, t \in \{1, \ldots, x\}$. Note x is unknown beforehand. The decomposition $\mathcal{C}$ can be equivalently represented by a matrix $\mathbf{z} = (z_{s,t})_{s \in N, t \in N}$, where $z_{s,t} \in \{0, 1\}$, $z_{s,t} = 0$ means s and t belong to the identical group, and $z_{s,t} = 1$ indicates they belong to separate groups. The second task is the prediction of a labeling of all targets to represent the actions of each individual, which is represented by $\mathbf{y} = (y_1, \ldots, y_n)$, where $y_i \in \mathcal{Y} \ \forall i$. Solving these two tasks provides us the result illustrated by Figure 1 such that targets grouped into the same cluster have interaction with

Authorized licensed use limited to: West Virginia University. Downloaded on February 06,2022 at 13:41:12 UTC from IEEE Xplore. Restrictions apply.

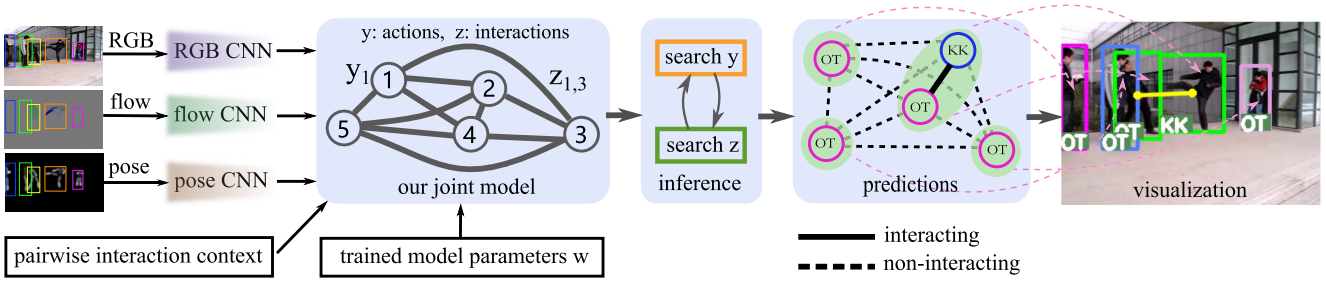

Fig. 3. The proposed framework for human activity understanding. Given any frame in a video as input, we detect human bodies, extract optical flow and estimate human poses at first. Then we extract per-person CNN features from RGB channel, flow channel and pose channel respectively using body detection as region of interest (ROI). These features, together with contextual information of person-person interactions are taken to build our joint model, which is illustrated by a complete graph with nodes representing the action labels of people, and edges encode the relation if any pair of targets are interacting or not. The prediction corresponds to minimizing an energy function to assign action labels to all nodes and to decompose the complete graph into a series of complete sub-graphs (shaded regions). Each generated sub-graph corresponds to a group of interacting people. Here *KK* means *kick* and *OT* denotes *others* (including all human actions beyond primary ones). Note a *KK* person and an *OT* person are grouped together since they are interacting). The prediction is visualized in the right-most diagram where colors of bounding boxes encode different groups.

each other, and vice versa. The specific interaction category of each group, such as handshake, hug and talk can be determined by summarizing the action labels of people within this group.

A. Existing Formulation

Let $\mathcal{G} = (\mathcal{V}, \mathcal{E})$ be a complete graph with n nodes, where $\mathcal{V} = N$ denotes its node set and $\mathcal{E} = \{\{i, j\}\}_{i\in\mathcal{V}, j\in\mathcal{V}, i\neq j}$ denotes its edge set. Here $\{i, j\}$ represents an undirected edge of $\mathcal{G}$. According to Equation (7) in [30], the HIU problem can be defined in the form

$$\begin{aligned} &\min_{\mathbf{y},\mathbf{z}} \sum_i \theta_i(y_i) + \sum_{\{u,v\}\in\mathcal{E}} \theta_{u,v}(y_u, y_v)(1 - z_{u,v}) \\ &\text{s.t.} \quad z_{u,v} \leq \sum\nolimits_{\{a,b\}\in\mathcal{O}\setminus\{\{u,v\}\}} z_{a,b}, \\ &\qquad \forall\{u, v\} \in \mathcal{O}, \forall\mathcal{O} \in \mathrm{CC}(\mathcal{G}), \end{aligned} \tag{3}$$

where $\mathrm{CC}(\mathcal{G})$ denotes a set containing all chordless cycles within $\mathcal{G}$. As $\mathcal{G}$ is complete, $\mathrm{CC}(\mathcal{G}) = \{\{i, j, k\}|i, j, k \in \mathcal{V}, i \neq j, j \neq k, i \neq k\}$. The cycle-constraints in Equation (3) define a feasible set of the optimization, such that its solution $\mathbf{z}^*$ actually delivers a valid decomposition $\mathcal{C} = \{c_i\}_{i=1}^N$ of $\mathcal{G}$, where each $c \in \mathcal{C}$ is complete. Indeed, each chordless cycle $\mathcal{O}$ contains exactly three nodes, and each node triplet of $\mathcal{G}$ together with their edges form a chordless cycle in $\mathcal{G}$. It is easy to check that the constraints in Equation (3), when applying to any chordless cycle $\mathcal{O}$, guarantee generating a valid decomposition of $\mathcal{O}$. Note that the cycle constraints are applied to all chordless cycles. Consequently, minimizing the energy function in Equation (3) generates valid decomposition $\mathcal{C}$ of $\mathcal{G}$, with each decomposed graph c being complete. With this formulation, we could avoid the false-positive and false-negative issues in Figure 2. Note that in comparison with the formulation (2), the only difference within the above formulation is the additional linear constraints, and the optimization (3) is more challenging to solve. Fortunately, high-quality approximations could be found efficiently using the KLj algorithm proposed in [30].

B. Our Reformulation

Our reformulation contains three ingredients: 1) We add a data-independent penalty term for the inconsistent labeling of any pair of people who are interacting; 2) We add a data-dependent term to enhance the prediction of $\mathbf{z}$ variables; 3) We parameterize the entire energy function such that all model parameters could be learned from data in a joint fashion.

1) Labeling Consistency: The penalty term has the form

$$\zeta(y_s, y_t) = \begin{cases} \omega & \text{if } (y_s, y_t) \in \mathcal{P}_c, \\ \omega_{\max} & \text{otherwise.} \end{cases} \tag{4}$$

Here $\mathcal{P}_c$ denotes a set that contains all pairwise consistent labeling such as (handshake, handshake), (hug, hug), (pass, receive) and (receive, pass). With this function, any inconsistent labeling incurs a penalty of $\omega_{\max}$, while a consistent labeling has a reward ω. We will learn such penalties to encourage consistent predictions of human actions such that the contradiction issue illustrated by Figure 2 could be resolved. Different from $\theta_{u,v}(y_u, y_v)$, which is a joint feature representation of data and labels, the labeling consistency term ζ is data-independent and its output only depends on the action labels of the associated targets.

2) Data-Driven Interaction Prediction: Taking features extracted from images as input, the function $\rho_{u,v}(z_{u,v})$ (defined by Equation (9)) outputs the cost when $z_{u,v}$ takes a specific value.

3) Reformulation: Absorbing ζ and ρ, the objective function in (3) becomes

$$\begin{aligned} \sum_{i\in\mathcal{V}} \theta_i(y_i) + \sum_{\{u,v\}\in\mathcal{E}} \Big\{ &\big[\theta_{u,v}(y_u, y_v) + \zeta(y_u, y_v)\big](1 - z_{u,v}) \\ &+\rho_{u,v}(z_{u,v})\Big\}. \end{aligned} \tag{5}$$

Note the formulation in our primary model *CGD* [24] is identical to (5) except that the former does not takes the edge energies $\theta_{u,v}(y_u, y_v)$ and the features used to calculate $\theta_i(y_i)$ are different (see Section III-C for details). Hence we call our new formulation in this paper *CGD-New*. Essentially, we combine the formulations proposed in [24] and [30] (Equation (3)) to obtain a more powerful representation for the joint prediction of $\mathbf{y}$ and $\mathbf{z}$.

4) Parameterization: We now parameterize the energy function (5) by defining each energy term as a dot product between

Authorized licensed use limited to: West Virginia University. Downloaded on February 06,2022 at 13:41:12 UTC from IEEE Xplore. Restrictions apply.

its specific feature and parameter vectors:

$$\theta_i(y_i) = \langle \phi_i(y_i), \mathbf{w}_u \rangle, \tag{6}$$

$$\theta_{u,v}(y_u, y_v) = \langle \phi_{j,k}(y_j, y_k), \mathbf{w}_p \rangle \tag{7}$$

$$\zeta(y_j, y_k) = \langle \eta(y_j, y_k), \mathbf{w}_c \rangle, \tag{8}$$

$$\rho_{j,k}(z_{j,k}) = \langle \psi^0_{j,k}, \mathbf{w}^0_g \rangle (1 - z_{j,k}) + \langle \psi^1_{j,k}, \mathbf{w}^1_g \rangle z_{j,k}. \tag{9}$$

Above $\mathbf{w}_c = [\omega, \omega_{\max}]$, $\langle \mathbf{a}, \mathbf{b} \rangle$ denotes the inner product of vectors $\mathbf{a}$ and $\mathbf{b}$, ϕ, η, ψ denote the so-called joint features in structured prediction [14] with details given in Section III-C. With such definitions, problem (5) translates into

$$\begin{aligned} &J_{\mathbf{w}}(\mathbf{y}, \mathbf{z}; \mathbf{x}, \mathcal{G}) \\ &= \frac{1}{N} \sum_i \langle \phi_i(y_i), \mathbf{w}_u \rangle + \frac{1}{R} \sum_{\{j,k\}} \Big\{ \\ &\quad \Big[\langle \phi_{j,k}(y_j, y_k), \mathbf{w}_p \rangle + \langle \psi^0_{j,k}, \mathbf{w}^0_g \rangle + \langle \eta(y_j, y_k), \mathbf{w}_c \rangle \Big] (1 - z_{j,k}) \\ &\quad + \langle \psi^1_{j,k}, \mathbf{w}^1_g \rangle z_{j,k} \Big\}, \end{aligned} \tag{10}$$

where $\mathbf{w} = [\mathbf{w}_u, \mathbf{w}_p, \mathbf{w}^0_g, \mathbf{w}^1_g, \mathbf{w}_c]$ are model parameters to be learned from data (detailed in Section III-D), N and R are normalization constants with $R = \frac{N \cdot (N-1)}{2}$.

Given $\mathbf{x}$, $\mathbf{w}$ and $\mathcal{G}$, we find the best $\mathbf{y}$, $\mathbf{z}$ through solving

$$\begin{aligned} \min_{\mathbf{y},\mathbf{z}} \quad & J_{\mathbf{w}}(\mathbf{y}, \mathbf{z}; \mathbf{x}, \mathcal{G}) \\ \text{s.t.} \quad & z_{u,v} \leq \sum\nolimits_{\{a,b\} \in \mathcal{O} \setminus \{\{u,v\}\}} z_{a,b}, \\ & \forall \{u, v\} \in \mathcal{O}, \forall \mathcal{O} \in \mathrm{CC}(\mathcal{G}), \end{aligned} \tag{11}$$

which is NP-complete in general and we provide an efficient solution (with approximations) in Section IV.

Compared with the joint graph decomposition and labeling formulation (3) [30], the differences of our formulation (11) include three aspects. First, we use dense graphs $\mathcal{G}$ instead of sparse ones to guarantee the inclusion of important subtle edges which could be potentially excluded in sparse structures. For example, in basketball games, passing and receiving a basketball can involve two players distant from each other, and building sparse graphs according to the distance heuristic has a risk of excluding such useful connections. Second, we introduce an additional labeling consistency term to penalize unreasonable assignment of action labels to pairs of targets who have interaction with each other. Finally, instead of training each energy term separately, our parameterization of the energy function allows us to train all energy terms jointly (detailed in Section III-D).

C. Feature Representation

Our feature representation combines three data streams including RGB image, optical flow and human poses. The motivation here is that we take optical flow (extracted from adjacent frames) to learn the motion feature of dynamic human actions, which complements the static appearance feature extracted from RGB images. The pose stream includes key elements of human bodies and excludes cluttered background and occluded foreground in complex scene. Hence we incorporate it to learn intrinsic representations of human actions. To prepare data for feature extraction, we detect human bodies for each video frame $\mathbf{x}$, extract optical flow for the image, and estimate per-person body poses using the cropped images from their bounding box regions as ROIs. Some examples of flow and pose images are provided in Figure 4. The joint features in Equation (10) are defined as

Fig. 4. Examples of estimated optical-flow and human poses. Each column corresponds one out of three examples. The top row shows input images, the middle row presents flow images and the bottom row demonstrates human poses.

$$\phi_i(y_i) = \big[p^a_i(y_i), p^m_i(y_i), p^s_i(y_i), \bar{p}_i(y_i), 1 \big], \tag{12}$$

$$\phi_{j,k}(y_j, y_k) = [\bar{p}_j(y_j) + \bar{p}_k(y_k), \bar{p}_j(y_j) \cdot \bar{p}_k(y_k), 1], \tag{13}$$

$$\psi^0_{j,k} = \big[p^a_{j,k}, p^m_{j,k}, \bar{p}_{j,k}, \kappa^1_{j,k}, \kappa^2_{j,k}, \kappa^e_{j,k} \big], \tag{14}$$

$$\psi^1_{j,k} = \big[1 - p^a_{j,k}, 1 - p^m_{j,k}, 1 - \bar{p}_{j,k}, \kappa^1_{j,k}, \kappa^2_{j,k}, \kappa^e_{j,k} \big], \tag{15}$$

$$\eta(y_j, y_k) = \big[\mathbb{1}(y_j, y_k), 1 - \mathbb{1}(y_j, y_k) \big]. \tag{16}$$

1) Appearance and Motion Feature: Above $p^a_i(y_i)$, $p^m_i(y_i)$ represent the soft-max probabilities of assigning an action label y_i to person i, which are calculated by appearance and motion CNNs [1] taking as input the image patch occupied by person i. Similarly, $p^a_{j,k}$ and $p^m_{j,k}$ are soft-max probabilities (obtained by applying appearance and motion CNNs trained for interaction classification) when the interaction classification for person j and person k is positive. We prepare data for such classification in a way depicted by Figure 5. Specifically, for each pair of detected human bodies, we create a black background image whose size equals the area of the tightest bounding rectangle of the detected targets. Afterwards patches cropped from human body regions in the original image are embedded into the background image such that the distance between human bodies is always preserved.

2) Pose Feature: Human poses have been widely exploited in the community of action recognition [33]–[35]. For each instance of estimated human poses, we create a gray-scale pose silhouette as shown in Figure 4. We observed that some body parts, such as arms and legs are more discriminative than others in terms of action recognition. Hence we made them brighter than the less discriminative parts in the image of pose silhouette. Taking as inputs such silhouette images, we train a residual CNN [6] to compute a soft-max probability $p^s_i(y_i)$ to represent the confidence of assigning an action label y_i to the associated person. Note that the primary work [24] of this paper does not incorporate human poses.

Authorized licensed use limited to: West Virginia University. Downloaded on February 06,2022 at 13:41:12 UTC from IEEE Xplore. Restrictions apply.

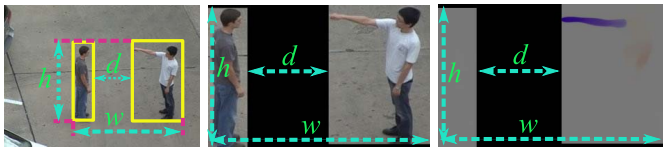

Fig. 5. Creating inputs for CNN to predict if two people are interacting or not. Left shows an image and its detected human bodies. Middle and right show the created inputs for appearance and motion CNNs. Note distances (denoted by d) are always preserved within the created data.

3) Aggregated Feature: To further enhance the feature presentation, we average the appearance feature, the motion feature and the pose feature to generate $\bar{p}_i(y_i)$ for action classifications, and we average the appearance and motion features to obtain $\bar{p}_{j,k}$ for interaction classification.

4) Contextual Feature: We take two types of contextual features to capture the discriminative properties of human interactions. The first type is characterized by $\mathbb{1}(y_j, y_k)$ which gives 1 if $(y_s, y_t) \in \mathcal{P}_c$, and gives 0 in rest cases. Consequently, the dot product between η and $\mathbf{w}_c$ recovers the label consistency term ζ in Equation (5) when $\mathbf{w}_c = [\omega, \omega_{\max}]$. The second type corresponds to vectors κ^1, κ^2 and κ^e in Equation (14) and (15), which embeds the bounding box information using equations defined by

$$\kappa^1_{j,k} = [\upsilon_{j,k}, h_{j,k}, d_{j,k}, s_{j,k}], \tag{17}$$

$$\kappa^2_{j,k} = [\upsilon^2_{j,k}, h^2_{j,k}, d^2_{j,k}, s^2_{j,k}], \tag{18}$$

$$\kappa^e_{j,k} = [e^{-\upsilon_{j,k}}, e^{-h_{j,k}}, e^{-d_{j,k}}, e^{-s_{j,k}}], \tag{19}$$

where $\upsilon_{j,k}, h_{j,k}, d_{j,k}, s_{j,k}$ are calculated by

$$\upsilon_{j,k} = \frac{|\upsilon_j - \upsilon_k|}{\max(\upsilon_j, \upsilon_k)}, \quad h_{j,k} = \frac{|h_j - h_k|}{\max(h_j, h_k)},$$
$$d_{j,k} = \frac{\|\mathbf{c}_j - \mathbf{c}_k\|_2}{d_{\max}}, \quad s_{j,k} = \frac{|s_j - s_k|}{\max(s_j, s_k)}. \tag{20}$$

Here $\upsilon_i, h_i, s_i, \mathbf{c}_i$ represent the width, height, area and center coordinate of the bounding box i respectively, and $d_{\max}$ denotes the largest Euclidean distance between human bodies, which is found by enumerating all pairs of bounding boxes within the same frame in the training set.

D. Parameter Learning

Let $D = \{(\mathbf{x}^{(i)}, \mathbf{z}^{(i)}, \mathbf{y}^{(i)})\}_{i=1}^M$ be a training set of M examples, where each instance contains both labeling groundtruth $\mathbf{y}$ and interaction configuration groundtruth $\mathbf{z}$. To train $\mathbf{w}$, we use a large-margin-style formulation:

$$\min_{\mathbf{w}} \frac{1}{2}\|\mathbf{w}\|^2 + \frac{\lambda}{M}\sum\nolimits_{i=1,\ldots,M} \xi_i,$$
$$\text{s.t. } J_{\mathbf{w}}(\hat{\mathbf{y}}^{(i)}, \hat{\mathbf{z}}^{(i)}) - J_{\mathbf{w}}(\mathbf{y}^{(i)}, \mathbf{z}^{(i)}) \geq \Delta(\mathbf{y}^{(i)}, \hat{\mathbf{y}}^{(i)}, \mathbf{z}^{(i)}, \hat{\mathbf{z}}^{(i)}) - \xi_i, \forall \hat{\mathbf{y}}^{(i)}, \hat{\mathbf{z}}^{(i)}, \ \xi_i \geq 0 \ \forall i \in \{1, \ldots, M\}, \tag{21}$$

where the label cost Δ is defined by

$$\Delta(\mathbf{y}, \hat{\mathbf{y}}, \mathbf{z}, \hat{\mathbf{z}}) = \frac{1}{N}\sum_i \delta(y_i \neq \hat{y}_i) + \frac{1}{R}\sum_{\{j,k\}} \delta(z_{j,k} \neq \hat{z}_{j,k}). \tag{22}$$

Here $\delta(\cdot)$ is an indicator function which gives 1 if the associated inequality holds, otherwise it outputs 0. In order to learn all parameters via gradient descent, we transform (21) into an equivalent form without constraints:

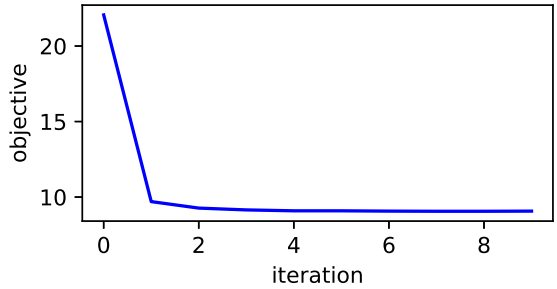

Fig. 6. Train $\mathbf{w}$ on BIT using mini-batch gradient descent. The objective in Equation (23) gradually converges to a fixed point.

$$\frac{1}{2}\|\mathbf{w}\|^2 + \frac{\lambda}{M}\sum_{i=1}^{M}\Big[J_{\mathbf{w}}(\mathbf{y}^{(i)}, \mathbf{z}^{(i)}) + \max_{\hat{\mathbf{y}}^{(i)}, \hat{\mathbf{z}}^{(i)}}\big[-J_{\mathbf{w}}(\hat{\mathbf{y}}^{(i)}, \hat{\mathbf{z}}^{(i)}) + \Delta(\mathbf{y}^{(i)}, \hat{\mathbf{y}}^{(i)}, \mathbf{z}^{(i)}, \hat{\mathbf{z}}^{(i)})\big]\Big]_+, \tag{23}$$

where $[a]_+ = \max(0, a)$. We define the concatenation of all aggregated joint features for any $\mathbf{x}$ and its solution $(\hat{\mathbf{y}}, \hat{\mathbf{z}})$ as

$$\Psi(\hat{\mathbf{y}}, \hat{\mathbf{z}}) = \Big\{\frac{1}{N}\sum_{\upsilon}\phi_\upsilon(\hat{y}_\upsilon), \frac{1}{R}\Big[\sum_{\{\{j,k\}|\hat{z}_{j,k}=0\}}\theta_{j,k}(\hat{y}_j, \hat{y}_k), \sum_{\{\{j,k\}|\hat{z}_{j,k}=0\}}\psi^0_{j,k}, \sum_{\{\{j,k\}|\hat{z}_{j,k}=1\}}\psi^1_{j,k}, \sum_{\{\{j,k\}|\hat{z}_{j,k}=0\}}\eta(\hat{y}_j, \hat{y}_k)\Big]\Big\}. \tag{24}$$

Then we have $J_{\mathbf{w}}(\hat{\mathbf{y}}, \hat{\mathbf{z}}) = \langle \mathbf{w}, \Psi(\hat{\mathbf{y}}, \hat{\mathbf{z}})\rangle$. For any training instance $(\mathbf{x}', \mathbf{y}', \mathbf{z}') \in D$, it is straightforward to derive that the sub-gradient of (23) with respect to $\mathbf{w}$ is

$$\mathbf{w} + \frac{\lambda}{M}\big[\Psi(\mathbf{y}', \mathbf{z}') - \Psi(\mathbf{z}^*, \mathbf{y}^*)\big], \tag{25}$$

where $(\mathbf{z}^*, \mathbf{y}^*)$ is the solution of

$$\min_{\hat{\mathbf{y}}, \hat{\mathbf{z}}} J_{\mathbf{w}}(\hat{\mathbf{y}}, \hat{\mathbf{z}}) - \Delta(\mathbf{y}', \hat{\mathbf{y}}, \mathbf{z}', \hat{\mathbf{z}})$$
$$\text{s.t. } \hat{z}_{u,\upsilon} \leq \sum\nolimits_{\{a,b\}\in\mathcal{O}\setminus\{\{u,\upsilon\}\}} \hat{z}_{a,b}, \quad \forall\{u, \upsilon\} \in \mathcal{O}, \forall\mathcal{O} \in \mathrm{CC}(\mathcal{G}). \tag{26}$$

With the sub-gradients calculated via (25), we learn $\mathbf{w}$ from data with mini-batch gradient descent. As shown by Figure 6, the algorithm is able to find a local solution in a few iterations for the optimization problem (21).

IV. Inference

Our observation is that the objective function in Equation (26) can be translated into the same form as the objective function in Equation (11). Consequently, both optimizations could be approximately and efficiently solved with the algorithms introduced in this section.

Authorized licensed use limited to: West Virginia University. Downloaded on February 06,2022 at 13:41:12 UTC from IEEE Xplore. Restrictions apply.

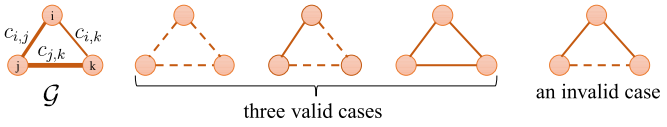

Fig. 7. Valid and invalid decomposition of a complete graph $\mathcal{G}$ (thicker edges indicate higher costs) with three nodes. The decomposition is achieved via cutting some edges off (setting $z = 1$, which is visually depicted by the dashed edges) such that the constraints in (28) is satisfied and the total energy is minimized. Rightmost shows an invalid solution which violates the constraints.

A. Problem Reduction

On one hand, fixing the grouping variable unchanged as $\mathbf{z}^*$, the optimization (11) reduces to

$$\min_{\mathbf{y}} \sum_i \frac{\theta_i(y_i)}{N} + \sum_{\{\{j,k\}|z^*_{j,k}=0\}} \frac{\theta_{j,k}(y_j, y_k) + \zeta(y_j, y_k)}{R}. \quad (27)$$

Problem (27) is well-known as the discrete energy minimization [36], which is NP-complete in general. Here we simply use loopy belief propagation to solve it approximately.

On the other, fixing $\mathbf{y}$ unchanged as $\mathbf{y}^*$, the inference problem (11) reduces to

$$\begin{aligned} &\min_{\mathbf{z}} \sum_{\{\{j,k\}\}} c_{j,k} \cdot z_{j,k}, \\ &\text{s.t.} \quad \text{the constraints in the problem (11),} \end{aligned} \quad (28)$$

where

$$c_{j,k} = \frac{\rho_{j,k}(1) - \rho_{j,k}(0) - \zeta(y^*_j, y^*_k) - \theta_{j,k}(y^*_j, y^*_k)}{R}. \quad (29)$$

Solving the problem (28) is conceptually the same as decomposing a complete graph $\mathcal{G}$ into a set of smaller complete graphs such that the objective is minimized. Note each edge $\{j,k\} \in \mathcal{E}$ is associated with a cut cost $c_{j,k}$, which is learned from data and can be either positive or non-positive. With such costs, the decomposition of $\mathcal{G}$ is achieved by cutting some edges of $\mathcal{G}$ off (equivalent to set $z = 1$) so as to meet the constraints and to minimize the energy. Figure 7 shows a toy example of $\mathcal{G}$ with three nodes as well as its feasible and infeasible decomposition results.

B. Complete Graph Decomposition

Note the 0-1 integer linear programming (28) is also NP-complete. In order to solve it efficiently, we provide a heuristic solution yet effective in practice. Specifically, we start from an arbitrary decomposition $\mathcal{C}^0 = \{C^0_i\}_{i=1}^x$, which is feasible to (28), and iteratively decrease the energy using two elementary operations:

- *Merging operation*: merge two partitions to generate a larger united partition.
- *Moving operation*: move a node from one partition to another.

Here, which operation to choose relies on several heuristics introduced next.

1) Inner and Outer Costs: Consider any two elements C_i and C_j in $\mathcal{G}$. For any $a \in C_i$, the inner cost I^a_i is the summation of all costs $c_{a,s}, \forall s \in C_i \setminus \{a\}$, and the outer cost $O^a_{i,j}$ is the summation of all costs $c_{a,t}, \forall t \in C_j$.

Proposition 1: Merging C_i and C_j decreases the energy by $\Delta_{i+j} = \sum_{a \in C_i} O^a_{i,j}$.

Proof: Let A denote the total cost of the partition excluding C_i and C_j. Then the total partition cost after merging C_i and C_j is exactly A. Since the total partition cost before merging C_i and C_j is $A + \sum_{a \in C_i} O^a_{i,j}$, the proposition holds. □

Proposition 2: Moving a node k from C_i to C_j does not change the feasibility of the resulting decomposition and decreases the energy by $O^k_{i,j} - I^k_i$.

Proof: Note that C_i and C_j are two connected components of $\mathcal{G}$. Hence the moving operation does not change the feasibility. To calculate the decrease, we denote by B the total partition cost excluding $O^k_{i,j}$. Then the total partition cost before moving is $B + \sum_{s \in C_j} c_{k,s} = B + O^k_{i,j}$. After moving, the total partition cost is $B + I^k_i$ since the outer cost of k after moving equals its inner cost before moving. The proposition is proved. □

Proposition 3: Consider moving a node k from C_i to C_j again. For any $s \in C_i \setminus \{k\}$, let $O^{'s}_{i,j}, I^{'s}_i$ denote the outer and inner costs of s before moving, and let $O^{''s}_{i,j}, I^{''s}_i$ denote the outer and inner costs of s after moving. For any $t \in C_j$, let $O^{'t}_{j,i}, I^{'t}_j$ denote the outer and inner costs of t before moving, and let $O^{''t}_{j,i}, I^{''t}_j$ denote the outer and inner costs of t after moving. We have

$$O^{''s}_{i,j} - I^{''s}_i = O^{'s}_{i,j} - I^{'s}_i + 2c_{k,s}, \quad (30)$$

$$O^{''t}_{j,i} - I^{''t}_j = O^{'t}_{j,i} - I^{'t}_j - 2c_{k,t}. \quad (31)$$

Proof: Equation (30) holds because $O^{''s}_{i,j} - I^{''s}_i = (O^{'s}_{i,j} + c_{k,s}) - (I^{'s}_i - c_{k,s})$. Similarly Equation (31) also holds. □

Definition 1: Consider moving a sequence of nodes from C_i to C_j to decrease the energy in a greedy way. We call the sequence that decreases the energy most the largest moving sequence*, which is denoted by $S_{i \to j}$. The total energy-decrease earned by these moving operations is denoted by $\Delta_{i \to j}$.*

The compute graph decomposition algorithm (pseudo-code provided in Algorithm 1) starts with an arbitrary initialization $\mathcal{C}$ (which is feasible to (28)), our decomposition algorithm iteratively decreases the objective using the heuristics induced by the propositions. During each iteration, the algorithm chooses a pair of components from the current decomposition for updating. The updating routine maintains for each node of the selected components the difference between its inner and outer costs. It modifies the components with a series of merging and/or moving operations. Which operation to choose depends on how much they decrease the energy according to Proposition 1 and 2. If a moving operation is undertaken, the algorithm updates the differences between inner and outer costs of relevant nodes according to Equation (30) and (31). After the modification, the updating routine tries to split each component to produce a number of unit components (each

Authorized licensed use limited to: West Virginia University. Downloaded on February 06,2022 at 13:41:12 UTC from IEEE Xplore. Restrictions apply.

Algorithm 1 Pseudo-code to solve (28).

Input: $\mathcal{G}$, $c_{j,k}\forall\{j,k\} \in \mathcal{G}$, threshold ϵ.
Initialization: Let $\mathcal{C} = \{\mathcal{G}\}$, $\sigma^0 = \epsilon + 1, \sigma^1 = 0$.
// T iterations at most.
for $t \in \{1, \ldots, T\}$ **do**
  // Stop as the energy decrease is vanishing.
  if $\sigma^{t-1} - \sigma^t \leq \epsilon$ **then**
    break
  end
  $\sigma^t \longleftarrow 0$.
  while *choose* (C_i, C_j) *from* $\mathcal{C}$ **do**
    /* Get the largest moving sequences (Definition 1) and the corresponding local energy decreases earned by moving (Definition 1) and merging (Proposition 1) operations. */
    Compute $S_{i\to j}$, $S_{j\to i}$, Δ_{i+j}, $\Delta_{i\to j}$, $\Delta_{j\to i}$.
    /* Select an operation (merging, moving from i to j, or moving from j to i) and perform it. */
    if $\Delta_{i+j} > \Delta_{i\to j}$ & $\Delta_{i+j} > \Delta_{j\to i}$ **then**
      $\mathcal{C} \longleftarrow \mathcal{C} - \{C_i, C_j\} + \{C_i\} \cup \{C_j\}$,
      $\sigma^t \longleftarrow \sigma^t + \Delta_{i+j}$.
    else if $\Delta_{i\to j} > \Delta_{j\to i}$ **then**
      $C_i \longleftarrow C_i - S_{i\to j}$, $C_j \longleftarrow C_j + S_{i\to j}$,
      $\sigma^t \longleftarrow \sigma^t + \Delta_{i\to j}$.
    else
      $C_i \longleftarrow C_i + S_{j\to i}$, $C_j \longleftarrow C_j - S_{j\to i}$,
      $\sigma^t \longleftarrow \sigma^t + \Delta_{j\to i}$.
    end
  end
  // Split the partitions to produce new unit partitions.
  for $u \in C_i, v \in C_j$ **do**
    $C_k \longleftarrow \{\}, C_l \longleftarrow \{\}$.
    if $O^u_{i,k} - I^u_i > 0$ **then**
      $C_i \longleftarrow C_i - \{u\}$, $\mathcal{C} \longleftarrow \mathcal{C} + \{u\}$.
    end
    if $O^v_{j,l} - I^u_j > 0$ **then**
      $C_j \longleftarrow C_j - \{v\}$, $\mathcal{C} \longleftarrow \mathcal{C} + \{v\}$.
    end
  end
end
Return $\mathcal{C}, \sigma^t$.

unit component contains only one node) till the component is non-separable (*i.e.* no gain can be made by any further separation). At the end of each iteration, we calculate the total decrease of the objective earned by all updates and splits, and early stop the decomposition if no progress was attained. Figure 8 depicts the increase of running-time of the proposed complete graph decomposition algorithm as the number of nodes increases from 3 to 100. For each case, we randomly generate $c_{j,k}$ $\forall\{j,k\} \in \mathcal{E}$ 100 rounds, and calculate their mean time-cost. One can see that the proposed algorithm is quite efficient for problems up to a moderate-scale (this is the case for the human interaction understanding task and the fashion grouping task in Section V), while it takes several seconds to solve the problem with a larger number of nodes.

Algorithm 2 The inference algorithm to solve (11).

Input: $N, \phi, \psi, \eta, \mathcal{G}, \mathbf{w}, \epsilon$.
Initialization: Let $z^0_{s,t} = 0$ $\forall\{s,t\} \in \mathcal{E}$.
for $t \in \{1, \ldots, T\}$ **do**
  Find best individual action labels $\mathbf{y}^t$ with loopy belief propagation (*i.e.,* solving (27)) and $\mathbf{z}^{t-1}$;
  Update $c_{s,t}$ $\forall\{s,t\} \in \mathcal{E}$ using $\mathbf{y}^t$;
  Find best graph-decomposition $\mathcal{C}^t$ via Algorithm 1 (*i.e.,* solving (28));
  Calculate $z^t_{s,t}$ $\forall\{s,t\} \in \mathcal{E}$.
end
Return $\mathbf{y}^T$ and $\mathbf{z}^T$.

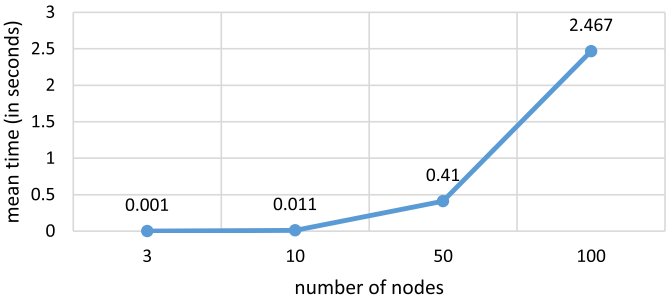

Fig. 8. The average running time of the complete graph decomposition algorithm (Algorithm 1) as the number of nodes increases.

2) Inference With Alternating Search: The pseudo-code for approximately solving the challenging problem (11) is provided in Algorithm IV-B.1. The algorithm iteratively (in T iterations) searches over $\mathbf{y}$ (by solving (27)) and $\mathbf{z}$ (by solving (28)) directions in turn. Though there is no guarantee of convergence, we found in practice that this local search algorithm is able to improve the solution of (11) progressively, and it usually converges in 5 iterations.

V. Experiments and Results

We show the effectiveness of our proposed approach by comparison against the state-of-the-arts, and verify the functions of different modules of our model via an ablation study.

A. Dataset

We evaluate our method on two benchmarks of HIU, which are briefly introduced below.

UT-Interaction (UT) [37] is a benchmark for human interaction recognition. The dataset includes seven human actions in total: six of them are specific actions including *handshake (HS)*, *hug (HG)*, *kick (KK)*, *point (PT)*, *punch (PC)* and *push (PS)*, and the rest one is *others (OT)*, which means any other actions performed by one person. It has 20 short videos which cover both symmetric (for example, *HS* involves two handshaking targets) and asymmetric actions (for instance, *KK* involves one person who is kicking the other labels as *OT*). UT videos are officially divided into two sets, and each set contains 10 videos. We use the first set for training and the second set for testing. The consistent labeling set $\mathcal{P}_c$ on this dataset is {(*HS, HS*), (*HG, HG*), (*KK, OT*), (*PT, OT*), (*PC, OT*), (*PS, OT*)}.

Authorized licensed use limited to: West Virginia University. Downloaded on February 06,2022 at 13:41:12 UTC from IEEE Xplore. Restrictions apply.

BIT-Interaction (BIT) [11] contains nine action classes including *bend (BD), box (BX), handshake (HS), highfive (HF), hug (HG), kick (KK), pat (PT), push (PS), others (OT)*, and each class includes 50 videos. Different from UT, each frame of BIT contains exactly one interaction class. The consistent labeling set $\mathcal{P}_c$ on this dataset is {(*BD, OT*), (*BX, OT*), (*HS, HS*), (*HF, HF*), (*HG, HG*), (*KK, OT*), (*PT, OT*), (*PS, OT*)}.

B. Methods and Implementation Details

1) Baselines: We consider CNNs with different types of inputs as baseline approaches. For action classification, we use three different inputs including RGB images cropped from ROIs, optical-flow data cropped from ROIs and human silhouette images as that shown in Figure 4. While for interaction classification, we only use RGB images and flow images created by the method described in Section III-C. We use the same ResNet101 architecture [6] for different types of inputs. To enhance the representation for action recognition, we also fuse features extracted from RGB image and optical-flow with the classic two-stream architecture in [1].

We compare against five state-of-the-art approaches:

- CGD [24]. Results are picked from the primary version [24] of this paper. As noted in Section III-B, the earlier version lacks the $\theta_{u,v}$ term in comparison with our reformulation (5). In addition, our new model also incorporates pose information to enhance the feature representation in Equation (12).
- CGD-Pose. This method is identical to CGD except that the pose information is included here.
- Spatial-temporal CRF (ST-CRF) [4]. This method processes actions and interactions separately. First a binary classifier is trained to determine if any pair of individuals are interacting or not. Then a CRF model is taken to fuse learned descriptors, hand-engineered descriptors (HoG and HoF) and spatial-temporal context to recognize per-person actions.
- Joint + AS. The existing method (2) proposed in [15].
- KLj-r. The existing formulation (3) proposed in [30].

For fair comparison, all methods share the same definitions of energy terms unless otherwise specified.

2) Evaluation Metrics: To evaluate the performance of different approaches, we quantify HIU in terms of classifications on both per-person actions (muti-classification) and pairwise human interactions (binary classification). Our evaluation metrics include *precision*, *recall* and *F1-score* (for multi-classification we calculate metrics for each label, and find their unweighted mean).

3) Implementation Details: For any two consecutive frames, we extract optical flow from them using FlowNet 2.0 [38]. To estimate human poses, we use OpenPose [39]. For Pose CNN, we use a ResNet-101 architecture [6]. For the training of all joint optimization models, the batch size is 256, the trade-off parameter λ is 10^4, and the learning rate is 10^{-4}. The training typically converges in 20 iterations. For UT, we choose the first 10 videos as training data and the remaining 10 videos as testing. While for BIT, the first 34 videos of each class are used for training and the rest for testing, as suggested by the creator of this dataset. For testing, we detect human bodies for each frame using YOLO [40]. Since these datasets are quite small (compared with dataset like ImageNet), we augment training examples via rotating, cropping and flipping transformations. Then we sample training examples separately for each action-class to reduce the imbalance on numbers of training instances of different classes. To train the action classifier, we implemented the two-steam net [1] (a popular deep net for human action recognition) in PyTorch using ResNet-101 [6] as the architecture of image and flow CNNs, where image CNN takes a $224 \times 224 \times 3$ RGB image as input and flow CNN takes a $224 \times 224 \times 2$ optical-flow data as input. We extract optical flow using FlowNet2.0 [38]. We pretrain both CNNs on UCF 101 from scratch, then finetune their weights on UT and BIT using mini-batch gradient descent with momentum (0.9). The initial learning rate is 0.001 decreased by a factor of 0.1 every 7 epochs, and the training stops in 50 epochs. Then, we train our CGD model using mini-batch gradient descent.

TABLE I

COMPARISON AGAINST THE STATE-OF-THE-ARTS ON UT. HERE "R-**y**", "P-**y**" AND "F1-**y**" DENOTE RECALL, PRECISION AND F1-SCORE RESPECTIVELY ON INDIVIDUAL ACTION CLASSIFICATION, AND "R-**z**", "P-**z**" AND "F1-**z**" MEAN RECALL, PRECISION AND F1-SCORE RESPECTIVELY ON INTERACTION CLASSIFICATION. THE BEST RESULTS ARE IN BOLD FOR EACH COLUMN

Method	R-**y**	P-**y**	F1-**y**	R-**z**	P-**z**	F1-**z**
ST-CRF [4]	0.625	0.778	0.681	0.826	**0.901**	0.859
KLj-r [30]	0.592	0.759	0.647	0.734	0.638	0.670
Joint + AS [15]	0.583	0.755	0.639	0.839	0.878	0.857
CGD (our [24])	0.580	0.739	0.629	0.842	0.886	0.863
CGD-Pose (ours)	0.624	0.778	0.681	**0.875**	0.850	0.862
CGD-New (ours)	**0.635**	**0.781**	**0.690**	0.856	0.872	**0.864**

TABLE II

COMPARISON AGAINST THE STATE-OF-THE-ARTS ON BIT

Method	R-**y**	P-**y**	F1-**y**	R-**z**	P-**z**	F1-**z**
ST-CRF [4]	0.620	0.625	0.618	0.905	0.911	0.908
KLj-r [30]	0.583	0.680	0.617	0.656	0.649	0.652
Joint + AS [15]	0.587	**0.685**	0.620	**0.919**	0.927	0.923
CGD (our [24])	0.628	0.638	0.628	0.870	0.949	0.905
CGD-Pose (ours)	**0.631**	0.642	0.632	0.866	**0.953**	0.904
CGD-New (ours)	0.623	0.666	**0.638**	0.910	0.941	**0.924**

C. Comparison Against the State-of-the-Art

1) Quantitative Evaluation: The results are provided in Table I and Table II for experiments on UT and BIT respectively, from which we can draw four conclusions. First, our *CGD-New* performs significantly better than *KLj-r* on both datasets (11.9 points on UT and 14.7 points on BIT in terms of mean F1 scores), which suggest that it is vital to restrain incompatible action labels (through incorporating the ζ terms in Equation (5)) and to explicitly model the pairwise interacting relations within the energy function (*e.g.*, using the ρ terms in Equation (5)). Second, *Joint + AS* performs much worse than our approach (2.9 points less on UT and 0.95 points less on BIT in terms of mean F1 scores) though it also predicts actions and interacting relations jointly. The reason is that *Joint + AS* does not address the issues illustrated

Authorized licensed use limited to: West Virginia University. Downloaded on February 06,2022 at 13:41:12 UTC from IEEE Xplore. Restrictions apply.

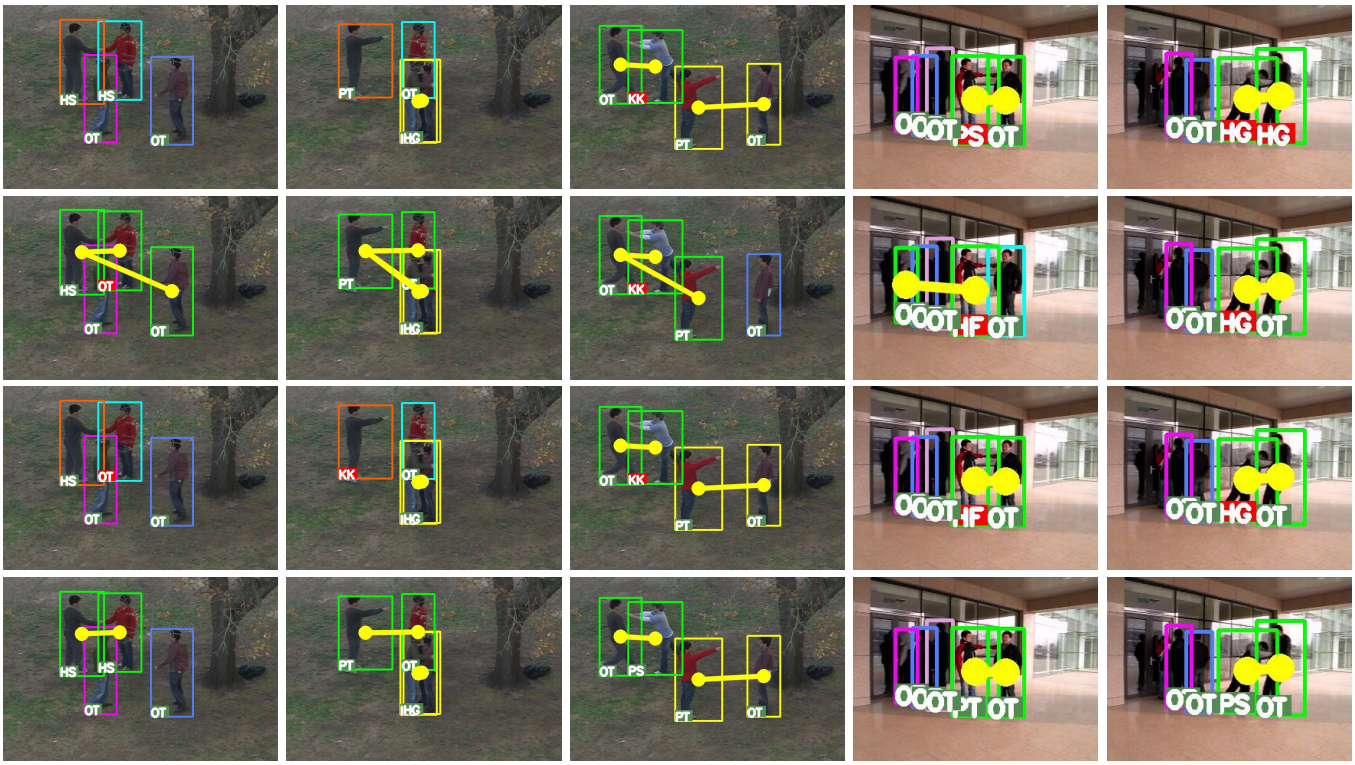

Fig. 9. Visualization of action labeling and people grouping (best viewed in color). Each column represents one example. From top to bottom, the four rows correspond to results of *ST-CRF*, *KLj-r*, *Joint + AS* and *CGD-New* respectively. Here *KK*, *HS*, *HF*, *PT*, *PS*, *HG* and *OT* denote seven action classes including *kick*, *handshake*, *highfive*, *point*, *push*, *hug* and *others*. Note that individuals belonging to the same group take the same color, and are connected by edges. Correct action predictions are marked by white texts with dark green background, and incorrect ones are highlighted by red background. Note these examples contain long-range human-human interactions and it is impossible to perfectly identify all interacting instances according to heuristics. While our approach is able to predict correct grouping and labeling results, other state-of-the-art approaches tend to make defective predictions to an extent.

by Figure 2, while our model is proposed to resolve these problems by leveraging the labeling consistency (4) and the cycle constraints in (11). Here we would like to note that compared with *Joint + AS* [15] (see Equation (2) for its energy function), which gives the second best result in terms of F1-**z** on BIT (92.3% *v.s.* 92.4% of *CGD-New*), our proposed *CGD-New* (see Equation (5) for its formulation) has not introduced any data-driven term for the prediction of **z**, but takes a new set of the so-called cycle constraints to enforce the labeling consistency of **z** predictions, meanwhile it introduces a new data-driven term ζ for the prediction of **y**. It turned out that the newly introduced data term has played a more important role than the cycle constraints, and the performance gain on **y**-predictions is more salient than that of the **z** predictions. Third, compared against *ST-CRF*, our *CGD-New* performs much better on BIT (1.8 points higher), which validates the importance of modeling actions and interacting relations jointly. Note that the benefits earned by the joint modeling are less distinguished on UT (only 0.7 points higher). The reason is probably that most UT images only contain two individuals, and *ST-CRF* with fixed graphs (built according to the pairwise-interacting-relation classification results with a pre-trained binary classifier) is adequate to model human interactions in this dataset. Note that in terms of **z**-predictions, the state-of-the-art result on UT is 85.9%, which is achieved by *ST-CRF* [4]. Our *CGD-New* gets 86.4% on this dataset, which improves by 0.5 points compared against the best result. The reason the improvement is limited is probably because *ST-CRF* incorporates additional temporal information to learn the labeling consistency across time, while our *CGD-New* focuses on the joint estimations of person-wise action labels and pairwise interactive relations and neglects the temporal representation for simplicity. Nevertheless, the proposed *CGD-New* performs consistently better than *ST-CRF* in terms of both **y**-predictions and **z**-predictions. Finally, thanks to the introduced pose feature and the pairwise θ terms in Equation (5), our *CGD-New* achieves salient performance gains (5.5 points on BIT and 2.6 points on UT in terms of mean F1 scores) in comparison with our primary method CGD.

The state-of-the-art result on UT is 85.9%, which is achieved by *ST-CRF* [10]. Our *CGD-New* gets 86.4% on this dataset, which improves by 0.5 points compared against *ST-CRF*. The reason that the improvement is limited is probably because that *ST-CRF* incorporates additional temporal information to learn the labeling consistency across time, while our *CGD-New* neglects such information for simplicity.

2) Qualitative Evaluation: In order to show that our approach is able to address the issues illustrated by Figure 2, we visualize the prediction results on a few examples in Figure 9. We can see that our model (*CGD-New*) gives correct labeling and coherent grouping for all examples, while other approaches tend to make defective predictions especially when the images contain long-range interactions (*e.g.*, the second, third and fourth columns).

Authorized licensed use limited to: West Virginia University. Downloaded on February 06,2022 at 13:41:12 UTC from IEEE Xplore. Restrictions apply.

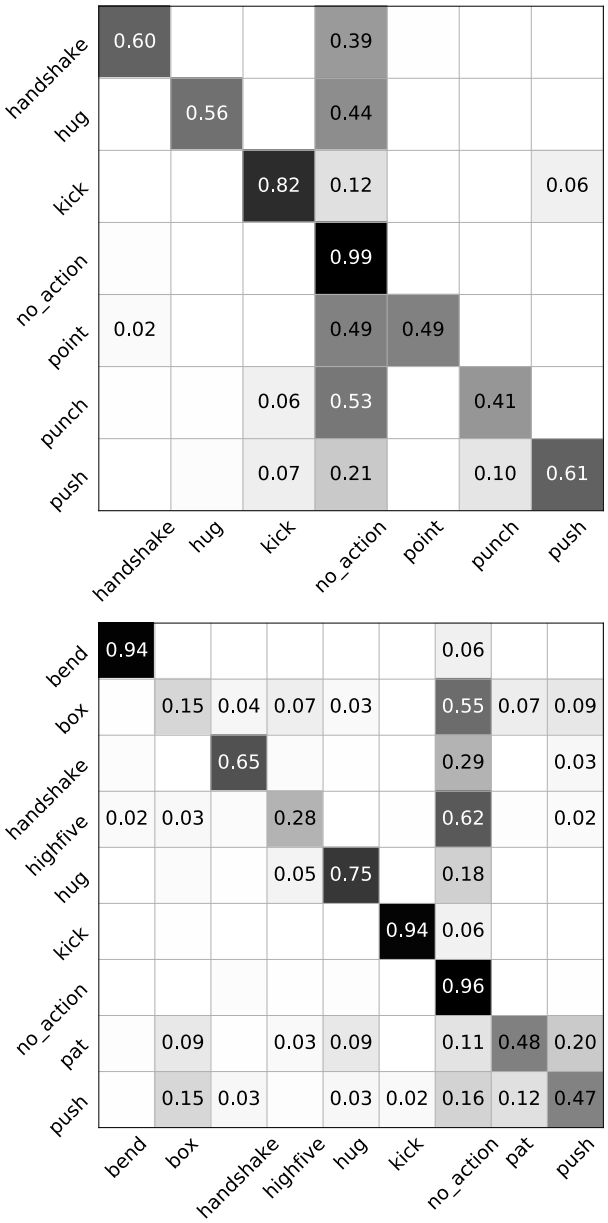

Fig. 10. Confusion matrices for the classification of individual actions. The top is UT and the bottom is BIT.

3) Confusion Matrices: Though the proposed *CGD-New* outperforms the state-of-the-arts, its performance is still far from perfect especially on the prediction of individual actions (the **y** variables). In order to further inspect the challenges of the task, we plot the confusion matrices on action and interaction classifications (see Figure 10 and Figure 11). The chief observation is that all datasets are dominated by the *no-action* class, which induces the training to take more care of it to decrease the loss. Though there exist many techniques to tackle the imbalance data in machine learning, this topic is out of the scope of this paper.

D. Ablation Study

1) CNNs With Different Streams: We first explore the performance of CNNs with different streams of input on UT (Table III) and BIT (Table IV). Clearly *two-stream CNN* [1], which fuses the features of image stream and flow stream, performs best on both datasets. It also outperforms *Image CNN* and *Flow CNN* by a large margin, which indicates that optical flow and RGB data are complementary to each other in terms of representing human actions. It worth noting that *RGB CNN* performs much better than *Flow CNN*. This might be because instances of several action classes, like *no_action*, *hug*, *handshake* and *point*, last without (or with

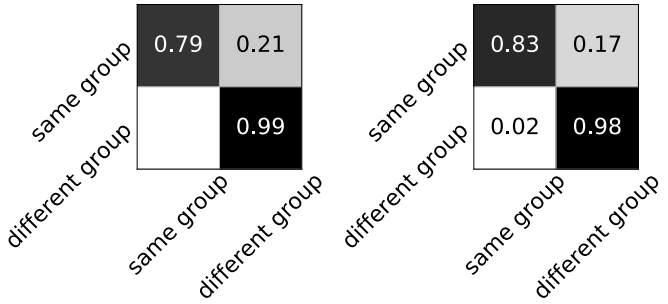

Fig. 11. Confusion matrices for the binary classification of pairwise interacting relations. Left is UT and right is BIT.

TABLE III
THE PERFORMANCE OF DEEP CNNS ON UT USING DIFFERENT STREAMS

Method	R-**y**	P-**y**	F1-**y**	R-**z**	P-**z**	F1-**z**
Image CNN	0.539	0.698	0.576	**0.863**	0.801	0.828
Flow CNN	0.470	0.563	0.505	0.687	0.868	0.744
Pose CNN	**0.690**	0.596	0.585	N/A	N/A	N/A
two-stream CNN	0.560	0.748	0.626	0.826	**0.901**	0.859
CGD-New (ours)	0.635	**0.781**	**0.690**	0.856	0.872	**0.864**

TABLE IV
THE PERFORMANCE OF DEEP CNNS ON BIT USING DIFFERENT STREAMS

Method	R-**y**	P-**y**	F1-**y**	R-**z**	P-**z**	F1-**z**
Image CNN	0.566	0.642	0.594	**0.910**	0.904	0.907
Flow CNN	0.443	0.404	0.413	0.810	0.825	0.817
Pose CNN	0.514	0.589	0.534	N/A	N/A	N/A
two-stream CNN	0.585	0.650	0.609	0.905	0.911	0.908
CGD-New (ours)	**0.623**	**0.666**	**0.638**	**0.910**	**0.941**	**0.924**

TABLE V
ABLATION STUDY ON UT BY REMOVING DIFFERENT COMPONENTS WITHIN OUR CGD-NEW MODEL. HERE THE SUFFIXES "-CC", "-θ", "-ζ" AND "-ρ" INDICATE REMOVING THE *Cycle Constraints*, THE *Pairwise θ Term*, THE *ζ Term* AND THE *ρ Term* RESPECTIVELY, WHILE KEEPING ALL REST COMPONENTS UNCHANGED

Method	R-**y**	P-**y**	F1-**y**	R-**z**	P-**z**	F1-**z**
CGD-New-CC	0.625	0.776	0.681	0.837	0.771	0.799
CGD-New-θ	0.624	0.778	0.681	**0.875**	0.850	0.862
CGD-New-ζ	0.623	0.777	0.679	0.874	0.766	0.809
CGD-New-ρ	0.625	0.778	0.681	0.826	**0.901**	0.859
CGD-New	**0.635**	**0.781**	**0.690**	0.856	0.872	**0.864**

TABLE VI
ABLATION STUDY ON BIT BY REMOVING DIFFERENT COMPONENTS WITHIN OUR CGD-NEW MODEL

Method	R-**y**	P-**y**	F1-**y**	R-**z**	P-**z**	F1-**z**
CGD-New-CC	0.624	0.660	0.637	0.896	0.916	0.906
CGD-New-θ	**0.631**	0.642	0.632	0.866	**0.953**	0.904
CGD-New-ζ	0.629	0.633	0.627	0.877	0.942	0.906
CGD-New-ρ	0.620	0.625	0.618	0.905	0.911	0.908
CGD-New	0.623	**0.666**	**0.638**	**0.910**	0.941	**0.924**

Authorized licensed use limited to: West Virginia University. Downloaded on February 06,2022 at 13:41:12 UTC from IEEE Xplore. Restrictions apply.

Fig. 12. Grouping and labeling fashion images presented within the same landing page (https://www.1688.com) for online cloth-selling. Four testing examples are used here. For each testing example, the three rows from top to bottom show results of ResNet, DBSCAN and our CGD respectively. We put a frame on each image to show its color prediction. The images within the same group are stacked together and different groups are separated with a cyan bar.

faint) motions, which results in degenerated flow signals. *Pose CNN* performs slightly better than *RGB CNN* on UT, while performs much worse on BIT. We observe that compared against UT, BIT images usually contain much smaller individuals, serious occlusions, and a larger number of people. Extracting accurate human poses under such circumstances is much more challenging.

2) Effects of θ, ζ and ρ in CGD-New: We test the effects of these terms by removing each of them one by one, and the results are given by Table V and Table VI. When the cycle constraints are removed from the *GCD-New* formulation (11), the performance on $\mathbf{y}$ sees slight drop compared against our full model, while the performance on $\mathbf{z}$ decreases drastically. Hence cycle constraints are important to ensure valid decomposition of interacting groups. The effects of removing θ, ζ and ρ on BIT is more significant than on UT, probably because a more challenging dataset like BIT requires a more sophisticated model.

VI. Application on Fashion Grouping

A. Task Description

Given a stack of cloth images, the task here is predicting each image a label of the *primary color* (the most dominant color) of the cloth, meanwhile grouping these images according to their primary colors. Specifically, if two cloth photos take the same primary color, they need to be grouped together. We call this task *fashion grouping* which is essential to the automatic arrangement of cloth photos in landing pages of online-shopping websites.

B. Fashion Grouping Dataset (FGD)

This dataset crafted by us contains 53 different colors. A few colors are too visually close to be distinguished even with human eyes, and cloth with the identical primary color could be visually distinct due to the change of lighting conditions, which make the fashion grouping task challenging.

Authorized licensed use limited to: West Virginia University. Downloaded on February 06,2022 at 13:41:12 UTC from IEEE Xplore. Restrictions apply.

Fig. 13. Per-color classification accuracy as well as grouping (it can be viewed as a binary classification problem) accuracy using ResNet (red bars), DBSCAN (blue bars) and the proposed CGD (green bars).

For each color, we collect around 1,000 cloth photos from online-shopping websites. Some of these photos simply contain clothes alone, while many others present models wearing clothes. Photos are shot either in outdoor or indoor scenarios, which usually contain complex backgrounds. To extract ROIs for accurate classification of cloth colors, we estimate person joints [39] based on detected human bounding boxes at first. If no person is detected, we just use the entire image; Otherwise, we crop either the top from shoulder to hip (guided by the estimated body joints) by default, or the bottom when failed to extract upper-body joints. Then the cropped images, which are treated as surrogates of the original ones, are taken to perform classification and grouping. We then split all cropped images to produce 1,000 examples for fashion grouping, where each instance contains 3-5 color categories and each color category includes 2-10 cropped images. The consistent labeling set $\mathcal{P}_c$ on this dataset includes all possible color pairs with each pair taking the same color category.

C. Modeling

Our method for fashion grouping follows the modeling paradigm of HIU in this paper. The only difference is that the model here excludes the pairwise terms $\theta_{u,v}$ in (5). We extract deep representations for each cropped image by training a ResNet [6], and define the features for CGD modeling as

$$\phi_i(y_i) = \left[p_i^c(y_i), 1\right], \tag{32}$$

$$\psi_{j,k}^0 = \left[g_{j,k}, \kappa_{j,k}^1, \kappa_{j,k}^2, \kappa_{j,k}^e\right], \tag{33}$$

$$\psi_{j,k}^1 = \left[1 - g_{j,k}, \kappa_{j,k}^1, \kappa_{j,k}^2, \kappa_{j,k}^e\right], \tag{34}$$

$$\eta_{j,k}(y_j, y_k) = \left[\mathbb{1}(y_j, y_k), 1 - \mathbb{1}(y_j, y_k)\right]. \tag{35}$$

Here $p_i^c(y_i)$ represents the soft-max probabilities (the outputs of the soft-max layer of ResNet) of assigning a color label $y_i \in \{1, 2, \ldots, 53\}$ to photo i. As in (16), we append a constant 1 to the $\boldsymbol{\phi}$ feature in order to learn the bias parameter within $\mathbf{w}$. The features $g_{j,k}$, $\boldsymbol{\kappa}^1$, $\boldsymbol{\kappa}^2$ and $\boldsymbol{\kappa}^e$ are defined as

$$g_{j,k} = \frac{\sum_y p_i^c(y_i = y) p_j^c(y_j = y)}{\sum_{y,\hat{y}} p_i^c(y_i = y) p_j^c(y_j = \hat{y})}, \tag{36}$$

$$\kappa_{j,k}^1 = [h_{j,k}, d_{j,k}], \tag{37}$$

$$\kappa_{j,k}^2 = [h_{j,k}^2, d_{j,k}^2], \tag{38}$$

$$\kappa_{j,k}^e = [e^{-h_{j,k}}, e^{-d_{j,k}}], \tag{39}$$

where $g_{j,k}$ is the probability that j and k are of the same primary color based on the outputs of ResNet, $d_{j,k}$ is the L2 distance between the mean HSV colors of j and k, and $h_{j,k}$ denotes the L2 distance between the predicted colors (using the trained ResNet) of j and k in HSV space.

Authorized licensed use limited to: West Virginia University. Downloaded on February 06,2022 at 13:41:12 UTC from IEEE Xplore. Restrictions apply.

TABLE VII
GROUPING AND LABELING RESULTS ON THE FASHION GROUPING DATASET, USING RESNET, DBSCAN AND CGD

Method	R-**y**	P-**y**	F1-**y**	R-**z**	P-**z**	F1-**z**
ResNet [6]	0.735	0.726	0.727	0.786	0.931	0.835
DBSCAN [41]	N/A	N/A	N/A	**0.857**	0.768	0.795
KLj-r [30]	0.735	0.726	0.727	0.522	0.530	0.523
CGD (ours)	**0.748**	**0.739**	**0.740**	0.801	**0.934**	**0.847**

We use 800 examples for training (and validation) and the rest 200 for testing. We train the CGD model with mini-batch gradient descent, where the learning rate is set to 0.001 and decreases by a magnitude every 5 epochs. The training typically converges in 20 epochs.

D. Results and Discussion

Here the proposed CGD is compared against ResNet [6], KLj-r [30] and DBSCAN [41]. For ResNet, the grouping results are obtained by pooling instances with identical color-predictions into the same group. DBSCAN is a popular algorithm for clustering with unknown numbers of clusters. It requires to assign a cost for grouping each pair of instances into the same cluster. Here we set up costs using a linear combination of the $\psi^1_{j,k}$ features defined by Equation (34) (weights of the linear function are trained with a support vector machine). The results are provided in Table VII, which shows that CGD outperforms ResNet and KLj-r by 1.3% (F1-score) in terms of color-classification, and it outperforms ResNet, DBSCAN and KLj-r by $1.2\%, 5.2\%, 32.4\%$ respectively in terms of grouping (F1-score). We also visualize the results of fashion grouping using examples beyond the training and testing sets. The results are shown in Figure 12 (zooming in and best viewed in color). It is easy to find that overall the proposed CGD gives the best labeling and grouping results. Figure 13 demonstrates the per-color classification accuracy as well as the grouping accuracy (can be viewed as a binary classification problem), using ResNet, DBSCAN and our CGD approach. In comparison with ResNet (in red) and DBSCAN (in blue), the proposed CGD (in green) performs notably better on 21 out of 55 classes, and ties on 33 classes, which justifies the effectiveness of the proposed approach.

VII. CONCLUSION

We have presented an approach for human interaction understanding. Our key observation was that the studied problem could be decomposed into a graph decomposition task and a node labeling task over dense graphs. In order to solve the two tasks in a joint manner, we proposed a discrete optimization problem that is solved efficiently and approximately using an alternating search strategy. We demonstrated that the objective function of this discrete optimization could be set up as a linear transformation of deep and contextual features, and the weights of this linear transformation could be learned with stochastic gradient descent. By doing this, we found that RGB data, human poses and optical flow are all important to understand human interactions, and the results could be further improved by exploiting high-level features and labeling consistency. Experimental results demonstrated that the proposed approach outperforms the state-of-the-arts significantly on two benchmarks for human interaction understanding. We have also shown that the proposed joint decomposition and labeling framework could be applied to fashion grouping, which also endows remarkable performance gains.

ACKNOWLEDGMENT

The authors would like to thank the anonymous reviewers fo an earlier version of this work for their constructive feedback, particularly regarding the results in Table I and Table II, and the inference algorithm.

REFERENCES

[1] K. Simonyan and A. Zisserman, "Two-stream convolutional networks for action recognition in videos," in *Proc. Adv. Neural Inf. Process. Syst.*, 2014, pp. 1–11.
[2] L. Wang, Y. Xiong, Z. Wang, Y. Qiao, D. Lin, and X. Tang, "Temporal segment networks: Towards good practices for deep action recognition," in *Proc. Eur. Conf. Comput. Vis.*, 2016, pp. 20–36.
[3] A. Patron-Perez, M. Marszalek, I. Reid, and A. Zisserman, "Structured learning of human interactions in TV shows," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 34, no. 12, pp. 2441–2453, Dec. 2012.
[4] Z. Wang, S. Liu, J. Zhang, S. Chen, and Q. Guan, "A spatio-temporal CRF for human interaction understanding," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 27, no. 8, pp. 1647–1660, Aug. 2017.
[5] K. Simonyan and A. Zisserman, "Very deep convolutional networks for large-scale image recognition," 2014, *arXiv:1409.1556*. [Online]. Available: http://arxiv.org/abs/1409.1556
[6] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2016, pp. 770–778.
[7] Z. Tu, H. Li, D. Zhang, J. Dauwels, B. Li, and J. Yuan, "Action-stage emphasized spatiotemporal VLAD for video action recognition," *IEEE Trans. Image Process.*, vol. 28, no. 6, pp. 2799–2812, Jun. 2019.
[8] X. Shu, J. Tang, G.-J. Qi, W. Liu, and J. Yang, "Hierarchical long short-term concurrent memory for human interaction recognition," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 43, no. 3, pp. 1110–1118, Mar. 2021.
[9] R. Yao, G. Lin, S. Xia, J. Zhao, and Y. Zhou, "Video object segmentation and tracking: A survey," *ACM Trans. Intell. Syst. Technol.*, vol. 11, no. 4, pp. 1–47, 2020.
[10] R. Yao, G. Lin, C. Shen, Y. Zhang, and Q. Shi, "Semantics-aware visual object tracking," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 29, no. 6, pp. 1687–1700, Jun. 2019.
[11] Y. Kong, Y. Jia, and Y. Fu, "Learning human interaction by interactive phrases," in *Proc. Eur. Conf. Comput. Vis.*, 2012, pp. 300–313.
[12] Z. Wang, Q. Shi, C. Shen, and A. van den Hengel, "Bilinear programming for human activity recognition with unknown MRF graphs," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2013, pp. 1690–1697.
[13] W. Choi, K. Shahid, and S. Savarese, "Learning context for collective activity recognition," in *Proc. CVPR*, Jun. 2011, pp. 3273–3280.
[14] I. Tsochantaridis, T. Joachims, T. Hofmann, and Y. Altun, "Large margin methods for structured and interdependent output variables," *J. Mach. Learn. Res.*, vol. 6, no. 2, pp. 1453–1484, 2006.
[15] Z. Wang *et al.*, "Understanding human activities in videos: A joint action and interaction learning approach," *Neurocomputing*, vol. 321, pp. 216–226, Dec. 2018.
[16] Z. Wang, T. Liu, Q. Shi, M. P. Kumar, and J. Zhang, "New convex relaxations for MRF inference with unknown graphs," in *Proc. IEEE/CVF Int. Conf. Comput. Vis. (ICCV)*, Oct. 2019, pp. 9935–9943.
[17] J. Ye, G.-J. Qi, N. Zhuang, H. Hu, and K. A. Hua, "Learning compact features for human activity recognition via probabilistic first-take-all," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 42, no. 1, pp. 126–139, Jan. 2020.
[18] J. Zhao and C. G. M. Snoek, "Dance with flow: Two-in-one stream action detection," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2019, pp. 9935–9944.
[19] Y.-L. Li *et al.*, "Transferable interactiveness knowledge for human-object interaction detection," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2019, pp. 3585–3594.
[20] T. Zhou, W. Wang, S. Qi, H. Ling, and J. Shen, "Cascaded human-object interaction recognition," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2020, pp. 4263–4272.

Authorized licensed use limited to: West Virginia University. Downloaded on February 06,2022 at 13:41:12 UTC from IEEE Xplore. Restrictions apply.

[21] A. Stergiou and R. Poppe, "Analyzing human-human interactions: A survey," 2018, *arXiv:1808.00022*. [Online]. Available: http://arxiv.org/abs/1808.00022
[22] N. Dalal and B. Triggs, "Histograms of oriented gradients for human detection," in *Proc. IEEE Comput. Soc. Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2005, pp. 886–893.
[23] S. Sabour, N. Frosst, and G. Hinton, "Dynamic routing between capsules," in *Proc. Adv. Neural Inf. Process. Syst.*, 2017, pp. 1–11.
[24] J. Ge, Z. Wang, J. Meng, J. Zhang, and S. Chen, "Joint grouping and labeling via complete graph decomposition," in *Proc. Int. Conf. Neural Inf. Process.*, 2019, pp. 497–505.
[25] Z. Deng, A. Vahdat, H. Hu, and G. Mori, "Structure inference machines: Recurrent neural networks for analyzing relations in group activity recognition," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2016, pp. 4772–4781.
[26] M. S. Ibrahim and G. Mori, "Hierarchical relational networks for group activity recognition and retrieval," in *Proc. ECCV*, 2018, pp. 721–736.
[27] M. Qi, Y. Wang, J. Qin, A. Li, J. Luo, and L. Van Gool, "StagNet: An attentive semantic Rnn for group activity and individual action recognition," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 30, no. 2, pp. 549–565, Feb. 2020.
[28] J. Wu, L. Wang, L. Wang, J. Guo, and G. Wu, "Learning actor relation graphs for group activity recognition," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2019, pp. 9964–9974.
[29] S. Tang, B. Andres, M. Andriluka, and B. Schiele, "Subgraph decomposition for multi-target tracking," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2015, pp. 5033–5041.
[30] E. Levinkov, S. Tang, E. Insafutdinov, and B. Andres, "Joint graph decomposition and node labeling by local search," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jul. 2017, pp. 1–9.
[31] L. Yang *et al.*, "Toward unsupervised graph neural network: Interactive clustering and embedding via optimal transport," in *Proc. ICDM*, 2020, pp. 1358–1363.
[32] M. Keuper, E. Levinkov, N. Bonneel, G. Lavoué, T. Brox, and B. Andres, "Efficient decomposition of image and mesh graphs by lifted multicuts," in *Proc. IEEE Int. Conf. Comput. Vis. (ICCV)*, Dec. 2015, pp. 1751–1759.
[33] H. Wang and L. Wang, "Beyond joints: Learning representations from primitive geometries for skeleton-based action recognition and detection," *IEEE Trans. Image Process.*, vol. 27, no. 9, pp. 4382–4394, Sep. 2018.
[34] C. Li, Z. Cui, W. Zheng, C. Xu, R. Ji, and J. Yang, "Action-attending graphic neural network," *IEEE Trans. Image Process.*, vol. 27, no. 7, pp. 3657–3670, Jul. 2018.
[35] Y. Du, Y. Fu, and L. Wang, "Representation learning of temporal dynamics for skeleton-based action recognition," *IEEE Trans. Image Process.*, vol. 25, no. 7, pp. 3010–3022, Jul. 2016.
[36] J. H. Kappes *et al.*, "A comparative study of modern inference techniques for discrete energy minimization problems," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2013, pp. 1328–1335.
[37] M. Ryoo and J. Aggarwal, "UT-interaction dataset, ICPR contest on semantic description of human activities (SDHA)," in *Proc. IEEE Int. Conf. Pattern Recognit. Workshops*, Aug. 2010, p. 4.
[38] E. Ilg, N. Mayer, T. Saikia, M. Keuper, A. Dosovitskiy, and T. Brox, "FlowNet 2.0: Evolution of optical flow estimation with deep networks," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jul. 2017, pp. 2462–2470.
[39] Z. Cao, T. Simon, S.-E. Wei, and Y. Sheikh, "Realtime multi-person 2D pose estimation using part affinity fields," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jul. 2017, pp. 7291–7299.
[40] J. Redmon and A. Farhadi, "YOLO9000: Better, faster, stronger," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jul. 2017, pp. 7263–7271.
[41] M. Ester, H.-P. Kriegel, J. Sander, and X. Xu, "A density-based algorithm for discovering clusters in large spatial databases with noise," in *Proc. Int. Conf. Knowl. Discovery Data Mining*, 1996, pp. 226–231.

Zhenhua Wang received the bachelor's and master's degrees from Northwest A&F University, China, in 2007 and 2010, respectively, and the Ph.D. degree in computer vision from The University of Adelaide, Adelaide, SA, Australia, in 2014. He is currently a Lecturer with the College of Computer Science, Zhejiang University of Technology, Hangzhou, China. His research interests include computer vision and statistical machine learning.

Jinchao Ge received the master's degree in software engineering from the Zhejiang University of Technology, Hangzhou, China, in 2020. She is currently pursuing the Ph.D. degree with The University of Adelaide, Australia. Her research interests include computer vision and deep learning.

Dongyan Guo received the B.S. degree in application mathematics and the Ph.D. degree in pattern recognition and intelligent systems from the Nanjing University of Science and Technology, China, in 2008 and 2015, respectively. Since 2015, he has been a Faculty Member with the College of Computer Science and Technology, Zhejiang University of Technology, Hangzhou, China. His research interests include computer vision and machine learning.

Jianhua Zhang (Senior Member, IEEE) received the Ph.D. degree from the University of Hamburg, Hamburg, Germany, in 2012. He is currently a Professor with the School of Computer Science and Engineering, Tianjin University of Technology, Tianjin, China. His current research interests include SLAM, 3D vision, reinforcement learning, and machine vision.

Yanjing Lei received the Ph.D. degree from Northwestern Polytechnical University. She is currently a Lecturer with the College of Computer Science, Zhejiang University of Technology. Her main research interests include computer vision and wireless rechargeable sensor networks. She is a member of CCF.

Shengyong Chen (Senior Member, IEEE) received the Ph.D. degree in robot vision from the City University of Hong Kong, in 2003. He is currently a Professor with the Tianjin University of Technology, China. He received a fellowship from the Alexander von Humboldt Foundation of Germany and worked with the University of Hamburg, Hamburg, Germany, from 2006 to 2007. His research interests include computer vision, robotics, and image analysis. He is a Fellow of IET. He was a recipient of the National Outstanding Youth Foundation Award of China in 2013.

Authorized licensed use limited to: West Virginia University. Downloaded on February 06,2022 at 13:41:12 UTC from IEEE Xplore. Restrictions apply.