Jointly learning heterogeneous features for RGB-D activity recognition
Hu, Jian-Fang; Zheng, Wei-Shi; Lai, Jianhuang; Zhang, Jianguo

Published in:
IEEE Transactions on Pattern Analysis and Machine Intelligence

DOI:
10.1109/TPAMI.2016.2640292

Publication date:
2017

Document Version
Peer reviewed version

Link to publication in Discovery Research Portal

Citation for published version (APA):
Jointly Learning Heterogeneous Features for RGB-D Activity Recognition

Jian-Fang Hu, Wei-Shi Zheng, Jianhuang Lai, and Jianguo Zhang

Abstract—In this paper, we focus on heterogeneous features learning for RGB-D activity recognition. We find that features from different channels (RGB, depth) could share some similar hidden structures, and then propose a joint learning model to simultaneously explore the shared and feature-specific components as an instance of heterogeneous multi-task learning. The proposed model formed in a unified framework is capable of: 1) jointly mining a set of subspaces with the same dimensionality to exploit latent shared features across different feature channels, 2) meanwhile, quantifying the shared and feature-specific components of features in the subspaces, and 3) transferring feature-specific intermediate transforms (i-transforms) for learning fusion of heterogeneous features across datasets. To efficiently train the joint model, a three-step iterative optimization algorithm is proposed, followed by a simple inference model. Extensive experimental results on four activity datasets have demonstrated the efficacy of the proposed method. A new RGB-D activity dataset focusing on human-object interaction is further contributed, which presents more challenges for RGB-D activity benchmarking.

Index Terms—heterogeneous features learning, RGB-D activity recognition, action recognition

1 INTRODUCTION

The emergence of low-cost depth sensors (e.g., the Microsoft Kinect) opens a new dimension to address the challenges of human activity recognition. Compared to the conventional use of RGB videos, the information from depth channel is insensitive to illumination variations, invariant to color and texture changes, and more importantly reliable for body silhouette and skeleton (human posture) extraction [29]. Bearing on these merits, it is believed that the introduced depth information can greatly light up the research of human activity analysis [10], [23], [34], [48].

Nevertheless, using depth alone has limitations in distinguishing human activities and object context in challenging cases [37], [53]. Depth information (e.g. captured by existing Kinect device) often suffers from low spatial resolution and low depth precision. Moreover, the depth information is usually weak in capturing the important appearance information, such as object color and texture. These greatly limit the application of depth cameras on recognizing complex human activities with object and interactions, such as human-object interactions [10] and fine grained activities [15], where the color appearance is also important.

In fact, there indeed exists a connection between the information from RGB and depth channel, which could be unveiled after certain transformation. In Figure 1 we show some visualization results of the HOG features extracted from RGB image patches and the corresponding depth patches. Albeit extracted from different channels, these HOG features still look similar for each of the activities. This suggests that depth channel is related to the RGB channels and the heterogeneous features extracted from different channels could share some (hidden) structures (e.g., the gist of looking down in reading, and cup-to-mouth in drinking).

Nevertheless extracted from different feature channels; for instance, the RGB channel mainly captures the appearance (color) information, while the depth channel describes the geometry (shape) cues in depth. This suggests they indeed have their own characteristics in describing objects. Hence, learning RGB and depth features together should not only extract shared features that are robust and collaborative across feature channels but also exploit features complementary between different channels. However, the majority of existing RGB-D action recognition methods [5], [19], [28] neither seek to jointly learn the features extracted from RGB and depth channels simultaneously nor model their underlying connections.

In order to effectively capture the connections among different heterogeneous features, we propose a joint heterogeneous feature
In summary, the main contributions of our work are threefold: 1) a novel joint heterogeneous feature learning framework for RGB-D activity recognition, which is capable of learning hidden connections among heterogeneous features extracted from sequences of different channels; 2) a transfer RGB-D feature learning framework leveraging auxiliary datasets; 3) a new dataset collected for RGB-D human-object interaction recognition.

2 Related Work

Recently, recognizing human activities from low cost depth cameras has become a more and more important research direction with many applications including digital surveillance, virtual reality, human-computer interaction and Xbox One games etc. There are two emerging branches in activity recognition research: 1) depth-based representation, and 2) RGB-D based development. In this section, in addition to reviewing existing works of recognizing human activities captured by depth cameras, we further briefly describe the literature of learning heterogeneous features for generic visual recognition purpose, which is also relevant to ours.

**Depth-based representation.** On building depth-based representation, a straightforward way is to generalize the descriptors specially designed for RGB channel to depth channel for describing the shape geometry \([17], [50]\). For instance, Oreifej and Liu \([27]\) extended the histogram of gradient (HOG) descriptor by constructing a histogram to capture the distribution of surface normal orientation in 4D space. Yang et al. \([46]\) suggested that concatenating the normal vectors within a spatiotemporal depth sub-volume together can capture more informative geometric cues. \([36]\) sought to explicitly encode the geometric cues by computing the number of points that follow in each sampled sub-volume. Lu et al. \([29]\) directly investigated the relationship between sampled pixels in both actor and background regions. Most of these methods attempted to mine some discriminative local patterns for representing human activities without considering the holistic human poses, which has been demonstrated to be critical for describing complex human activities involving human-object interactions \([10], [59], [48]\). Due to the development of realtime human skeleton (3D posture) tracker from single depth image \([29]\), human motions can be effectively captured by the
RGB-D based development. Depth does not necessarily mean discriminant. Albeit invariant to lighting changes, it does lose some useful information such as texture context, which is critical to distinguish some activities involving human-object interactions. Recent works also showed that the fusion of the RGB and depth sequences can largely improve the recognition of activities with object interactions [5], [14], [15], [19], [28], [39], [49], [53]. For instance, Zhao et al. [53] combined interest point based descriptors extracted from RGB and depth sequences together for recognition. Liu and Shao [19] simultaneously fused the RGB and depth information using a deep architecture; Zhu et al. [55] employed a set of random forests to fuse spatiotemporal and human key joints (skeleton); Shahroudy et al. [28] selected to fuse the RGB information and skeleton cues using a structured sparsity method; [5] simply concatenated the skeleton features and silhouette-based features together for classification. However, these existing works treated the depth channel and RGB channel independently without considering their underlying connections (structures). Thus their recognition performance would often be hindered by the ignored structure learning. In this context, our model aims to jointly learn the hidden shared and specific structures among different heterogeneous features extracted from depth and color sensors, respectively. This leads to a better overall performance in the RGB-D activity recognition.

Shared-specific structures learning for activity recognition. Learning shared-specific structures for activity recognition is found to be beneficial. Shared-specific structures are defined and mined from different perspectives and for different purposes in the literature [8], [30], [35], [38], [43], [54]. For example, some researchers intended to exploit their discriminative shared-specific features by constructing shared and class-specific dictionaries [8], [35] or learning local motion patterns that are shared by different activities [54]; and recently, this idea was also introduced for recognizing activities captured from different views in [30], [43]. However, these methods assume that they can directly align different feature channels or extract shared and specific information without any pre-learning. Our proposed model differs from them significantly, since an i-transform is introduced for each feature channel in order to make the shared-specific structures learning be performed in a more suitable latent space. And this is highly demanded when processing heterogeneous features with different number of dimensions. Although the CCA in [3] is mostly close to ours, it is not for discriminative learning, and moreover it assumes that the specific component for each feature channel is a Gaussian distribution (or Gaussian noise) and this assumption may not hold and thus not be sufficient to describe the specific information of each channel. Our experimental results show that our JOULE model performs better than an advanced variant of CCA (MPCCA) in [3].
2) a new transfer learning based joint learning model was proposed by employing an auxiliary set to facilitate the feature learning on the target set; and 3) we have provided a rigorous and theoretical analysis about the convergence of the developed three-step optimization method in the supplementary file; and 4) we conducted a new group of experiments and added more comparisons on one additional dataset consisting of a set of complex composed activities [18]. 5) we have added extensive test and deeper analysis, including the comparison with additional methods (e.g., the heterogeneous feature machine [4]), the effect of regularization parameters, the influence of the newly introduced control parameter and the evaluation of Transfer-JOULE.

3 Heterogeneous Features Construction

We describe here in detail three descriptors utilized in our model: dynamic skeleton (DS) features, dynamic color pattern (DCP) and dynamic depth pattern (DDP). Each descriptor consists of two components: temporal pyramid Fourier features (TPF) from: i) the original feature signal and ii) the corresponding gradient signal, respectively. These six components form our heterogeneous feature set.

The use of TPF features is motivated from the work of Wang et al. [37]. Following their practice, we repeatedly partition the feature signal (e.g., temporal skeleton features in [37]) into 1, 2 and 4 sub-segments along the temporal dimension, and then concatenate the low frequency Fourier coefficients extracted from each segment.

In addition to computing TPF from the original feature series as in [37], we also calculate TPF from the temporal gradient signal of the original feature series. This proposed extension is motivated from the following observations: 1) the gradient could, to a certain extent, implicitly encode the velocity change of the motion in activity; 2) it could also capture the variation of pixel values, which helps to describe the interactions between human and objects. For instance, the rapid change of the pixel values near a mouth may indicate that some objects are coming near and interacting with the mouth (e.g., drinking). As illustrated in Figure 3 the temporal pyramid Fourier features of the gradient signal may capture more discriminative cues.

Dynamic Skeleton. Human pose and its dynamics are one of the key elements in activities [10, 27]. Here we extract the pose dynamics using skeleton information from the depth sequences for our activity modeling. Specifically, for each video sequence, the real-time skeleton tracker [29] is used to extract the trajectories of human key joints (skeleton). Following the implementations in [37], we then compute the relative positions between each pair of trajectories and concatenate them together. The temporal pyramid Fourier features are further extracted from the relative positions as well as its gradient version to represent the dynamic pose information. It was noted that the sequence length may vary from video to video. Relative positions of each trajectory pair are interpolated by cubic spine to have the same length before computing the Fourier coefficients, which ensures that the frequency locations of computed TPF features are properly calibrated and aligned before comparison.

Dynamic Color and Depth Pattern. Using the 3D joint positions without local appearance is often insufficient to characterize complex activities including human-object interactions. To compensate this, the local appearance features (both in RGB and depth) are extracted around each human joint, which could capture characteristic shape, texture and manipulated object’s appearance during interactions. Specifically, for each joint in a trajectory, we first compute the HOG feature in its local region for all the associated frames. All of the HOG features of one joint trajectory constitute a temporal HOG tube. Then for the trajectory of each bin of the vectorized HOG feature along the time dimension, we extract the TPF features including the original and gradient version, and then concatenate them together to form our original descriptor. The HOG-TPF extracted from RGB sequence and depth sequence form our dynamic color pattern (DCP) and dynamic depth pattern (DDP), respectively.

4 Heterogeneous Feature Learning

Different features may share some similar structural components as illustrated in Figure 1. To effectively quantify the shared structures among different features with varied dimensions, we introduce a set of subspaces to represent these features so that they can be compared directly. These subspaces are learned by balancing the trade-off between the shared structures and feature-specific cues. In the following, we define our notations first, and then present a detailed description of the proposed joint learning model.

4.1 The Joint Learning Model

Suppose there are S types of heterogeneous features to learn together. For each feature type i (i = 1, ..., S), let \( X_i \in \mathbb{R}^{d_i \times n_i} \) denote the feature matrix of \( n_i \) training instances, where \( d_i \) represents the feature dimensionality. We attempt to learn a projection matrix \( \Theta_i \) for each \( X_i \) to project it into a subspace spanned by the columns of \( \Theta_i \). Here for simplicity and clarity, we call this projection matrix \( \Theta_i \) as intermediate transform (i-transform).

In total, we have \( S \) subspaces, which are set to have the same dimensionality such that both the shared and feature-specific structures across different feature types can be easily quantified in the projected feature space by two weight matrices \( \mathbf{W}_0, \mathbf{W}_i \in \mathbb{R}^{M \times L} \), where \( M \) is the dimensionality of the subspace, and usually \( M < < d_i \). \( L \) indicates the number of activity classes. We use \( Y_i \in \{-1, L - 1\}^{L \times n_i} \) to represent the labels of all the training samples for the \( i \)-th feature. Each column of \( Y_i \) is defined as a zero-mean vector \([-1, ..., -1, L - 1, -1, ..., -1]\). Note that all of the \( Y_i \)'s are label vectors and they are the same for different types of features. For a sample with class label \( l \) (\( l = 1, ..., L \)), the \( l \)-th entry of the zero-mean vector equals to a constant positive number \( L - 1 \).

Now, we formulate our joint heterogeneous features learning (JOULE) model in a multi-task learning framework with orthogonality constraints considered as follows:

\[
\begin{align*}
\min_{\mathbf{W}_0, \{\mathbf{W}_i\}, \{\Theta_i\}} & \sum_{i=1}^{S} \left( \frac{R_1(\mathbf{W}_0, \{\mathbf{W}_i\}, \{\Theta_i\})}{\mathbb{E}(\mathbf{W}_0, \{\mathbf{W}_i\}, \{\Theta_i\})} \right) \\
R_1(\mathbf{W}_0, \{\mathbf{W}_i\}, \{\Theta_i\}) & = \sum_{i=1}^{S} \left( (\lambda \mathbf{W}_0 + (1 - \lambda) \mathbf{W}_i)^T \Theta_i^T \mathbf{X}_i - \mathbf{Y}_i \right)^2 \\
R_2(\Theta_i) & = \| \mathbf{X}_i - \Theta_i \|^2_F + \frac{\alpha}{S} \| \mathbf{W}_0 \|^2_F + \beta \| \mathbf{W}_i \|^2_F \\
s.t. \quad & \Theta_i^T \Theta_i = \mathbf{I}, i = 1, 2, ..., S
\end{align*}
\]

Our heterogeneous feature learning model intends to jointly learn the subspaces (encoded by i-transform \( \{\Theta_i\} \)), shared and feature-specific components (represented by \( \mathbf{W}_0 \) and \( \{\mathbf{W}_i\} \), respectively) in a unified framework. We cast it as a least-square problem with both prediction (the first term...
\( R_1(\mathbf{W}_0, \{\mathbf{W}_i\}, \{\Theta_i\}) \) and reconstruction loss (the second term \( R_2(\{\Theta_i\}) \)) as well as the regularization term \( R_3(\mathbf{W}_0, \{\mathbf{W}_i\}) \) considered together. In the following, we discuss these terms in detail one by one.

**Prediction loss term** \( R_1(\mathbf{W}_0, \{\mathbf{W}_i\}, \{\Theta_i\}) \). This term is defined as \( \|((\lambda \mathbf{W}_0 + (1 - \lambda) \mathbf{W}_i)^T \Theta_i^T \mathbf{X}_i - \mathbf{Y}_i\|_F^2 \) such that the empirical risk of each feature can be minimized, and thus it would guide our shared-specific structures learning for the purpose of better recognition. We formulate the prediction loss term in the multi-task learning framework in order to jointly learn the shared and specific structures across different features and classes together. Here, we model the structures in the weight space such that the shared-specific structures can be mined in a discriminative framework. Specifically, we use a weight matrix \( \mathbf{W}_0 \) that is owned jointly by different features to encode the shared structures. We also employ a matrix \( \mathbf{W}_i \) only privately possessed by the \( i^{th} \) feature to capture its specific component. Discovering the shared and specific structures in a joint learning model is essential for connecting and transferring information among different tasks. Therefore our method could generalize well to the case of knowledge transfer from some auxiliary data to facilitate the model learning, which will be further elaborated in Section 4.2. Here, we utilize parameter \( \lambda \in [0, 1] \) to control the tradeoff between the shared and specific structures. Larger \( \lambda \) leads to a larger weight on the shared structure and smaller weight on the specific structures.

**Reconstruction loss term** \( R_2(\{\Theta_i\}) \). This term is defined as the reconstruction loss term to ensure that a good reconstruction (controlled by the parameter \( \gamma \)) can be derived from the learned subspace using i-transform during optimization, which leads to a meaningful solution of the model.

To facilitate the formulation of reconstruction loss term, an orthogonal constraint \( \Theta_i^T \Theta_i = \mathbf{I} \) was imposed on the i-transforms. The purposes are 1) to reduce the redundancy to certain extent while preserving data information; and more importantly, 2) to establish a feasible link between points in the original and projected feature spaces. For instance, given a point \( \mathbf{y} = \Theta_i^T \mathbf{x} \) in the projected feature space (via \( \Theta_i \)), its corresponding point in the original feature space is given by \( \Theta_i \mathbf{y} \); and subsequently, 3) to simplify the reconstruction loss term (shown as follows). Therefore, we can formulate the reconstruction loss as \( \|\mathbf{X}_i - \Theta_i \Theta_i^T \mathbf{X}_i\|_F^2 \).

To simplify this term further:

\[
\|\mathbf{X}_i - \Theta_i \Theta_i^T \mathbf{X}_i\|_F^2 = \text{tr}(\mathbf{X}_i^T (\mathbf{I} - \Theta_i \Theta_i^T) (\mathbf{I} - \Theta_i \Theta_i^T) \mathbf{X}_i)
\]

\[
= \text{tr}(\mathbf{X}_i^T (\mathbf{I} - \Theta_i \Theta_i^T) \mathbf{X}_i) - 2 \text{tr}(\mathbf{X}_i^T \Theta_i \Theta_i^T \mathbf{X}_i) + \|\Theta_i \mathbf{X}_i\|_F^2
\]

Here, \( \text{tr}(\cdot) \) represents a matrix trace operator. By discarding the constant term \( \|\mathbf{X}_i\|_F^2 \), the reconstruction term can be reformulated as

\[
R_2(\{\Theta_i\}) = -\gamma \|\Theta_i^T \mathbf{X}_i\|_F^2.
\] (2)

**Regularization term** \( R_3(\mathbf{W}_0, \{\mathbf{W}_i\}) \). The regularization term \( \frac{\alpha}{S} \|\mathbf{W}_0\|_F^2 + \beta \|\mathbf{W}_i\|_F^2 \), a Frobenius Norm on matrices \( \mathbf{W}_0 \) and \( \mathbf{W}_i \) parameterized by \( \alpha \) and \( \beta \) (\( S \) is a constant, the number of heterogeneous features), aims to achieve a reliable generalization of our joint learning model. The two parameters can also control the values of the mined shared and specific components. Larger \( \alpha \) leads to a smaller shared component and larger \( \beta \) results in smaller specific components. Integrating this regularization term can also help deriving a closed form solution of \( \mathbf{W}_0 \) and \( \mathbf{W}_i \) during the iterative optimization presented later.

By substituting all the terms into the objective function, our problem can be rewritten as:

\[
\begin{aligned}
& \min_{\mathbf{W}_0, \{\mathbf{W}_i\}, \{\Theta_i\}} \sum_{i=1, \ldots, S} \left( \|((\lambda \mathbf{W}_0 + (1 - \lambda) \mathbf{W}_i)^T \Theta_i^T \mathbf{X}_i - \mathbf{Y}_i\|_F^2 - \gamma \|\Theta_i^T \mathbf{X}_i\|_F^2 \right) + \alpha \|\mathbf{W}_0\|_F^2 + \beta \sum_{i=1, \ldots, S} \|\mathbf{W}_i\|_F^2 \\
& \quad \text{s.t. } \Theta_i^T \Theta_i = \mathbf{I}, i = 1, 2, \ldots, S
\end{aligned}
\] (3)

### 4.2 Three-step Iterative Optimization

We solve our joint learning model by a coordinate descent algorithm that optimizes over one set of the parameters at each step while keeping the others fixed. The optimization is achieved by iterating the following three steps, which in a row monotonically decreases the objective function in Formula (2) with a guaranteed convergence to a local optimal solution.

**STEP 1.** Fixing the coefficients \( \mathbf{W}_i \) and \( \Theta_i \), minimize the following function \( J_1 \) over \( \mathbf{W}_0 \):

\[
\min_{\mathbf{W}_0} \sum_{i=1}^{S} \left( \|((\lambda \mathbf{W}_0 + (1 - \lambda) \mathbf{W}_i)^T \Theta_i^T \mathbf{X}_i - \mathbf{Y}_i\|_F^2 + \alpha \|\mathbf{W}_0\|_F^2 \right)
\]

(4)

This is an unconstrained minimization problem, whose solution can be given by \( \mathbf{W}_0 = (\lambda^2 \sum_{i=1}^{S} \Theta_i^T \mathbf{X}_i \mathbf{X}_i^T \Theta_i + \alpha \mathbf{I})^{-1} \sum_{i=1}^{S} \Theta_i^T \mathbf{X}_i \mathbf{Y}_i^T - (1 - \lambda) \lambda \mathbf{X}_i^T \Theta_i \mathbf{W}_i \mathbf{W}_i^T \Theta_i \mathbf{X}_i \mathbf{W}_0 \).

We also note that the second derivative of the objective function \( J_1 \) can be given by

\[
\frac{\partial^2 J_1}{\partial \mathbf{W}_0^2} = 2(\lambda^2 \sum_{i=1}^{S} \Theta_i^T \mathbf{X}_i \mathbf{X}_i^T \Theta_i + \alpha \mathbf{I}) \geq 0
\]

where \( \geq 0 \) indicates positive semidefinite. Hence, the derived optimal solution \( \mathbf{W}_0 \) would decrease the value of the objective function.

**STEP 2.** Fixing the coefficients \( \mathbf{W}_0 \) and \( \Theta_i \), optimize \( \mathbf{W}_i \):

\[
\min_{\mathbf{W}_i} \sum_{i=1}^{S} \|((\lambda \mathbf{W}_0 + (1 - \lambda) \mathbf{W}_i)^T \Theta_i^T \mathbf{X}_i - \mathbf{Y}_i\|_F^2 + \beta \|\mathbf{W}_i\|_F^2
\]

The above problem can be decomposed into \( S \) independent Frobenius-regularized unconstrained least square problems:

\[
\min_{\mathbf{W}_i} \|((\lambda \mathbf{W}_0 + (1 - \lambda) \mathbf{W}_i)^T \Theta_i^T \mathbf{X}_i - \mathbf{Y}_i\|_F^2 + \beta \|\mathbf{W}_i\|_F^2
\]

(5)

By setting the first order derivatives of the above function [5] to zero, we can obtain the optimal solution: \( \mathbf{W}_i = (1 - \lambda)^2 \Theta_i^T \mathbf{X}_i \mathbf{X}_i^T \Theta_i + \beta \mathbf{I}^{-1} \Theta_i^T \mathbf{X}_i (\mathbf{Y}_i^T - \lambda \lambda^2 \Theta_i \mathbf{W}_0 \mathbf{W}_0^T \Theta_i) \). Similar to STEP 1, we can easily derive the second derivative as

\[
\frac{\partial^2 J_2}{\partial \mathbf{W}_i^2} = 2((1 - \lambda)^2 \Theta_i^T \mathbf{X}_i \mathbf{X}_i^T \Theta_i + \beta \mathbf{I}) \geq 0
\]

Here, \( J_2 \) indicates the objective function in Formula [5]. Hence, it is convex with respect to \( \mathbf{W}_i \), which indicates that the updating scheme at STEP 2 would decrease the value of our objective function in Formula [5] and minimize the function.
STEP 3. Finally, we fix \( W_0, W_i \) and optimize \( \Theta_i \):

\[
\min_{\Theta_i} \sum_{i=1}^{S} \left( \frac{1}{\lambda} \| W_0 + (1 - \lambda) W_i \|^T \Theta_i^T X_i - Y_i \|^2_F - \gamma \| \Theta_i^T X_i \|^2_F \right)
\]

s.t. \( \Theta_i^T \Theta_i = I, i = 1, 2, \ldots, S \)

Note that all the \( \Theta_i \)'s in the above system are independent. Hence, we turn to solving the following \( S \) independent subproblems:

\[
\min_{\Theta_i} \left( \frac{1}{\lambda} \| W_0 + (1 - \lambda) W_i \|^T \Theta_i^T X_i - Y_i \|^2_F - \gamma \| \Theta_i^T X_i \|^2_F \right)
\]

s.t. \( \Theta_i^T \Theta_i = I \)

(6)

It is not easy to solve the problem in Formula (6) directly in the Euclidean space due to the non-convex constraints. We optimize each subproblem with a gradient based method on the Stiefel manifold where the approximate solution is required to satisfy the orthogonality constraint in each iteration [40]. Specifically, given the \( t^{th} \) step estimator of \( \Theta_i(t) \), we first define a skew-symmetric matrix \( \nabla = G \Theta_i(t) G^T - \Theta_i(t) \), where \( G \) is the gradient of the objective function in the Euclidean space and it can be indicated by \( G = X_i (\lambda W_0 + (1 - \lambda) W_i) \Theta_i(t)^T X_i - Y_i ) \) with \( \lambda W_0 + (1 - \lambda) W_i \) - 2\( \gamma \)X_i\( \Theta_i(t)^T X_i \). Then the new updated point can be determined by the Grank-Nicolson-like scheme \( \Theta_i(t+1) = (I + \tau \nabla)^{-1} (I - \frac{\tau}{2} \nabla) \Theta_i(t) \), where \( \tau \) is the iteration step size and an optimal step size could be determined by a line search method within each iteration. We summarize the optimization for the objective function in Formula (3) in Algorithm [4].

Here, we would like to point out that the employed updating scheme at STEP 3 still makes the objective function decrease. We provide our proof in the supplementary file based on some tricks provided in [40].

As discussed above, all the three steps in our optimization method would decrease the objective function in our JOULE model. Since

\[
-\| \Theta_i^T X_i \|^2_F = -\| X_i \|^2_F + \| X_i - \Theta_i \Theta_i^T X_i \|^2_F \geq -\| X_i \|^2_F,
\]

the objective function in Formula (3) is lower bounded when \( \alpha, \beta, \gamma \geq 0 \). Therefore, the proposed optimization algorithm can converge to a minimum in practice.

4.3 Inference

Given the model parameters \( W_0, W_i \) and \( \Theta_i \), the inference is to predict the best activity label for a new sample with heterogeneous features \( x_i, i = 1, 2, \ldots, S \). We first define two confidence vectors to encode the shared and specified components of \( x_i \) as

\[
C_{\text{shared}}^i = \alpha W_0^T \Theta_i^T x_i \in \mathbb{R}^L
\]

\[
C_{\text{specified}}^i = (1 - \alpha) W_i^T \Theta_i^T x_i \in \mathbb{R}^L
\]

(7)

Here, \( \lambda \) is the model parameter used to balance the contribution of the shared and specific structures during training. Specifically, when \( \lambda = 0 \), a model without forming any shared components is formulated, while setting \( \lambda = 1 \) formulates a baseline without specific structures explored. The effect of \( \lambda \) will be discussed in the experimental section.

Inspired by the construction of augmented features in [16], here we treat all the shared and specific confidence vectors as higher-level augmented features and concatenate them together to form our final representation. To speed up our testing, a linear SVM classifier was first trained on the augmented features from the training set and then subsequently used to make the final decision for a test image.

5 Transfer Joint Heterogeneous Feature Learning

It is challenging to learn a set of reliable i-transforms and shared-specific structures from a target set (a dataset where testing is carried out) with limited training samples. This situation could be mitigated by using some non-target sets (widely known as transfer learning). Thanks to the nature of joint learning, our JOULE model could generalize well to this case. Here, we introduce a transfer learning model to enhance our feature learning on the target set by the assistance of the learning on other non-target datasets.

Specially, we utilize samples from a non-target set as our auxiliary set to assist our feature learning on the target set. For clarity, in the following, we will use \( \text{auxiliary set} \) to denote non-target set. Let \( W_0, \{ W_i \} \) and \( W_0, \{ W_i \} \) be the shared and specific structures to be mined in the target (and auxiliary) set, respectively. For transferring the learning from an auxiliary set to a target set, we assume that the i-transforms \( \{ \Theta_i \} \) can be shared for the same type of features across datasets, so that the data in the auxiliary set can provide a strong prior for our feature learning on the target set. Therefore, the feature learning on the target and auxiliary sets are connected by \( \{ \Theta_i \} \) and they can be optimized jointly. Let \( \{ X^i \}, \{ Y^i \} \) and \( \{ X^j \}, \{ Y^j \} \) denote the feature representation and label information of the auxiliary set and the training samples from target set. Our transfer joint learning model is formulated
as:

\[
\min_{\mathbf{W}_t, (\Theta_i), \mathbf{W}_0} \rho \sum_{i=1}^{S} \frac{F([X^i], [Y^i], \mathbf{W}_0, \{\mathbf{W}_t\}, \{\Theta_i\})}{|\Theta|}
\]

\[
\text{Learned on the training samples of target set}
\]

\[
(1 - \rho) F([X^i], [Y^i], \mathbf{W}_0, \{\mathbf{W}_t\}, \{\Theta_i\})
\]

\[
s.t. \mathbf{W}_t^T \Theta_i = \mathbf{I}, i = 1, 2, \ldots S
\]

(8)

Here, \( F(\cdot) \) is the objective function of our JOULE model \( \Theta \) in the form of \( \sum_{i=1}^{S} (R_1(\cdot) + R_2(\cdot) + R_3(\cdot)) \). The first \( F(\cdot) \) function is defined on the training samples from the target set and the second \( F(\cdot) \) function on the auxiliary sets. We use the parameter \( \rho \in [0, 1] \) to control the effect of the auxiliary set. Specifically in the case of \( \rho = 0 \), the i-transforms \( \{\Theta_i\} \) are solely determined by the feature learning in the auxiliary set.

Similar to JOULE, we develop a three-step optimization algorithm to solve problem (8), i.e., iteratively optimizing the objective function over one set of parameters with the others fixed (e.g., at one step, we optimize over the shared components \( \mathbf{W}_0 \) and \( \mathbf{W}_t^0 \) by fixing the others.). The only difference is that the i-transforms \( \{\Theta_i\} \) are optimized simultaneously on both target and auxiliary datasets. The gradient of the objective function in problem (8) with respect to \( \Theta_i \) can be given by

\[
\rho \frac{\partial F([X^i], [Y^i], \mathbf{W}_0, \{\mathbf{W}_t\}, \{\Theta_i\})}{\partial \Theta_i} + (1 - \rho) \frac{\partial F([X^i], [Y^i], \mathbf{W}_0, \{\mathbf{W}_t\}, \{\Theta_i\})}{\partial \Theta_i}
\]

which is a combination of gradients in the target set and auxiliary set. It is easy to see that in the extreme cases when \( \rho = 0 \) (or \( \rho = 1 \)), the i-transforms \( \{\Theta_i\} \) will be derived solely from the auxiliary set (or the target set). After all the parameters are learned, the inference step is actually identical to the JOULE model described in Section 4.3 and the corresponding decisions are made on the testing samples from the target set using the learned parameters: \( \mathbf{W}_0, \{\mathbf{W}_t\}, \{\Theta_i\} \).

6 EXPERIMENTS

We evaluated our methods extensively on three benchmark 3D activity datasets and one newly collected human-object interaction dataset. In the following, we first briefly introduce the implementation details, and then describe the experiments and results.

6.1 Implementation Details

The model parameters \( \alpha, \beta, \gamma, \lambda \) were fixed as \( 10^{-1}, 10^{-1}, 1 \) and \( \frac{1}{2} \), respectively through all our experiments. The dimensionality \( M \) of the subspace is specified empirically for each dataset. Intuitively, it is suggested to be smaller than the number of training samples. We will investigate its effect in detail in Section 6.6.

When computing DCP and DDP features, one image patch of size \( 60 \times 60 \) was extracted around each joint position in a trajectory in order to capture the context cues. A set of image patches were extracted for each trajectory. For computational efficiency, all the image patches were then resized to \( 32 \times 32 \) and the cell size of HOG was set to 8.

6.2 MSR Daily Activity Dataset

We tested the proposed methods on the MSR Daily Activity dataset [17], which has become a standard set for studying 3D human activities. It contains 320 video clips of 16 different activities (drinking, eating, walking, cheering up, reading book, etc) performed by 10 participants in two different poses, namely sitting and standing. Most of the activities involve human-object interactions (see Table 4). We followed the same experimental settings as in other related works, where half of the participants were used for training and the rest for testing.

To evaluate our proposed JOULE model, we compare with a baseline implementation that fuses different features together with a standard SVM classifier, MTDA [51] and HFM [3]. We denote these baselines as “SVM”, MTDA, and HFM. In addition, we also compare with the MPCCA model presented in [3], which intends to discover shared-specific structures in a non-discriminative learning framework. We also present the recently reported results of other 10 different methods for comparison. The dimensionality \( M \) for our JOULE model was set to 40.

Results. Table 1 shows the results and comparison. Our method obtains an accuracy of 95%, which exceeds most of the latest reported results and is comparable with the state-of-the-art [20]. However, we would like to point out that Lu et al. [20] requires a clear pixel-wise segmentation of the actor, background and occlusion objects, which may render it unsuitable for activities with more complex interactions and cluttered background. Compared to the closely related methods focusing on feature fusion using deep model [19] and structured sparse model [28], our model outperforms both of them by a considerable margin (more than 9.4%), which implies our feature learning system is superior to other RGB-D activity fusion systems. Compared with the baseline of SVM, the performance gain (95% vs. 90%) by our JOULE model demonstrates the benefits of the shared and specific components modeling. Our JOULE outperforms MTDA and HFM considerably by 4.4% and 10.6% using exactly the same set of features. It is worth noting that MTDA did not seek to learning feature-specific structures. The superior performance of our JOULE over MTDA indicates that modeling feature-specific structures is essential for capturing the complex connections among the employed heterogeneous features. It is also observed that HFM performed worse than the baseline of SVM. Bear in mind that, in order to compute the similarity of two training instances, HFM needs to manually select a proper kernel for each feature type, which is a big challenge in the presence of noisy heterogeneous features (e.g., part of our DCP features were extracted from the background pixels). Therefore, the resulting similarity matrix could be unreliable and HFM might not cope with our features well. In our implementation of HFM, we used both RBF kernel and linear kernel to measure the similarity between two features, which was suggested in [4]. In contrast, the SVM will adaptively learn a set of weights to encode the contribution of each feature dimension in a discriminative framework and thus can be more applicable in our RGB-D activity recognition. The MPCCA is an approach close to the proposed JOULE, but it performed clearly inferior to JOULE. One of the reasons is that the Gaussian noise assumption in MPCCA is not sufficient to describe the specific information of each feature channel. Moreover, JOULE also benefited from learning discriminant shared-specific structure.

The confusion matrix of the results by our JOULE model is shown in Figure 4. It can be seen that our model achieves perfect classification results on 10 classes. The larger error is due to the mis-classification of the activity of writing on a paper as reading book, which may be largely attributed to high similarity between the object and activity contexts in these two activities.
### 6.3 Cornell Activity Dataset 60 (CAD 60)

This public dataset consists of 68 video clips captured by Microsoft Kinect device \[31\]. Four actors were asked to perform 13 specific activities (still, talking on the phone, and etc.) and one random activity in 5 different environments: office, kitchen, bedroom, bath room, and living room. We followed the same experimental setting in \[37\] by adopting the leave-one-person-out cross validation for each environment, which ensures that person participating in the training cannot be seen in the testing. The final accuracy was calculated by averaging the accuracies of all the possible splits (totally 20 in this set).

Our methods are compared with the results reported in the state-of-the-art \[37\]. We also ran the released code of HON4D on this set and listed the recognition results as “Reported Results” in Table 2. Since there is no default parameter settings suggested by the author on this set, we report the best results by varying their parameters in a wide range. Similar to MSR Daily set, we also highlight the benefits of using JOULE model by comparing with the baseline SVM, MTDAD and HFM. Here, the dimensionality \( M \) of \( W_i \) (and \( W_0 \)) is set as 4 on this dataset.

**Results.** The results and comparison are shown in Table 2. Our method achieves an accuracy of 84.1\%, which significantly outperforms the state-of-the-art result \[37\] by a large margin (9.4\%). It is worth noting that most of our baseline implementations including the simple combination of our heterogeneous features with a standard SVM classifier can achieve a performance comparable to the state-of-the-art method with carefully designed classifiers, which proves that our feature is superior to that developed in \[37\]. Especially, by considering the shared and specific components, our model (JOULE) obtains a gain of 9.1\% compared with the fusion methods using standard SVM classifier without explicitly modeling shared and specific components (84.1\% vs. 75\%), and a significant gain of 11.4\% compared with HFM. In addition, our JOULE works better than MTDAD on CAD 60 set with a smaller performance gain than on the MSR Daily set.

The confusion matrix of the results by our JOULE model is presented in Figure 5. It can be seen that our model can distinguish well the five activities of rinsing mouth with water, wearing contact lenses, cooking(chopping), working on computer and random activities, which demonstrates that our model can effectively capture the interactions between human and the manipulated object. It can also be observed that the activities of talking on couch and relaxing on couch are often confused by our model, mainly due to the inaccurate human skeletons captured by the Kinect camera.
6.4 Composable Activities Dataset

This dataset consists of 693 video clips performed by 14 participants. Each participant was asked to perform 16 complex activities (Walk while calling with hands, Walk while hand waving, and etc.) several times. All the considered activities in this set are composed by a number of mid-level actions such as walking, waving hand, reading etc., and about 75% of them contain human-object interactions. For a fair comparison, we followed exactly the same leave-one-subject-out experimental setting as in [18], where each time the activity samples performed by 13 participants were all used to train a model and the rest were used for testing. And finally, the average accuracies were computed and reported.

Here, we directly compare the performance of our method with the results reported in the state-of-the-art [18]. Meanwhile, we also ran the released code of HON4D by the author on this set, and again report the best results by varying their parameters in a wide range. In addition, we further compared the JOULE with the baseline “SVM”, MTDA and HFM. In this experiment, we set the wide range. In this experiment, we set the dimensionality $M$ of the subspace as 100. Its influence would be further discussed in Section 6.6.

Results. The results and comparison are shown in Table 3. As shown, simply feeding the concatenation of all primal heterogeneous features into a SVM classifier without explicitly considering their hidden structures and connections achieves an accuracy of 88.32% and outperforms the state-of-the-art [18] by a margin of 2.6%. As expected, the performance gap becomes larger ($\geq 5.9\%$) when our proposed JOULE model is employed to explicitly model the shared and specific structures among different heterogeneous features. Similar to the observations on other datasets, our JOULE outperforms MTDA by over 2% on the Composed Activities Datasets, which once again experimentally confirms that the learning of feature-specific structures is beneficial.

By closely examining the confusion matrix in Figure 6, we can observe that JOULE achieves perfect recognition performance on most of the activities. The most challenging activities for our model are “Walk while calling with hands” and “Walk while hand waving”, which are often confused with each other. This is not surprising, because these two activities contain highly similar motions, and the subtle difference between them is that activity “calling with hands” often involves a motion of moving fingers or hands back and forth, while “waving hands” refers to a slight hand movement of moving between left and right. However, it is quite challenging to capture these tiny differences by the prevailing Kinect cameras available in the market with standard specification of spatial and depth resolution.

To appear in IEEE Transactions on Pattern Analysis and Machine Intelligence 2017

<table>
<thead>
<tr>
<th>Method</th>
<th>Accuracy (%)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Reported Results</td>
<td>HON4D [27]</td>
</tr>
<tr>
<td></td>
<td>Hierarchical Model [18]</td>
</tr>
<tr>
<td></td>
<td>85.7</td>
</tr>
<tr>
<td>Our Results</td>
<td>HFM [4]</td>
</tr>
<tr>
<td></td>
<td>84.44</td>
</tr>
<tr>
<td></td>
<td>SVM</td>
</tr>
<tr>
<td></td>
<td>88.32</td>
</tr>
<tr>
<td></td>
<td>MPCCA [53]</td>
</tr>
<tr>
<td></td>
<td>90.76</td>
</tr>
<tr>
<td></td>
<td>MTDA [53]</td>
</tr>
<tr>
<td></td>
<td>92.07</td>
</tr>
<tr>
<td></td>
<td>JOULE</td>
</tr>
<tr>
<td></td>
<td>94.24</td>
</tr>
</tbody>
</table>

Fig. 6. Confusion matrix of JOULE on Composable Activities dataset.

### Table 3

Comparison of the Composable Activities dataset.

<table>
<thead>
<tr>
<th>Dataset Description</th>
<th>Data</th>
<th>Cla. No.</th>
<th>Sub. No.</th>
<th>Vid. No.</th>
<th>HOI Ra</th>
</tr>
</thead>
<tbody>
<tr>
<td>CAD 60 [41]</td>
<td>RGB-D</td>
<td>14</td>
<td>4</td>
<td>68</td>
<td>85.7%</td>
</tr>
<tr>
<td>MSRDaily [37]</td>
<td>RGB-D</td>
<td>16</td>
<td>10</td>
<td>320</td>
<td>87.5%</td>
</tr>
<tr>
<td>MSRAAction [17]</td>
<td>Depth</td>
<td>10</td>
<td>20</td>
<td>567</td>
<td>70%</td>
</tr>
<tr>
<td>Comp. Activities [18]</td>
<td>RGB-D</td>
<td>16</td>
<td>14</td>
<td>693</td>
<td>75%</td>
</tr>
<tr>
<td>Multiview [39]</td>
<td>RGB-D</td>
<td>8</td>
<td>8</td>
<td>3815</td>
<td>100%</td>
</tr>
<tr>
<td>SYSU 3D HOI</td>
<td>RGB-D</td>
<td>12</td>
<td>40</td>
<td>480</td>
<td>100%</td>
</tr>
</tbody>
</table>

6.5 SYSU 3D Human-Object Interaction Set

Dataset Description. We have collected a new RGB-D activity dataset focusing on human-object interactions to further evaluate all methods. We name this as SYSU 3D Human-Object Interaction (HOI) dataset. For building this set, 40 participants were asked to perform 12 different activities freely. For each activity, each participant manipulates one of the six different objects: phone, chair, bag, wallet, mop and besom. Therefore, there are totally 480 video clips collected in this set. The contained activity samples have different durations, ranging from 1.9s to 21s. For each video clip, the corresponding RGB frames, depth sequence and skeleton data were captured by a Kinect camera. Activity samples are shown in Figure 8. We highlight the differences between our 3D HOI set and relevant existing sets in Table 4. Compared to those datasets (MSRDaily, CAD 60, MSRAAction, Composable Activities dataset, and Multiview set), our dataset presents new challenges: 1) the involved motions and the manipulated objects’ appearance are highly similar among some activities; for instance, the manipulated objects besom and mop involved in the activities mopping and sweeping are highly similar; 2) the number of participants is three times (or even larger than in most cases) that of existing ones, so that more inter-subject variations could be observed for the same type of activities due to the different characteristics of participants.

Evaluation Protocol. We tested all the compared methods in two different settings. In the first setting (setting-1), for each activity class, we selected half of the samples for training and the rest for testing. In the second setting (setting-2), video sequences
performed by half of the participants were used to learn model parameters and the rest for testing, where there is no overlap of participants between the training and test set. This is a cross-subject setting. For each setting, we report the mean accuracy and standard deviation of the results over 30 random splits.

**Baselines.** Similar to that on the MSRDaily and CAD60 sets, the baselines SVM, HFM, MPCCA, MTDA and HON4D are compared to show the effectiveness of our joint learning model (JOULE). We set \( M = 30 \) in our model. In total, we report a comprehensive set of results of up to six different implementations on this new dataset.

**Results.** Table 5 reported the results. Again, using the proposed JOULE model to fuse different heterogeneous features is always beneficial in all settings. The accuracies in setting-2 are higher than that of setting-1 without considering cross-subject split. This is because the prediction could be biased by appearance when activities with similar motion and object context (e.g. mopping vs. sweeping) performed by the same participant are contained in both training and test sets, which may occur in the setting-1. The performances of JOULE and MTDA are comparable with JOULE performing perceivably better. It was noted that the performance gap between our models and the baselines is smaller (e.g., 84.9% vs. 82.8%) than that on the other three datasets. This somehow indicates the new dataset is more challenging for feature fusion.

By examining the confusion matrices of our JOULE model in Figure 7, we observed that our model often confuses the activities of mopping with sweeping in both settings, which is mainly due to similar motions and objects appearance in the two interactions. In addition, the activities of taking from wallet share similar motions with activities of playing phone and taking out wallet, which are occasionally misidentified as playing phone or taking out wallet.

### 6.6 Analysis and Discussion

**Convergence.** Our method converges to a minimum after a limited number of iterations. We empirically observed that 20 iterations (outer iterations i.e. term \( \text{IterOut} \) in Algorithm 1) are sufficient for obtaining a reliable solution in all of our experiments. See Figure 8 for an example illustrating the convergence of our method on the MSR Daily activity set, where the objective value of each step was recorded during each iteration. Excluding the time for computing the features, one round training of our algorithm takes about 1.26 minutes per training sample. However, our testing is pretty fast, and takes about 0.5 second per sample. Computing the DS, DCP, and DDP features cost time. It takes about 0.24 second for processing each frame of a RGB-D video using MATLAB on a normal desktop PC (CPU i5-4570, memory 28G).

**Effect of dimensionality M.** We investigate the effect of the dimensionality \( M \) of the subspace. Figure 10 shows the performances of our method JOULE with different values of \( M \). Generally, a very small \( M \) leads to an inferior performance, as the smaller dimensionality of the subspace is, the less representative it is for the original features. When \( M \) becomes larger (typically larger than a value about \( \frac{1}{6} \sim \frac{1}{4} \) of the number of training samples), the performances start to remain stable, which means our algorithm is not sensitive \( M \) in a reasonable range.

**Effect of TPF on gradient signal.** In this work, we have modified temporal pyramid Fourier features (TPF) developed in \([37]\) so as to

---

**Table 5**

Comparison on the SYSU 3D HOI dataset.

<table>
<thead>
<tr>
<th>Method</th>
<th>Mean Acc±std (%)</th>
</tr>
</thead>
<tbody>
<tr>
<td>HON4D [27]</td>
<td>73.39 (±2.59)</td>
</tr>
<tr>
<td>HFM [1]</td>
<td>75.03 (±2.68)</td>
</tr>
<tr>
<td>MPCCA [3]</td>
<td>76.25 (±2.36)</td>
</tr>
<tr>
<td>SVM</td>
<td>77.34 (±2.53)</td>
</tr>
<tr>
<td>MTDA [51]</td>
<td>79.19 (±4.27)</td>
</tr>
<tr>
<td>JOULE</td>
<td>79.63 (±2.13) 84.89 (±2.29)</td>
</tr>
</tbody>
</table>

Fig. 7. Confusion matrices of JOULE on SYSU 3D HOI set under setting-1 (a) and setting-2 (b).

Fig. 9. Illustration of the convergence of our method. The vertical axis indicates the value of objective function and the horizontal axis is the number of iterations.
apply both the original feature signal and its gradient to implicitly encode human motions, since they are complementary to each other. The TPF of original signal captures the original signal cues, whereas the TPF of gradient signal encodes the first derivative (velocity) information. Table 6 shows the results of our model with and without temporal Fourier features computed from the gradient signal on all of the three datasets. It can be seen that, while the improvement on the SYSU 3D HOI dataset is relatively mild, TPF features on gradient consistently improve the results in all of the cases, with the biggest gain (7.6%) achieved on the CAD60 dataset. This indicates that the proposed extension of TPF features to the gradient signal is promising and effective.

**TABLE 6**

<table>
<thead>
<tr>
<th></th>
<th>MSRD</th>
<th>CAD60</th>
<th>Comp. Act.</th>
<th>3DHOI (s-1)</th>
<th>3DHOI (s-2)</th>
</tr>
</thead>
<tbody>
<tr>
<td>With</td>
<td>95</td>
<td>84.1</td>
<td>94.24</td>
<td>79.63</td>
<td>84.89</td>
</tr>
<tr>
<td>Without</td>
<td>91.25</td>
<td>76.5</td>
<td>92.22</td>
<td>78.83</td>
<td>83.63</td>
</tr>
</tbody>
</table>

**Fig. 10. Effects of parameter $M$ on the system performance.**

**Fig. 11. Effects of parameters $\alpha$ (the vertical) and $\beta$ (the horizontal) on the system performance (%) on the cross-subject settings of Composable Activities Dataset and SYSU set.**

**Effect of $\alpha$ and $\beta$.** As discussed in previous sections, the parameters $\alpha$ and $\beta$ were employed to control the generalization ability of our joint learning model. Here, we investigate their influence on Composable Activities dataset and SYSU with setting-2, where cross-subject settings (i.e., half of the subjects for training, and the rest for testing) are employed. In this test, parameters $\alpha$ and $\beta$ were both selected from $\{0, 10^{-2}, 10^{-1}, 10^0, 10^1, 10^2\}$, and therefore we have a total of 36 different parameter settings. We present the recognition results in Fig. 11. It could be observed that, generally large $\alpha$ and $\beta$ ($\gtrsim 10$) lead to an inferior performance. This is because the larger the $\alpha$ and $\beta$ are, the less the shared and specific components are discovered for recognition. However, when $\alpha$ and $\beta$ are smaller than 1, the performance would remain relatively stable in most cases, which demonstrates that our method is insensitive to the parameters in a reasonable range. This study also reveals that the optimal ranges of $\alpha$ and $\beta$ are approximately the same, which indicates that we can simply set $\alpha = \beta$ (e.g., both were set as 0.1 in all of the other experiments) to reduce the number of parameters without affecting the system performance too much.

**Influence of $\lambda$.** In our joint learning framework, we introduce a parameter $\lambda$ to explicitly control the trade-off between the shared structure $W_0$ and feature-specific structures $\{W_i\}_{i=1,2,\ldots,S}$. Here, we evaluate its influence by setting $\lambda$ as $0, 0.25, 0.5, 0.75,$ and 1, respectively, and then report the achieved performances in Table 7. As expected, a proper combination of the shared and specific structures gives a better result; generally too small or too large $\lambda$ would result in an inferior performance. Especially, without modeling the specific structures ($\lambda = 1$) or shared

**Fig. 8. Snapshots of activities in SYSU 3D HOI set, one sample per class. The rows headed with RGB show the samples in RGB channel and the rows underneath headed with Depth show the corresponding depth channel superimposed with skeleton data. Best viewed in color.**
structure ($\lambda = 0$), the performance decreased in both cases. Overall, albeit not always the best, on all of the four datasets considered, $\lambda = 0.5$ is an acceptable setting.

**Influence of $\gamma$.** In the JOULE model (Formula (1)), we employed a reconstruction loss term (parametered by $\gamma$) to regularize the i-transforms learning in order to preserve as much information as possible. Here, we investigate its influence by varying it systematically. The results are presented in Table 8. As shown, the model performed the best when $\gamma = 1$ on most of the datasets. In general, a smaller or larger $\gamma$ would lead to lower recognition accuracies. In particular, when $\gamma$ is zero and the reconstruction term is not used to constrain the i-transforms learning, lower recognition results were observed.

**Single vs. Multi Channels.** In the JOULE model, we have integrated the learning of features from different channels (RGB, depth (DEP) and skeleton (SKL)) in a framework so that the learning of one channel can facilitate the learning of other channels. To investigate the benefits of joint learning, we tested the JOULE by feeding it with 1) features from one channel only and 2) features from two or more channels, respectively. Therefore, we tested 7 cases for each dataset. In total we conducted 35 experiments, and results are summarized in Table 9. It can be seen that the performances of learning features from two channels are higher than each of them alone. Using features from three channels always outperform one or two channels. This demonstrates that jointly learning the features from different channels is beneficial.

### Table 7
Effects of parameter $\lambda$ on recognition (%).

<table>
<thead>
<tr>
<th>Dataset</th>
<th>$\lambda = 0$</th>
<th>$\lambda = 0.25$</th>
<th>$\lambda = 0.5$</th>
<th>$\lambda = 0.75$</th>
<th>$\lambda = 1$</th>
</tr>
</thead>
<tbody>
<tr>
<td>MSRD</td>
<td>90.62</td>
<td>91.87</td>
<td>95</td>
<td>92.5</td>
<td>91.25</td>
</tr>
<tr>
<td>CAD60</td>
<td>82.58</td>
<td>83.33</td>
<td>84.1</td>
<td>85.61</td>
<td>82.58</td>
</tr>
<tr>
<td>Comp. Act.</td>
<td>91.21</td>
<td>93.80</td>
<td>94.24</td>
<td>93.37</td>
<td>91.50</td>
</tr>
<tr>
<td>SYSU(s-1)</td>
<td>78.83</td>
<td>80.24</td>
<td>79.63</td>
<td>79.50</td>
<td>78.89</td>
</tr>
<tr>
<td>SYSU(s-2)</td>
<td>83.81</td>
<td>84.65</td>
<td>84.89</td>
<td>85.15</td>
<td>84.58</td>
</tr>
<tr>
<td>Average</td>
<td>83.44</td>
<td>86.78</td>
<td>87.57</td>
<td>87.23</td>
<td>85.76</td>
</tr>
</tbody>
</table>

### Table 8
Effects of parameter $\gamma$ on recognition accuracy (%).

<table>
<thead>
<tr>
<th>Dataset</th>
<th>$\gamma = 0$</th>
<th>$\gamma = 0.01$</th>
<th>$\gamma = 0.1$</th>
<th>$\gamma = 100$</th>
<th>$\gamma = 10000$</th>
</tr>
</thead>
<tbody>
<tr>
<td>MSRD</td>
<td>90.6</td>
<td>91.5</td>
<td>95</td>
<td>93.8</td>
<td>90</td>
</tr>
<tr>
<td>CAD60</td>
<td>79.6</td>
<td>81.1</td>
<td>84.1</td>
<td>78.0</td>
<td>76.5</td>
</tr>
<tr>
<td>Comp. Act.</td>
<td>91.4</td>
<td>93.4</td>
<td>94.2</td>
<td>93.7</td>
<td>92.80</td>
</tr>
<tr>
<td>SYSU(s-1)</td>
<td>77.0</td>
<td>80.0</td>
<td>79.6</td>
<td>76.8</td>
<td>76.9</td>
</tr>
<tr>
<td>SYSU(s-2)</td>
<td>81.6</td>
<td>83.0</td>
<td>84.9</td>
<td>84.1</td>
<td>82.9</td>
</tr>
</tbody>
</table>

### Table 9
Effects of jointly learning in different channels. s-1 denotes setting-1 and s-2 for setting-2 applied on the SYSU 3D HOI dataset (%).

<table>
<thead>
<tr>
<th>Data Channel</th>
<th>MSRD</th>
<th>CAD60</th>
<th>Comp. Act.</th>
<th>SYSU(s-1)</th>
<th>SYSU(s-2)</th>
</tr>
</thead>
<tbody>
<tr>
<td>RGB</td>
<td>86.9</td>
<td>78.0</td>
<td>88.9</td>
<td>71.6</td>
<td>80.0</td>
</tr>
<tr>
<td>DEP</td>
<td>84.4</td>
<td>79.6</td>
<td>88.3</td>
<td>74.3</td>
<td>82.3</td>
</tr>
<tr>
<td>SKL</td>
<td>75</td>
<td>77.9</td>
<td>91.2</td>
<td>75.4</td>
<td>76.9</td>
</tr>
<tr>
<td>DEP+RGB</td>
<td>87.5</td>
<td>80.3</td>
<td>90.1</td>
<td>74.8</td>
<td>82.6</td>
</tr>
<tr>
<td>RGB+SKL</td>
<td>91.3</td>
<td>81.1</td>
<td>93.2</td>
<td>76.9</td>
<td>81.4</td>
</tr>
<tr>
<td>DEP+SKL</td>
<td>90.6</td>
<td>82.6</td>
<td>93.2</td>
<td>79.7</td>
<td>83.5</td>
</tr>
<tr>
<td>DEP+RGB+SKL</td>
<td>95</td>
<td>84.1</td>
<td>94.2</td>
<td>80.2</td>
<td>84.9</td>
</tr>
</tbody>
</table>

### 6.7 Experiments on Transfer-JOULE
In this section, we tested the performance of Transfer-JOULE (Formula (8)) and show how the auxiliary set can benefit our heterogeneous features learning on the target set. The experiments were carried out on the SYSU 3DHOI and Composable Activities sets as they are the two largest datasets among those considered.

Firstly, we evaluated the effect of the control parameter $\rho$ by varying its value from 0 to 1. In this evaluation, one of the two datasets is considered as a target set, and the other as the auxiliary set. When SYSU 3DHOI was used as the target set, we followed two different settings (setting-1 and setting-2) as in Section 6.5. When Composable Activities dataset was used as the target set, we followed the leave-one-subject-out setting as described in 6.4. Thus in total, we have three different test cases: 1) SYSU 3DHOI $\rightarrow$ Composable Activities dataset; 2) Composable Activities dataset $\rightarrow$ SYSU 3DHOI (setting-1); 3) Composable Activities dataset $\rightarrow$ SYSU 3DHOI (setting-2), where $\rightarrow$ indicates the direction of transfer, i.e. auxiliary set $\rightarrow$ target set. In each case, we employed the same evaluation protocol as in section 6.5. When Composable Activities dataset was used as the target set, we followed the leave-one-subject-out setting as described in 6.4. Thus in total, we have three different test cases: 1) SYSU 3DHOI $\rightarrow$ Composable Activities dataset; 2) Composable Activities dataset $\rightarrow$ SYSU 3DHOI (setting-1); 3) Composable Activities dataset $\rightarrow$ SYSU 3DHOI (setting-2), where $\rightarrow$ indicates the direction of transfer, i.e. auxiliary set $\rightarrow$ target set. In each case, we employed the same evaluation protocol as in section 6.5. When Composable Activities dataset was used as the target set, we followed two different settings (setting-A and setting-B) as in Section 6.5. When Composable Activities dataset was used as the target set, we followed two different settings (setting-A and setting-B) as in Section 6.5. When Composable Activities dataset was used as the target set, we followed two different settings (setting-A and setting-B) as in Section 6.5. When Composable Activities dataset was used as the target set, we followed two different settings (setting-A and setting-B) as in Section 6.5.
samples per class to train the model and used the rest for testing. In setting-B, we randomly selected a certain number of participants and used all the samples performed by them as the training set. For a fixed number of training samples (or participants) in each setting, we report the average accuracy obtained by 30 trials. The Composable Activities dataset is used as the auxiliary set in both settings. The results are presented in Figure 12. It is observed that in all of the cases tested, with the help of auxiliary set, the performance of our Transfer-JOULE model is consistently higher than that of JOULE. When the number of training samples is smaller (e.g., less than 15), the performance gap gets much larger. The performance gain of using the auxiliary set becomes smaller but clearly noticeable when the number of training samples gets larger, which is as expected. In particular, in the case of one-shot activity recognition where only one target training sample per class is available for the model training, our Transfer-JOULE model can obtain accuracies of 39.17% and 43.04% in the setting-A and setting-B, respectively, which are about 13% higher than the (non-transferred) JOULE model. This clearly demonstrates that with the help of an auxiliary set, our Transfer-JOULE model can learn a set of parameters with better generalization than the (non-transferred) JOULE model.

7 Conclusion
We have proposed a new RGB-D method called joint heterogeneous features learning (JOULE) model to jointly learn heterogeneous features with different number of dimensions for RGB-D activity recognition. A transfer version is also introduced to further facilitate the joint learning on target set via exploiting shared intermediate transforms (i-transforms) from non-target data. Extensive results are reported on four RGB-D activity sets, demonstrating the effectiveness of the proposed methods. A limitation of our method is the assumption that all the considered activities should be fully executed and observed by the system, which makes it less applicable for identifying ongoing activities containing partial activity execution. In the future, we would like to extend the JOULE model so that it can be used for real-time activity recognition or prediction.

Acknowledgment
This work was supported partially by the National Key Research and Development Program of China(2016YFB1001002, 2016YFB1001003), NSFC (61573387, 61472456, 61522115, 61661130157, 61628212), Guangdong Natural Science Funds for Distinguished Young Scholar under Grant S2013050014265, the GuangDong Program (2015B010105005), the Guangdong Science and Technology Planning Project (2016A010102012, 2014B0118003), and Guangdong Program for Support of Top-notch Young Professionals (2014TQ01X779).

References


Jian-Fang Hu received the B.S. degree from the School of Mathematics and Computational Science, Sun Yat-Sen University, Guangzhou, China, in 2010 and 2012, respectively, and he is currently pursuing the Ph.D. degree with the Department of Applied Mathematics. His current research interests include human-object interaction modeling, 3D face modeling, and RGB-D activity recognition.

Wei-Shi Zheng received the Ph.D degree in applied mathematics from Sun Yat-Sen University in 2008. He is a professor in Sun Yat-sen University. He has been a postdoctoral researcher on the EU FP7 SAMURAI Project at Queen Mary University of London. His research interests include person/object association and recognition in visual surveillance. He is a recipient of excellent young scientists fund of the national natural science foundation of China. He has joined Microsoft Research Asia Young Faculty Visiting Programme.

Jian-Huang Lai received the Ph.D. in mathematics from Sun Yat-Sen University in 1999. He is a Professor and the Dean of the School of Information Science and Technology. His current research interests are in the areas of digital image processing, pattern recognition, multimedia communication, wavelet, and its applications. He has published over 100 scientific papers in international journals and conferences including IEEE TPAMI, IEEE TNN, IEEE TIP, IEEE TSMC-B, PR, ICCV, CVPR, and ICDM.

Jianguo Zhang is currently a Reader of Visual Computation at University of Dundee, UK. He received a PhD in National Lab of Pattern Recognition, Institute of Automation, Chinese Academy of Sciences, Beijing, China, 2002. His research interests include visual surveillance, object recognition, image processing, medical image analysis and machine learning.