Category-blind Human Action Recognition: A Practical Recognition System

Wenbo Li[1*] Longyin Wen[2*] Mooi Choo Chuah[1] Siwei Lyu[2†]
[1]Department of Computer Science and Engineering, Lehigh University
{wel514,mcc7}@lehigh.edu
[2]Department of Computer Science, University at Albany, SUNY
{lwen,slyu}@albany.edu

Abstract

Existing human action recognition systems for 3D sequences obtained from the depth camera are designed to cope with only one action category, either single-person action or two-person interaction, and are difficult to be extended to scenarios where both action categories co-exist. In this paper, we propose the category-blind human recognition method (CHARM) which can recognize a human action without making assumptions of the action category. In our CHARM approach, we represent a human action (either a single-person action or a two-person interaction) class using a co-occurrence of motion primitives. Subsequently, we classify an action instance based on matching its motion primitive co-occurrence patterns to each class representation. The matching task is formulated as maximum clique problems. We conduct extensive evaluations of CHARM using three datasets for single-person actions, two-person interactions, and their mixtures. Experimental results show that CHARM performs favorably when compared with several state-of-the-art single-person action and two-person interaction based methods without making explicit assumptions of action category.

1. Introduction

Human action recognition is a major component of many computer vision applications, *e.g.*, video surveillance, patient monitoring, and smart homes, to name a few [3]. There have been many approaches developed for recognizing human actions from monocular videos. However, monocular videos are insufficient for the practical applicability of action recognition algorithms in the real-world environment, mainly due to two problems. First, 3D information is lost in monocular videos. Second, a single camera view usually cannot fully capture human action due to the occlusion.

The recent advent of cost-effective depth sensors enables real-time estimation of 3D joint positions of a human skeleton [17]. The availability of 3D joint positions in real time spawns approaches with higher practical applicability for action recognition.

(a) **(b)** **(c)**

Figure 1. Examples of human actions depicting the complexities in action recognition: (a) a multi-person action with each individual performing a single-person action; (b) a multi-person action involving a single-person action and multi-person interaction; (c) a multi-person action that cannot be reduced to the combination of two single-person actions. See texts for more details.

The majority of existing human action recognition methods using sequences of 3D joint positions are designed for two general categories: single-person action [19, 22, 25] and multi-person interaction [12, 13, 23]. However, human actions in real-world scenarios are much more complex because multiple action instances belonging to both categories usually co-exist in a sequence. For example[1], Figure 1(a) describes three actions all belonging to the single-person action category: three children are skating without interacting with each other; Figure 1(b) depicts two actions belonging to two categories respectively: "two children are fighting for a toy" (a two-person interaction) while "a woman lifts two hands to hold her forehead" (a single-person action). Thus, it is more desirable to have a method that can recognize human actions without involving any prior categorization.

However, existing algorithms designed to recognize single-person actions from sequences of 3D joint positions [10, 19, 25] cannot be used to recognize multi-person interactions, and vice versa [4, 23]. This is because the two categories of actions are exclusive by definition. A simple approach fusing methods for different categories to recognize an action in a competitive manner is unlikely to work as shown by the example in Figure 1(c). From the perspective

* indicates equal contributions.
† indicates the corresponding author.

[1]We show frames from videos to illustrate the idea, but the same scenario holds for sequences of 3D joint positions.

of single-person action recognition, the image depicts one person pushing his hand, but considering the other person, the same action can be recognized as a two-person interaction of "patting on the shoulder".

In this paper, we present a unified recognition model for single-person actions and multi-person interactions. Our method uses a sequence of estimated 3D joint positions as input, and outputs actions that occur in such a sequence. As such, we term our method as *Category-blind Human Action Recognition Method* (*CHARM*). Given a sequence of 3D joint positions of each person in a video, we first generate possible combinations of mutually inclusive potential actions[2]. We then model potential actions with a category-blind visual representation, which models an action as a set of weighted graphs (one for each action class) with the same topology. In each weighted graph, nodes represent *motion primitives*, and edges represent the co-occurrence of two motion primitives in a particular action class. The weight on an edge represents the co-occurring probability of two motion primitives. The likelihood of a potential action being classified into a particular class is computed by identifying a maximum clique of motion primitives from the corresponding weighted graph. Then CHARM can classify a potential action by identifying the class with the maximum likelihood score. After all potential actions are classified into potential action classes with their associated likelihood scores, CHARM computes the reliability score for each possible combination by averaging the likelihood scores of all involved potential actions. Finally, CHARM outputs the most reliable action combination and identifies the person(s) performing each action. The overall procedure that is more systematic is presented in § 3.

This paper includes the following major contributions: First, we design a category-blind visual representation which allows an action instance to be modeled as a set of weighted graphs which encode the co-occurring probabilities of motion primitives (§ 4.1). Second, such a category-blind visual representation allows the recognition of co-existing actions of different categories (§ 3 and § 4). Third, we design a novel action classification algorithm based on finding maximum cliques of motion primitives on the weighted graphs of the motion primitive co-occurrence patterns (§ 4.2). Finally, we collect a new dataset to evaluate the performance of CHARM in scenarios where actions of different categories co-exist (§ 5).

2. Related Work

Our work is mainly related to two types of human action recognition approaches, each of which is designed to cope with only one action category, *i.e.*, either single-person action or multi-person interaction.

Single-person action recognition. Existing techniques for single-person action recognition are extensively surveyed in [2, 3, 8] with the majority of such methods using monocular RGB videos [9, 20, 21, 26]. Since the advent of cost-effective depth sensors which enable the real-time estimation of 3D skeleton joint positions, many approaches have been developed to extract reliable and discriminative features from skeletal data for action recognition. Vemulapalli *et al*. [18] propose to represent 3D joint positions as elements in a Lie group, *i.e*., a curved manifold, and perform action recognition after mapping the action curves from the Lie group to its Lie algebra. For online action recognition, Zhao *et al*. [25] extract structured streaming skeletons features to represent single-person actions, and use sparse coding technique to do the classification. Wu *et al*. [19] model transition dynamics of an action, and use a hierarchial dynamic framework that first extracts high-level skeletal joints features and then use the learned representation for estimating emission probability to recognize actions.

Multi-person interaction recognition. Kong *et al*. [12, 13] focus on recognizing two-person interaction from 2D videos and propose interactive phrases, high-level descriptions, to express motion relationships between two interacting persons. They propose a hierarchical model to encode interactive phrases based on the latent SVM framework where interactive phrases are treated as latent variables. Yun *et al*. [23] create an interaction dataset containing sequences of 3D joint positions, and extract relevant features, including joint distance and motion, velocity features, *etc*. They use both SVM and MILBoost classifiers for recognition. Also using the sequences of 3D joint positions as input, Alazrai *et al*. [4] design a motion-pose geometric descriptor (MPGD) as a two-person interaction representation. Such a MPGD representation includes a motion profile and a pose profile for each person. These two profiles can be concatenated to form an interaction descriptor for the two interacting persons. The interaction descriptor is then fed into the SVM classifiers for action recognition.

3. Overview

In this section, we describe the overall procedure of CHARM. As shown in Figure 2, the input to CHARM is a sequence of 3D joint positions of human skeletons[3]. Given the input, the goal of CHARM is two-fold, that is, recognizing all actions occurring in this video and identifying the person performing each action. CHARM entails the following steps: (a) CHARM enumerates potential actions for the current sequence, *e.g*., determining the number of persons and the number of pairs of persons. (b) CHARM generates possible combinations of mutually inclusive potential ac-

[2] We define that two potential actions without any common person involved as mutually inclusive.

[3] The human skeletons are tracked by the Microsoft Kinect SDK, so the 3D joint positions for different persons can be distinguished.

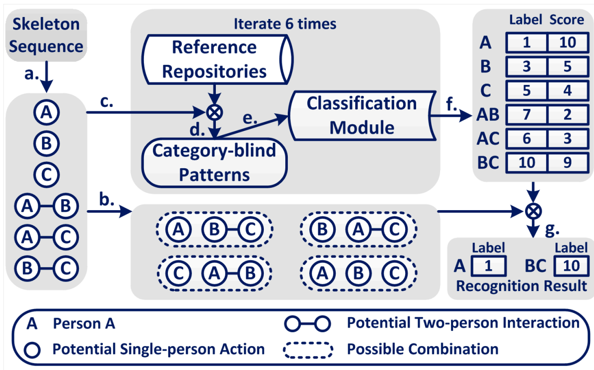

Figure 2. Main steps of CHARM. This example involves 3 persons (*i.e*., Person *A*, *B*, and *C*) and 6 potential actions. The recognition result indicates that *A* is performing a single-person action of Class 1, and *B*&*C* are performing a two-person interaction of Class 10.

tions, *e.g*., with three persons, there are four possible combinations as shown in Figure 2. (c) It extracts relevant body motion data from these potential actions. (d) CHARM extracts category-blind patterns for the current potential action based on the information available in the reference repositories. Such reference repositories are constructed in the training stage. (e) The extracted category-blind patterns are fed to the classification module. (f) The classification module classifies the current potential action and outputs an action label with an associated likelihood score. (g) After all potential actions are classified, CHARM computes a reliability score for each possible combination by averaging the likelihood scores of all involved potential actions, and chooses the most reliable combination as the recognition result.

Steps (c), (d), (e), and (f) are four core steps of the overall procedure of CHARM, and we will describe these four steps in § 4. In particular, in § 4.1.3, we will describe the construction of the reference repositories in the training stage.

4. Methodology

In this section, we describe the four core steps in CHARM, *i.e*., steps (c), (d), (e), and (f). These four steps are divided into two phases: steps (c) and (d) model a potential action using the category-blind visual representation, and steps (e) and (f) classify a modeled potential action into an action class by solving several maximum clique problems (MCP). These two phases are presented in § 4.1 and § 4.2 respectively.

4.1. Modeling a Potential Action Instance using the Category-blind Visual Representation

In CHARM, we model a potential action instance using a category-blind visual representation so that we can directly compare the likelihood scores of any two potential actions which belong to different categories.

We assume that a human action can be represented as a combination of *motion units* (*MUs*). For a single-person action, an MU corresponds to the motion of a single body part (*e.g*., the right upper arm), while for a two-person interaction, an MU corresponds to the motions of a pair of body parts from two interacting persons (*e.g*., a pair of right upper arms). Thus, an action instance of any action category can be modeled using the category-blind visual representation in two steps: (a) First, we model all MUs of an action instance (§ 4.1.1) and then (b) model the combinations of MUs that can match potential action instances (§ 4.1.2).

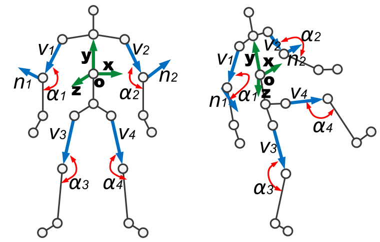

Figure 3. The representation of body part configurations.

4.1.1 MU Model

Let us first consider how to model MUs of a single-person action. Given an input sequence of 3D joint positions, there are many ways to represent body part configurations [3]. In CHARM, we adopt the approach used in [15], which uses eight bilateral symmetrical human body parts, *i.e*., upper arms, lower arms, thighs, and legs. The reason for choosing this body part configuration representation is that it is invariant to the body position, orientation and size, because the person-centric coordinate system is used, and the limbs are normalized to the same length. As shown in Figure 3, all the 3D joint coordinates are transformed from the world coordinate system to a person-centric coordinate system. Upper arms and thighs are attached to the torso at the ball-and-socket joints and move freely in 3D. These four body parts are modeled as four 3D unit vectors v_1, v_2, v_3, and v_4 as shown in Figure 3, and are computed from the coordinates of their endpoints. Lower arms and legs can only bend $0° - 180°$ at the elbow and knee joints. Thus, we model their relative positions with respect to the upper body parts, *e.g*., the upper arms and thighs using four angles α_1, α_2, α_3, and α_4 as shown in Figure 3. These four angles are planar angles because a upper body part and its corresponding lower body part are represented as two intersecting line segments in CHARM. Besides these angles, we also keep track of the planes containing the upper and lower arms which are represented by the unit normals n_1 and n_2 to the planes. We assume the normal direction of the plane formed by legs and thighs remains unchanged with regards to the human centric coordinate system, since the lower leg does not move flexibly with regards to its upper thigh. The four 3D unit vectors $\{v_i\}$, four planar angles $\{\alpha_i\}$, and two 3D normals $\{n_i\}$ form a 22-element body pose vector. Thus, the MUs of a single-person action can be collectively represented as

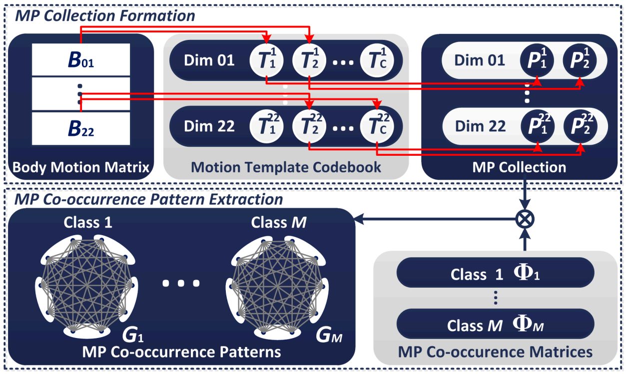

Figure 4. The upper row shows how an MP collection can be formed based on the extracted body motion matrix using the motion template codebook. The lower half is about the extraction of the MP co-occurrence patterns. Red arrows indicate the quantization procedure. The light grey blocks are reference repositories. In the block that shows the MP co-occurrence pattern, each action class should have 22 disjoint groups (each group is illustrated as an ellipse) with each corresponding to a dimension. However, we merely show five disjoint groups to make the diagram clearer. The notation used herein and the rest of the paper are defined in Table 1

a *body motion matrix*, with each row corresponding to a dimension of the body pose vector, and each column corresponding to a particular video frame. In the rest of this paper, each row of the body motion matrix is referred to as a dimension.

A natural choice of modeling MUs of a two-person interaction is to directly use two body motion matrices, one for each person. However, doing so does not capture inter-person temporal correlations, which are very important cues in recognizing two-person interactions. Hence, we augment the MU model with additional inter-person temporal correlations for two-person interactions. We use four types of inter-person temporal correlations, which we will illustrate using the following example. For Persons A and B, we can represent the world coordinates of the origin and coordinate axes of their person-centric coordinates as $\{o_w^A, x_w^A, y_w^A, z_w^A\}$ and $\{o_w^B, x_w^B, y_w^B, z_w^B\}$. The Euclidean distance between A and B can be represented as $d_{AB} = \|o_w^A - o_w^B\|^2$, and the angles between corresponding coordinate axes can be represented as α_x, α_y and α_z. d_{AB}, α_x, α_y, and α_z can form a 4-element inter-person correlation vector. Thus, the inter-person temporal correlation of two persons can be represented as an *inter-person temporal correlation matrix*, with each row corresponding to a dimension of the inter-person correlation vector, and each column corresponding to a video frame.

4.1.2 MU Combination Model

A natural choice for modeling the combination of MUs is to concatenate the representations of all individual MUs. However, due to the complexity of MUs, similar MUs need not be numerically similar. The variations between similar MUs will complicate the training of classifiers with potential overfitting problems. Inspired by [14], we assume an MU can be quantized into several *motion primitives* (*MPs*) which are common patterns for each dimension shared by a variety of human actions.

$\mathcal{B}_i$	The ith dimension of the body motion matrix.
$\mathcal{T}_i^j$	The ith motion template on the jth dimension of the codebook.
$\mathcal{C}$	The number of motion templates on each dimension of the codebook.
$\mathcal{K}$	The number of nearest motion templates that are matched to each $\mathcal{B}_i$.
$\mathcal{N}$	The number of dimensions for an action instance.
$\mathcal{P}_i^j$	The ith MP on the jth dimension.
Φ_i	MP co-occurrence matrix for the ith action class.
$\mathbf{G}_i$	MP co-occurrence pattern of an action instance for class i.

Table 1. Notation.

As such, the task of modeling the combination of MUs for a potential action can be formulated as modeling the combination of MPs of this potential action. We first discuss how to quantize the MUs of a potential action to form a collection of MPs. Using the MU model described in § 4.1.1, the MUs of a single-person action is represented as a body motion matrix, and the MUs of a two-person interaction is represented as two body motion matrices. The formation of MP collection relies on a reference repository, namely *motion template codebook* (see § 4.1.3) which stores a number of motion templates identified from the training action instances. An MP for a single-person action is obtained by finding its best match motion template in the codebook while an MP for a two-person interaction is obtained by selecting the best pair of motion templates in the codebook. The formation of MP collection for both action categories is described as follows:

(a) As shown in the upper half of Figure 4, given the body motion matrix of a potential single-person action, each dimension of the body motion matrix is matched to $\mathcal{K}$ nearest motion templates from the same dimension in the codebook. Each matched motion template becomes an MP for the corresponding dimension. The intuition of generating multiple MPs for each dimension is that the skeletal data collected by the depth camera might be noisy due to some degrading factors, *e.g.*, illumination changes, and the noisy data will impact the quantization from the MUs to MPs; thus, we generate multiple MPs per dimension to increase the tolerance to the quantization error.

(b) The formation of an MP collection for a two-person interaction is similar to the single-person action case. Given two body motion matrices of two interacting persons, we first respectively match each dimension of their body motion matrices to $\mathcal{K}$ nearest motion templates on the same dimension. Then, a pair of matched motion templates (one for each person) from the same dimension compose an MP of a two-person interaction.

MP Co-occurrence Pattern Extraction. Based on the MP collection, we model a potential action by extracting its MP co-occurrence pattern (with each pattern representing a combination of MPs) for each class. To extract such co-occurrence patterns, we rely on a reference repository, namely the *MP co-occurrence matrices* (see § 4.1.3), with each matrix representing the conditional probabilities for any two MPs co-occurring in an action instance of a class.

We define the MP co-occurrence pattern of a potential action for class c as a weighted graph $\mathbf{G}_c = \{\mathbf{V}_c, \mathbf{E}_c, \boldsymbol{\omega}_c\}$, where $\mathbf{V}_c$, $\mathbf{E}_c$, $\boldsymbol{\omega}_c$ represent the set of nodes, edges and edge weights respectively. Each node represents an MP. The nodes in $\mathbf{V}_c$ are divided into 22 disjoint groups (see Figure 4) with each group $\mathbf{R}$ corresponding to a dimension. Thus, $\mathbf{R}_j = \{\mathcal{P}_1^j, \mathcal{P}_2^j, \ldots, \mathcal{P}_{\mathcal{K}^{\mathcal{H}}}^j\}$, where $\mathcal{P}_i^j$ denotes the ith MP of the j-th group. $\mathcal{K}$ is defined in Table 1, and $\mathcal{H}$ is the number of persons involved in this action. $\mathbf{E}_c$ includes the edges of $\mathbf{G}_c$ that connect nodes from different groups, but not within the same group. The edge weights correspond to the pairwise co-occurring probabilities of MPs, and such co-occurring probability is determined by two factors: (a) the conditional co-occurring probabilities of the two MPs and (b) the confidence level of using a particular MP to represent a dimension of the potential action. $\mathbf{G}_c$ is an undirected graph with the symmetrized edge weights calculated by averaging the two weights for different directions:

$$\begin{aligned}\boldsymbol{\omega}_c(\mathcal{P}_i^j, \mathcal{P}_l^m) = \frac{1}{2} \cdot \big(&\Phi'_c(\mathcal{P}_i^j, \mathcal{P}_l^m) \cdot \varphi(\mathcal{P}_l^m) \\ &+ \Phi'_c(\mathcal{P}_l^m, \mathcal{P}_i^j) \cdot \varphi(\mathcal{P}_i^j)\big),\end{aligned} \tag{1}$$

where $\Phi'_c(\mathcal{P}_i^j, \mathcal{P}_l^m) = \Phi_c(\mathcal{P}_i^j, \mathcal{P}_l^m) \cdot (\mathcal{O}_c)^{\frac{1}{2}}$, and $\Phi_c(\mathcal{P}_i^j, \mathcal{P}_l^m)$ is an entry of the MP co-occurrence matrix of class c, indicating the conditional probability of $\mathcal{P}_i^j$ occurring in an action instance of class c given that $\mathcal{P}_l^m$ occurs. $(\mathcal{O}_c)^{\frac{1}{2}}$ is a parameter used to normalize the effects of different co-occurrence matrix sizes on the value of $\Phi_c(\mathcal{P}_i^j, \mathcal{P}_l^m)$, and $\mathcal{O}_c$ is the order [4] of the MP co-occurrence matrix of class c. $\varphi(\mathcal{P}_l^m)$ reflects the confidence level of using $\mathcal{P}_l^m$ to represent a dimension of the potential action:

$$\varphi(\mathcal{P}_l^m) = \exp(-\beta \cdot \frac{1}{\mathcal{H}} \cdot \sum_{h=1}^{\mathcal{H}} \Delta(\mathcal{T}_{l,h}^m, \mathcal{B}_{m,h})), \tag{2}$$

where β is a sensitivity controlling parameter, and $\mathcal{H}$ is the number of persons involved in this potential action. We have $\mathcal{P}_l^m = \{\mathcal{T}_{l,h}^m\}_{h=1}^{\mathcal{H}}$ where $\mathcal{T}_{l,h}^m$ is a matched motion template of the hth person used to generate $\mathcal{P}_l^m$. $\mathcal{B}_{m,h}$ is the mth dimension of the body motion matrix of the hth person. $\Delta(\cdot,\cdot)$ is the dynamic time warping distance [7].

Inter-person Temporal Correlation Pattern Extraction. To augment the MP combination modeling of a potential two-person interaction, we extract its inter-person temporal correlation pattern for each interaction class. Such a temporal correlation pattern is represented as a confidence score, which describes how well the inter-person temporal correlations of this potential interaction instance is aligned with the correlation distribution of a specific interaction class. The confidence score of class c, ℓ_c, is computed as:

$$\ell_c = \prod_{i=1}^{4} \delta(f_{i,c}(\mathcal{I}_i) > \tau_i). \tag{3}$$

$\mathcal{I}_i$ is the ith row of the inter-person temporal correlation matrix (see § 4.1.1). $f_{i,c}$ is a Gaussian distribution which models the distribution of a temporal correlation type for class c. δ is a delta function that takes 1 when the condition is true, and 0 otherwise. τ_i is a threshold. Relevant Gaussian models associated with all interaction classes are stored in a reference repository (see § 4.1.3). Since a single-person action does not have inter-person temporal correlations, we set the values of these confidence scores to 1 for a potential single-person action to create a uniform representation.

4.1.3 Construction of Reference Repositories

Motion template codebook. We start with a training dataset with sequences covering action classes of interest. Each training sequence contains only one action instance, and is associated with an action class label. Each person in the training sequence is represented by twenty 3D joint positions. To construct the motion template codebook, two steps are performed: First, we use the MU model described in § 4.1.1 to represent the MUs of each person in the training sequences as a body motion matrix. Second, we pull all the body motion data from the same dimension together and apply k-means clustering to obtain $\mathcal{C}$ clusters per dimension. Then, we store the centroid of each cluster as a motion template in the codebook. We adopt dynamic time warping distance [7] in the clustering process so that we can cope with training sequences of different lengths.

MP co-occurrence matrices. Given the MUs of each person in the training sequences modeled as a body motion matrix with $\mathcal{N}$ dimensions, we can match each dimension of the body motion matrix to the nearest motion template on the same dimension of the codebook and hence represent an action instance as a collection of $\mathcal{N}$ MPs with each MP corresponding to a dimension. Next, we construct an MP co-occurrence matrix for each action class. We construct the MP co-occurrence matrix for an action class c, by computing the conditional probabilities of any two MPs using the following equation:

$$\Phi_c(\mathcal{P}_i^j, \mathcal{P}_l^m) = p(\mathcal{P}_i^j | \mathcal{P}_l^m, c) = \frac{\Theta_c(\mathcal{P}_i^j, \mathcal{P}_l^m)}{\Theta_c(\mathcal{P}_l^m)}, \tag{4}$$

where $\Theta_c(\mathcal{P}_i^j)$ is the number of training action instances of class c containing MP $\mathcal{P}_i^j$ and $\Theta_c(\mathcal{P}_i^j, \mathcal{P}_l^m)$ is the number of action instances of class c where $\mathcal{P}_i^j$ and $\mathcal{P}_l^m$ co-occur. If the denominator $\Theta_c(\mathcal{P}_l^m)$ equals zero, then $\Phi_c(\mathcal{P}_i^j, \mathcal{P}_l^m)$ is directly set as zero. However, if $\mathcal{P}_i^j$ and $\mathcal{P}_l^m$ are MPs corresponding to the same dimension, *i.e.*, $j = m$, $\Phi_c(\mathcal{P}_i^j, \mathcal{P}_l^m)$ is set to zero, *i.e.*, we do not allow them to co-occur since

[4] It can be easily seen that if class c is a single-person action class, $\mathcal{O}_c = \mathcal{N} \cdot \mathcal{C}$; if class c is a two-person interaction class, $\mathcal{O}_c = \mathcal{N} \cdot \mathcal{C}^2$

we only allow one MP for each dimension for any action instance. As shown in (4), the element $\Phi_c(\mathcal{P}_i^j, \mathcal{P}_l^m)$ is equivalent to the conditional probability $p(\mathcal{P}_i^j|\mathcal{P}_l^m, c)$ which is the probability of $\mathcal{P}_i^j$ occurring in an action instance of class c given that $\mathcal{P}_l^m$ occurs. The advantages of defining a co-occurrence matrix using the conditional probability are: (a) It enhances the robustness of CHARM such that it can tolerate action variations caused by personal-styles more because the conditional probability does not penalize those co-occurrences of MPs that happen less frequently. (b) The resulting co-occurrence matrix is asymmetric; such asymmetry property is helpful for coping with intra-class variations because it ensures that the probability of $\mathcal{P}_i^j$ occurring in an action instance given that $\mathcal{P}_l^m$ occurs is not necessarily equivalent to that of the reverse situation.

Gaussian models with respect to each interaction class. There are four Gaussian models for each interaction class, with each modeling the distribution of an inter-person temporal correlation type for this interaction class. The mean and standard deviation of each Gaussian model are computed using the relevant data for that correlation type from all training instances for that interaction class.

4.2. Action Classification using MCP

The category-blind representation of a potential action includes a set of class-specific patterns. We compute a likelihood score for each class-specific pattern and then choose the class label associated with the highest score as the label for that potential action. Recall that each class-specific pattern can be represented as a weighted graph, *e.g.*, $\mathbf{G}_c = \{\mathbf{V}_c, \mathbf{E}_c, \boldsymbol{\omega}_c\}$ for class c. Since in the MP co-occurrence pattern extraction process, we include multiple MPs per dimension, for each class c, we thus first need to identify a subgraph, $G_s = \{V_s, E_s, \omega_s\}$, from $\mathbf{G}_c$, which includes only one MP per dimension. A feasible G_s has to satisfy the following three constraints: (a) Given $\mathcal{N}$ disjoint groups (dimensions), one and only one node from each group should be selected. (b) If one node is selected to be in G_s, then exactly $(\mathcal{N}-1)$ of its edges should be included in G_s (this is because each selected node should be connected to one node at each of the rest $(\mathcal{N}-1)$ groups). (c) If an edge is included in G_s, the nodes incident to it should be also included and vice versa.

The metric that we use to identify a G_s from $\mathbf{G}_c$ is the co-occurring likelihood of the clique of MPs in G_s. Such a co-occurring likelihood is defined as follows:

$$\lambda(G_s) = \sum_{p=1}^{\mathcal{N}} \sum_{q=1, q\neq p}^{\mathcal{N}} \boldsymbol{\omega}_c(V_s(p), V_s(q)). \tag{5}$$

where $V_s(p)$ denotes the node within the pth group of G_s. Thus, we formulate the identification process of a subgraph G_s^* which contains the MP clique that is most likely to occur, as the following optimization problem:

$$G_s^* = \underset{V_s}{\operatorname{argmax}}\, \lambda(G_s). \tag{6}$$

Once we have such G_s^* for each action class, we compute a likelihood score, Υ_c, which measures how likely we should classify this potential action to a particular action class c:

$$\Upsilon_c = \ell_c \cdot \lambda(G_{s,c}^*), \tag{7}$$

where ℓ_c is defined in (3). Eventually, we classify this potential action to the class which yields the highest likelihood score. Such a classification task is formulated as follows:

$$c^* = \arg\max_c(\Upsilon_c). \tag{8}$$

The optimization problem in (6) is a maximum clique problem (MCP) that is NP-hard. Several approximation algorithms exist for solving MCP, *e.g.*, [5, 6, 24]. In this work, we adopt a mixed binary integer programming (MBIP) based solver [6]. We set the objective function of the MBIP as the optimization problem of (6), and set the constraints of the MBIP as the formulation of three aforementioned feasibility constraints. The MBIP is solved by the Cplex [1].

5. Experimental Evaluations

For evaluation, we use three test scenarios: (i) videos with three persons and a mixture of single-person actions and two-person interactions, (ii) videos with two interacting persons, and (iii) videos with a single person. The test scenario (i) is closer to real-world scenarios where the category of a video sequence is not given, and multiple action instances belonging to different categories co-exist in a video sequence. This scenario is used to highlight the advantage of our CHARM approach where no prior category information is given. In contrast, each video sequence in test scenario (ii) and (iii) contains only one action instance at one time and its action category (either single-person action or two-person interaction) is predefined.

Since no prior existing datasets include test scenario (i), we collect our own ”Hybrid Actions3D” dataset[5]. In addition, we adopt SBU Interaction dataset [23] for test scenario (ii), and MSRC12-Gesture dataset [11] for (iii).

Baselines. Since no prior work is designed to handle test scenario (i), we create a baseline scheme called ”SimpleComb” which combines two dedicated approaches: one is used to recognize single-person actions and the other one is used to recognize two-person interactions. For the single-person action recognition, we use the approach described in [19]. Since there is no publicly available code for previous two-person interaction recognition approaches, we use our CHARM by limiting it to only deal with two-person interactions. Each dedicated approach will label a potential action with a particular action class associated with a likelihood score. Before we decide which mutually inclusive action combination (see § 3) to be the recognition result, we

[5]The dataset is available at: http://www.lehigh.edu/ wel514/main.html, and www.cbsr.ia.ac.cn/users/lywen.

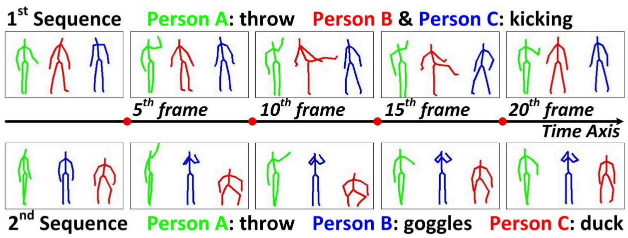

Figure 5. Sample frames from two test sequences of the Hybrid Actions3D dataset: The first row presents a hybrid-category test sequence involving two action instances, *i.e.*, Person *A* performing the single-person action "throw", while Person *B* and *C* are performing the two-person "kick" interaction. The second row presents a mono-category test sequence consisting of three single-person action instances, *i.e.*, Person *A* performing the "throw", *B* performing "goggles", and *C* performing "duck" action.

first scale the likelihood scores of both approaches so that their values are comparable. Then, we choose the combination which yields the highest reliability score computed as described in step (g) of CHARM (see § 3). For test scenario (ii), we compare CHARM with two previous approaches (based on Linear-SVM, and MILBoost respectively) described in [23]. For test scenario (iii), we compare CHARM with four state-of-the-art approaches [11, 25, 16, 19].

Parameter Settings. We use the same motion template codebook for both single-person action recognition and two-person interaction recognition. We set the default values for the parameters of CHARM as follows: The number of motion dimension of this codebook is $\mathcal{N} = 22$, and the number of motion templates per dimension is $\mathcal{C} = 22$. In the recognition phase, when we quantize the body motion data into MPs, we match each body motion data with $\mathcal{K} = 2$ nearest motion templates. The sensitivity controlling parameter in (2) is $\beta = 0.1$. The thresholds for four types of temporal correlation in (3) are: $\tau_d = 2 \times 10^{-5}$, $\tau_x = 10^{-12}$, $\tau_y = 0.1$, and $\tau_z = 10^{-12}$.

5.1. Action Recognition using Hybrid Actions3D

Datasets. Hybrid Actions3D is captured using the Kinect camera [17], which tracks human skeletons and estimates the 3D joint positions for each person at each frame. This dataset includes 10 action classes, 5 of which are single-person action classes (*i.e.*, *duck*, *goggles*, *shoot*, *beat both*, and *throw*), and the remaining ones are two-person interaction classes (*i.e.*, *kick*, *push*, *punch*, *hug*, and *exchange*). The single-person action classes are from the MSRC12-Gesture [11], and the two-person interaction classes are from the SBU Interaction [23]. 10 volunteers were recruited to create this dataset. We first ask each volunteer to perform single-person actions. Next, we formed 14 pairs out of these 10 volunteers and ask them to perform two-person interactions. In total, this dataset contains 910 pre-segmented video sequences including 580 for training and 330 for testing. Specifically, there are 280 two-person interaction training sequences (56 sequences per action class) and 300 single-person action training sequences (60 sequences per action class). The 330 test sequences consist of (a) 280 hybrid-category sequences (constructed by fusing relevant sequences), each contains a two-person interaction instance and a single-person action instance, and (b) 50 mono-category sequences containing three single-person action instances. The action instances in a test sequence are not required to have the same starting and ending time points. Some sample frames of the test sequences in the Hybrid Actions3D dataset are shown in Figure 5.

Method	Action-level Accur.		Seq-level Accur.
	Single-person Act.	Two-person Interact.	
SimpleComb	0.909	0.746	0.739
CHARM	0.921	0.811	0.800

Table 2. Comparison on Hybrid Actions3D (test scenario (i)).

Method	Accuracy
Joint Features + Linear SVM [23]	0.687
Joint Features + MILBoost [23]	0.873
CHARM	0.839

Table 3. Comparison on SBU Interaction (test scenario (ii)).

Evaluation metrics. Two metrics are used to evaluate the recognition capability of CHARM, namely (a) the sequence-level accuracy, and (b) the action-level accuracy. Since each test sequence in our Hybrid Action3D dataset contain more than one action instance, for sequence-level accuracy, we consider a recognition result as accurate only if all action instances are classified correctly. On the other hand, every action instance that is correctly classified is counted towards the action-level accuracy.

Table 2 shows the results for the test scenario (i). It shows that CHARM yields better performance than SimpleComb in terms of both sequence-level accuracy and action-level accuracy. We also observe that although SimpleComb uses the CHARM as a dedicated approach to recognize two-person interactions, its two-person interaction accuracy is lower than that of our CHARM. We believe that such a performance gap is caused by the difficulties in merging seamlessly the results from the two dedicated approaches such that the benefits of each dedicated approach cannot be fully utilized. We notice that the overall performance of CHARM for single-person actions is better than the performance for two-person interactions. In general, CHARM works well for most actions but tends to have problem classifying similar two-person interactions, *e.g.*, "push" and "hug". In CHARM, we did some tradeoffs between the expressiveness capability of our visual representation model towards any specific action category and its capability for a uniform visual representation, *e.g.*, compared to existing approaches (*e.g.*, [23]), we only use a very simple model in CHARM to capture the inter-person temporal correlations.

5.2. Action Recognition using SBU Interaction

Datasets. SBU Interaction dataset consists of approximately 300 pre-segmented two-person interaction sequences in the form of 3D joint positions. This dataset includes eight classes: *approach*, *depart*, *push*, *kick*, *punch*, *exchange objects*, *hug*, and *shake hands*. As in [23], we use the sequence-level accuracy as our evaluation metric.

We compare our CHARM with the two interaction recognition approaches in [23] using five-fold cross validation. As shown in Table 3, the only approach that slightly outperforms CHARM is the MILBoost based approach [23], which uses spatio-temporal distances between all pairs of joints of two persons as feature. However, the MILBoost based approach [23] focuses on recognizing two-person interactions (test scenario (ii)), and cannot be used to cope with the test scenario (i) and (iii).

5.3. Action Recognition using MSRC12-Gesture

Datasets. The MSRC12-Gesture dataset was collected by having volunteers perform certain actions either based on "Video + Text" based instructions or based on "Image + Text" instructions. We refer to these different procedures as different modality. This dataset is chosen to validate the effectiveness of CHARM to handle the streaming data sequences. It contains 594 sequences collected from 30 people performing 12 different single-person actions, including *lift outstretched arms*, *duck*, *push right*, *goggles*, *wind it up*, *shoot*, *bow*, *throw*, *had enough*, *change weapon*, *beat both*, and *kick*. Each sequence may contain multiple action instances, thus there is a total of 6244 action instances in this dataset. All sequences in this dataset are non-segmented, *i.e.*, there do not exist information about where the starting and ending times of an action instance are within a sequence. We only know the ending points of all action instances since they are manually labeled by the authors who release this dataset. The authors indicate that any recognition system which can correctly identify the ending time of an action instance within $\pm\xi = 10$ video frames should be considered as accurately identify this action instance.

To use CHARM on a non-segmented streaming data sequence, as in [11], we use a 35-frame sliding window to continuously segment potential action instances from the streaming data. In addition, inspired by [22], we introduce a background class and use a threshold for each class such that we can balance the precision and recall rates of our CHARM-based recognition system. Specifically, we redefine (8) as $\epsilon^* = \arg\max_\epsilon(\Upsilon_\epsilon)$, s.t. $\Upsilon_\epsilon > \theta_\epsilon$. As in [11, 22, 25], the optimal θ_ϵ is chosen such that it minimizes the recognition error, *e.g.*, we set the threshold θ_b for the background class to be zero. We conduct our experiments using the same test procedure described in [19] where training and test sequences can potentially come from different modality. We did two groups of experiments, namely "intra-modality" and "inter-modality". "Intra-modality" indicates that training and test sequences are from the same modality, *e.g.*, both collected using "Video+Text" instructions. "Inter-modality" indicates that training and test sequences are collected using different modality. It is clear from our results that sequences using the "Image+Text" instructions tend to have more variations and hence lower recognition accuracy.

Method	intra-modality	inter-modality
Randomized Forest [11]	0.621	0.576
Structured Streaming Skeletons [25]	0.718	N\A
Multi-scale Action Detection [16]	0.685	N\A
DBN-ES-HMM [19]	0.724	0.710
CHARM	0.725	0.700

Table 4. Comparison on MSRC12-Gesture (test scenario (iii)). "N\A" indicates that experimental results of the corresponding approaches are not available mainly because the authors neither provide the results in their paper nor publish their code.

As in [19], we use the criteria, F-score which combines the precision and recall to evaluate the performance of different action recognition methods. Table 4 shows that CHARM performs better than all baseline methods for single-person action recognition for the intra-modality scenario. For the inter-modality scenario, CHARM performs better than [11] and yields comparable performance to [19]. Methods in [25] and [16] are not compared because the authors neither publish their performance for the inter-modality scenario nor their code.

6. Conclusion

We presented a category-blind human action recognition method (CHARM) which is more suitable for real-world scenarios. Compared to existing action recognition approaches that are designed to cope with only one action category, and are difficult to be extended to scenarios where different action categories co-exist, CHARM achieves comparable or better performance without any prior action categorization. In CHARM, action instances of different action categories are all modeled as a set of weighted graphs which encode the co-occurring probabilities of motion primitives. Such a category-blind representation makes it possible for CHARM to simultaneously recognize actions of different categories which co-exist in any video sequence. The action classification is performed via finding maximum cliques of motion primitives on the weighted graphs.

The future work will focus on (a) enhancing our temporal inter-person correlation model, and (b) applying the CHARM to more complex action recognition in real life.

7. Acknowledgements

We would like to thank the students of Lehigh University for helping collect the dataset. W. Li and M. Chuah are supported via Lehigh fellowship and US National Science Foundation Research Grant (CSR-1016296 and CSR-1217379). L. Wen and S. Lyu are supported via US National Science Foundation Research Grant (CCF-1319800) and National Science Foundation Early Faculty Career Development (CAREER) Award (IIS-0953373).

References

[1] Ibm ilog cplex optimizer. http://www-01.ibm.com/software/commerce/optimization/cplex-optimizer/.

[2] J. K. Aggarwal and M. S. Ryoo. Human activity analysis: A review. *ACM Computing Surveys*, 43(3):16, 2011.

[3] J. K. Aggarwal and L. Xia. Human activity recognition from 3d data: A review. *Pattern Recognition Letters*, 48:70–80, 2014.

[4] R. Alazrai, Y. Mowafi, and G. Lee. Anatomical-plane-based representation for human-human interactions analysis. In *Pattern Recognition*, 2015.

[5] E. Althaus, O. Kohlbacher, H. Lenhof, and P. Müller. A combinatorial approach to protein docking with flexible side-chains. In *RECOMB*, pages 15–24, 2000.

[6] S. M. Assari, A. R. Zamir, and M. Shah. Video classification using semantic concept co-occurrences. In *CVPR*, 2014.

[7] D. J. Berndt and J. Clifford. Using dynamic time warping to find patterns in time series. In *KDD Workshop*, 1994.

[8] J. M. Chaquet, E. J. Carmona, and A. Fernández-Caballero. A survey of video datasets for human action and activity recognition. *CVIU*, 117(6):633–659, 2013.

[9] C. Chen and K. Grauman. Efficient activity detection with max-subgraph search. In *CVPR*, 2012.

[10] A. Eweiwi, M. S. Cheema, C. Bauckhage, and J. Gall. Efficient pose-based action recognition. In *ACCV*, 2014.

[11] S. Fothergill, H. M. Mentis, P. Kohli, and S. Nowozin. Instructing people for training gestural interactive systems. In *CHI*, 2012.

[12] Y. Kong, Y. Jia, and Y. Fu. Learning human interaction by interactive phrases. In *ECCV*, 2012.

[13] Y. Kong, Y. Jia, and Y. Fu. Interactive phrases: Semantic descriptions for human interaction recognition. In *PAMI*, pages 1775–1788, 2014.

[14] J. Liu, B. Kuipers, and S. Savarese. Recognizing human actions by attributes. In *CVPR*, 2011.

[15] S. M. S. Nirjon, C. Greenwood, C. Torres, S. Zhou, J. A. Stankovic, H. Yoon, H. Ra, C. Basaran, T. Park, and S. H. Son. Kintense: A robust, accurate, real-time and evolving system for detecting aggressive actions from streaming 3d skeleton data. In *PerCom*, 2014.

[16] A. Sharaf, M. Torki, M. E. Hussein, and M. El-Saban. Real-time multi-scale action detection from 3d skeleton data. In *WACV*, 2015.

[17] J. Shotton, A. W. Fitzgibbon, M. Cook, T. Sharp, M. Finocchio, R. Moore, A. Kipman, and A. Blake. Real-time human pose recognition in parts from single depth images. In *CVPR*, 2011.

[18] R. Vemulapalli, F. Arrate, and R. Chellappa. Human action recognition by representing 3d skeletons as points in a lie group. In *CVPR*, 2014.

[19] D. Wu and L. Shao. Leveraging hierarchical parametric networks for skeletal joints based action segmentation and recognition. In *CVPR*, 2014.

[20] G. Yu, J. Yuan, and Z. Liu. Unsupervised random forest indexing for fast action search. In *CVPR*, 2011.

[21] G. Yu, J. Yuan, and Z. Liu. Propagative hough voting for human activity recognition. In *ECCV*, 2012.

[22] G. Yu, J. Yuan, and Z. Liu. Discriminative orderlet mining for real-time recognition of human-object interaction. In *ACCV*, 2014.

[23] K. Yun, J. Honorio, D. Chattopadhyay, T. L. Berg, and D. Samaras. Two-person interaction detection using body-pose features and multiple instance learning. In *CVPR Workshop*, 2012.

[24] A. R. Zamir, A. Dehghan, and M. Shah. Gmcp-tracker: Global multi-object tracking using generalized minimum clique graphs. In *ECCV*, 2012.

[25] X. Zhao, X. Li, C. Pang, X. Zhu, and Q. Z. Sheng. Online human gesture recognition from motion data streams. In *ACM International Conference on Multimedia*, 2013.

[26] B. Zhou, X. Wang, and X. Tang. Understanding collective crowd behaviors: Learning a mixture model of dynamic pedestrian-agents. In *CVPR*, 2012.