Multiview Cauchy Estimator Feature Embedding for Depth and Inertial Sensor-Based Human Action Recognition

Yanan Guo, Lei Li, Weifeng Liu, Jun Cheng, and Dapeng Tao

Abstract—The ever-growing popularity of Kinect and inertial sensors has prompted intensive research efforts on human action recognition. Since human actions can be characterized by multiple feature representations extracted from Kinect and inertial sensors, multiview features must be encoded into a unified space optimal for human action recognition. In this paper, we propose a new unsupervised feature fusion method termed Multiview Cauchy Estimator Feature Embedding (MCEFE) for human action recognition. By minimizing empirical risk, MCEFE integrates the encoded complementary information in multiple views to find the unified data representation and the projection matrices. To enhance robustness to outliers, the Cauchy estimator is imposed on the reconstruction error. Furthermore, ensemble manifold regularization is enforced on the projection matrices to encode the correlations between different views and avoid overfitting. Experiments are conducted on the new Chinese Academy of Sciences - Yunnan University - Multimodal Human Action Database (CAS-YNU-MHAD) to demonstrate the effectiveness and robustness of MCEFE for human action recognition.

Index Terms—Pattern recognition, parameter estimation, computer interfaces.

I. INTRODUCTION

HUMAN action recognition [1], [2], [3] is an important but challenging area of research with many practical applications including in video surveillance [4], security [5], and healthcare [6]. Various sensors such as RGB sensors [7], [8], depth sensors (especially Kinect) [9], [10], and inertial sensors [11] have been used to improve human action recognition performance. Human action recognition systems can be classified into two main categories with respect to sensor types: accelerometer-based systems and computer vision-based systems.

Accelerometer-based human action recognition usually investigates acceleration signals from inertial sensors attached to the human body to recognize actions such as jumping, walking, and lying down. A core issue in accelerometer-based human action recognition is how many inertial sensors should be placed on the body and where they should be placed [12], [13]. By combining artificial neural networks with a tree structure, Ermes et al. [14] used wearable inertial sensors to recognize daily actions in an unsupervised manner with inertial sensors attached to the hip and wrist. Based on linear discriminant analysis (LDA), Khan et al. [15] proposed a hierarchical recognition scheme to extract features from a single tri-axial accelerometer attached to the individual’s chest. Feature representation is also important in human action recognition: statistical signal features including root mean square, standard deviation, variance, and mean are commonly used for accelerometer-based human action recognition [16]. In addition to statistical measures, discrete cosine transform (DCT) coefficients [17], autoregressive (AR) coefficients [17], and fast Fourier transform (FFT) coefficients [18] are also frequently used for accelerometer-based human action recognition. Accelerometer-based human action recognition has many advantages; for instance, most inertial sensors are cost effective and work in unconfined environments. However, sensor drift can occur over long operating periods and performance is heavily dependent on sensor location.

For computer vision-based human action recognition, many studies have used conventional RGB sensors, and a number of feature extraction methods [19] (including motion-history images [20] and spatio-temporal interest point (STIP) detector [21]) have been used to significantly boost performance when using RGB data. However, texture and illumination variance are difficult to overcome when using RGB sensors. Furthermore, hardware requirements are usually large. With the emergence of depth sensors, especially cost-effective Kinect depth cameras, real-time depth information can be captured for human action recognition. Compared to RGB sensor-derived images, depth images are insensitive to lighting variance, leading to high human action recognition performance. Feature extraction methods for depth sensor-based human action recognition are also very important: Yang et al. [22] extracted depth motion maps (DMM) by accumulating global actions before computing the histogram of oriented gradients (HOG) as the representation. Xia et al. [23] presented a filtering method to extract STIPs from depth data (DSTIP) that effectively suppressed noisy measurements and built a new depth cuboid similarity feature to describe the local 3D depth cuboid. Ji et al. [9] proposed depth motion sequences (DMS) to represent a depth sequence from depth motion data, and designed a spatio-temporal cuboid pyramid (STCP) to capture the temporal information and geometry cues from DMS. However, depth information is sensitive to transparent and light-absorbing materials.
As noted above, under realistic operating conditions, single sensor modalities are unsuitable under certain conditions. However, depth images and inertial signals are complementary; for example, depth images extract full body movement attributes while inertial signals extract local movement attributes. Therefore, one way to boost human action recognition performance is to simultaneously generate data from depth and inertial sensors. Furthermore, it has been shown that fusion of inertial and depth sensor data improves human action recognition performance [16]. Based on the fact that encoding multiview features into a unified space might optimize human action recognition and the noise from different views influences the discovery of the optimal unified space [24], [25], [26], [27], here we propose a new feature fusion method termed Multiview Cauchy Estimator Feature Fusion (MCEFE) for human action recognition. MCEFE integrates the complementary information encoded in multiple views to find the projected matrices and the unified data representation. Specifically, the Cauchy estimator is introduced in MCEFE to enhance robustness to outliers and ensemble manifold regularization is enforced on the projected matrices to encode correlations between different views and avoid overfitting.

Although multimodal sensor-based human action recognition systems are becoming more common, the number of publicly available human action databases with simultaneous capture of depth and inertial sensor data are limited [28], [29], [30]. Therefore, to facilitate depth and inertial sensor fusion research for human action recognition, we also present a standard multimodal database called the Chinese Academy of Sciences - Yunnan University - Multimodal Human Action Database (CAS-YNU-MHAD). CAS-YNU-MHAD differs from existing publicly available human action databases because: 1) most existing publicly available human action databases perform movements generally on the same spot, while most actions in CAS-YNU-MHAD do the movements from far away to near; and 2) data from CAS-YNU-MHAD were collected from a Kinect camera and 17 inertial sensors attached to different parts of the body to allow investigation of optimal sensor location. Synchronized depth videos, RGB videos, and inertial signals provide a comprehensive set of human actions. CAS-YNU-MHAD will benefit researchers working in different fields including wearable computing, sensor fusion, and computer vision.

For depth and inertial sensor-based human action recognition, we present a new four-step fusion scheme: 1) generating the DMS from depth sensors and then extracting STCPs from the DMS; 2) extracting FFT coefficients from inertial sensors; 3) using PCA on STCP and FFT coefficients respectively to remove redundant information; and 4) encoding the processed STCPs and FFT coefficients into a unified space by MCEFE for subsequent classification.

The remainder of the paper is organized as follows. Related fusion methods and human action databases are briefly reviewed in Section II. The proposed MCEFE is detailed in Section III. Experimental results on the new proposed database are reported in Section IV, and we conclude in Section V.

II. RELATED WORK

We first survey two main bodies of literature: 1) fusion methods for human action recognition, and 2) typical depth and inertial sensor-based human action databases.

A. Fusion methods for human action recognition

Existing fusion methods for human action recognition can be divided into three main categories: data-level fusion, feature-level fusion, and decision-level fusion.

Data-level fusion directly combines incoming raw samples from different sensors. Liu et al. [31] concatenated depth and inertial sensor samples as the fused sensor data and then used hidden Markov modeling (HMM) for classification. Cao et al. [32] proposed a coupled hidden Markov model (CHMM) to find complementary information and correlations across different sensor modalities. Feature-level fusion mainly finds fused features from features extracted from the original data. Chen et al. [16] directly concatenated the features from depth and inertial sensor data as the fused feature. Wu et al. [33] proposed a Bayesian co-boosting framework to combine features from depth and inertial sensor data. Ofli et al. [28] used the popular bag-of-words (BoW) method adapted to each sensor modality together with the outcomes of various combinations of sensor modalities using multiple kernel learning (MKL) [34]. Decision-level fusion fuses the decisions made by each classifier. Kwolek et al. [30] employed acceleration data to trigger depth data processing for fall detection to reduce false alarms. Chen et al. [16] utilized Dempster-Shafer theory [35] to fuse the classification results from two classifiers, each corresponding to one modality.

B. Depth and inertial sensor-based human action databases

The Berkeley multimodal human action database (MHAD) [28] contains 11 actions including: jumping in place, bending-hands up all the way down, jumping jacks, punching, waving right hand, waving two hands, clapping hands, throwing a ball, sit down, sit down and stand up, and stand up. Its data is synchronized from 12 RGB cameras, 6 wearable accelerometers, 2 Microsoft Kinect depth cameras, and 4 microphones. The 6 wearable accelerometers were attached to the ankles, hips, and wrists. The University of Texas at Dallas multimodal human action dataset (UTD-MHAD) [29] contains 27 actions including: (1) right arm swipe to the left, (2) right arm swipe to the right, (3) right hand wave, (4) two hand front clap, (5) right arm throw, (6) cross arms in the chest, (7) basketball shoot, (8) right hand draw X, (9) right hand draw circle (clockwise), (10) right hand draw circle (counter-clockwise), (11) draw triangle, (12) bowling (right hand), (13) front boxing, (14) baseball swing from right, (15) tennis right hand forehand swing, (16) arm curl (two arms), (17) tennis serve, (18) two hand push, (19) right hand knock on door, (20) right hand catch an object, (21) right hand pick up and throw, (22) jogging in place, (23) walking in place, (24) sit to stand, (25) stand to sit, (26) forward lunge (left foot forward), (27) squat (two arms stretch out). UTD-MHAD contains synchronized data from a Microsoft Kinect.
depth camera and an inertial sensor, where the Kinect device provided 3 modalities including depth videos, RGB videos, and skeleton positions. The inertial sensor was attached to the right wrist or right thigh according to the action. The University of Rzeszow fall detection (URFD) dataset [30] contains 2 action types: fall and activities of daily living (ADL). The fall data were collected from 2 Microsoft Kinect cameras and one 3-axis accelerometer, where the accelerometer was attached near the spine on the lower back. The ADLs included sitting, walking, lying, and crouching down in the fall detection method, and these daily activities contained depth and RGB images from one Microsoft Kinect camera.

III. MULTIVIEW CAUCHY ESTIMATOR FEATURE FUSION

In this section, we present our new feature fusion method for human action recognition, MCEFE, which can be divided into three parts: i.e., the multiview matrix approximation, the Cauchy estimator, and the ensemble manifold regularization.

A. Unified space approximation

Suppose there are \( V - 1 \) inertial sensors attached to a human body part, one depth sensor, and features are extracted from these \( V \) sensors. We consider the training set with \( V \) views \( D = \{ X^i | 1 \leq i \leq V \} \), where \( X^i = [x^i_1, x^i_2, ..., x^i_n] \in R^{D_i \times n} \) are \( n \) training samples and \( D_i \) is the feature size from the \( i \)-th view. A straightforward approach to find the unified space is by minimizing the empirical risk. Hence, the reconstruction error over the unified space \( Y \in R^d \times n \) can be measured using the \( L_2 \) loss as follows:

\[
\arg \min_{W,Y} \frac{1}{V} \sum_{i=1}^{V} \| X^i - W_i Y \|^2, \tag{1}
\]

where \( W_i \in R^{D_i \times d} \) is the \( i \)-th view projection matrix and \( W = \{ W_i | 1 \leq i \leq V \} \).

B. Cauchy estimators

While the objective formulation in (1) finds the unified space approximation and the projection matrices, the noise in different views adversely influences the obtained unified space approximation and the projection matrices because the noise from different views satisfies different distributions. Furthermore, as studied in robust statistics [36], the \( L_2 \) loss is not robust to outliers; the performance of unified space approximation will, therefore, be seriously degraded.

The Cauchy estimator \( \rho(x) = \log(1 + \frac{x^2}{c^2}) \) has a valuable property, namely that the influence of a single observation does not yield a significant offset [24]. In addition, half of the observations can be incorrect before the Cauchy estimator gives an incorrect result [37]. Hence, the Cauchy estimator can be deployed to replace the \( L_2 \) loss in (1)

\[
\arg \min_{W,Y} \frac{1}{V} \sum_{i=1}^{V} \log(1 + \frac{\| X^i - W_i Y \|^2}{c^2}), \tag{2}
\]

where \( c \) is a constant scale parameter.

C. Ensemble manifold regularization

To avoid overfitting, a regularization technique should be adopted on the projection matrices. Since different views have different statistical properties and physical meanings, the desired projection matrices should encode the correlations between different views by fully accounting for complementary properties. Inspired by ensemble manifold regularization [38], which suggests that the intrinsic manifold can be approximated by the optimal linear combination of candidate manifolds, we impose the weights \( \alpha = [\alpha_1, \alpha_2, ..., \alpha_V] (\alpha > 0 \text{ and } \sum_{i=1}^{V} \alpha_i = 1) \) on the projection matrices and combine (2) to give:

\[
\arg \min_{W,Y,\alpha} \frac{1}{V} \sum_{i=1}^{V} \log(1 + \frac{\| X^i - W_i Y \|^2}{c^2}) + \gamma \sum_{i=1}^{V} \alpha_i \| W_i \|^2, \tag{3}
\]

where \( \gamma \) is a non-negative trade-off parameter that can be determined using cross validation.

Overall, (3) jointly models the relationships between each projection matrix \( W_i \) and the unified space \( Y \).

D. Optimization

The objective function (3) is a non-convex and multivariable problem, and there is no optimal method that allows for the simultaneous optimization of all the variables. To address this problem, we can decompose it into three subproblems over the projection matrices \( W \), the unified space \( Y \), and the weights \( \alpha \) using alternating optimization. Specifically, we iteratively optimize one of the variables \((W,Y,\alpha)\) by fixing the other two.

1) Optimize \( Y \): Given fixed projection matrices \( W \) and weights \( \alpha \), (3) can be minimized over unified space \( Y \),

\[
\min_{Y} \mathcal{L} = \frac{1}{V} \sum_{i=1}^{V} \log(1 + \frac{\| X^i - W_i Y \|^2}{c^2}). \tag{4}
\]

Setting the gradient of \( \mathcal{L} \) with respect to \( Y \) to 0, we have

\[
\frac{1}{V} \sum_{i=1}^{V} \frac{2 W_i^T (X^i - W_i Y)}{c^2 + \| X^i - W_i Y \|^2} = 0, \tag{5}
\]

which can be rewritten as

\[
\sum_{i=1}^{V} (\frac{2 W_i^T W_i}{c^2 + \| X^i - W_i Y \|^2}) Y = \frac{V}{c^2 + \| X - W Y \|^2}. \tag{6}
\]

We denote \( r^i = X^i - W_i Y \), and \( r^i \) is referred to as the residual of the unified space \( Y \) on each view. A weight vector function is defined as

\[
h = \left[ \frac{1}{c^2 + \| r^2 \|^2}, \frac{1}{c^2 + \| r^2 \|^2}, ..., \frac{1}{c^2 + \| r^2 \|^2} \right]. \tag{7}
\]

which can be used to adjust the errors employed by different views and reduce the influence of noise.

Combining (6) and (7), we have

\[
Y = \left( \sum_{i=1}^{V} 2 h_i W_i^T W_i \right)^{-1} \sum_{i=1}^{V} h_i W_i^T X^i. \tag{8}
\]
Combining (11) and (12), we can update the projection matrix shown in Algorithm 1.

Considering that \( h \) depends on \( Y \), and motivated by the generalized Weiszfeld’s method \cite{24}, we iteratively update \( Y \) using (8) with a random initial value until convergence, which is similar to \cite{24}. The procedure of iteratively estimating \( Y \) is shown in Algorithm 1.

\begin{algorithm}
\caption{Iteratively Estimate \( Y \)}
\begin{algorithmic}
\Require Multiview feature matrix \( D = \{X^i|1 \leq i \leq V\} \), projection matrix \( W \), and initial unified space estimation \( Y^0 \).
\Ensure Converged \( Y^t \)
\State Set \( t = 1 \)
\Repeat
\State Compute weight vector function \( h \) through (7)
\State Compute (8) to update the estimate \( Y^t \)
\State Update the residuals \( \{r^t_i\}_{i=1}^V \)
\State update \( t = t + 1 \)
\Until the estimate \( Y \) convergence
\end{algorithmic}
\end{algorithm}

To optimize \( W \): Given fixed unified space \( Y \) and weights \( \alpha \), (3) can be minimized over each view projection matrix \( \{W^i\}_{i=1}^V \).

\[ \min_{W^i} L = \log(1 + \frac{||X^i - W^i Y||^2}{c^2}) + \gamma(\alpha)^r||W^i||^2. \]  

(9)

Setting the gradient of \( L \) with respect to \( W^i \) to 0, we have

\[ \frac{2\gamma}{c^2 + ||X^i - W^i Y||^2} 2Y^T c^2 + ||X^i - W^i Y||^2  + 2\gamma(\alpha)^rW^i = 0, \]  

(10)

which can be rewritten as

\[ W^i \left( \frac{2Y^T}{c^2 + ||X^i - W^i Y||^2} 2Y^T c^2 + ||X^i - W^i Y||^2 \right) = \frac{2X^i Y^T}{c^2 + ||X^i - W^i Y||^2} \]  

(11)

Given the residual \( r^i = x^i - W^i Y \), and the weight vector function

\[ h = \left[ \frac{1}{c^2 + ||r^1||^2}, \frac{1}{c^2 + ||r^2||^2}, \ldots, \frac{1}{c^2 + ||r^V||^2} \right], \]  

(12)

combining (11) and (12), we can update the projection matrix \( W^i \) by

\[ W^i = h^i X^i Y^T \left( h^i Y Y^T + \gamma(\alpha)^r \right)^{-1}. \]  

(13)

Similar to the optimization over the unified space \( Y \), the procedure of iteratively estimating \( \{W^i\}_{i=1}^V \) is shown in Algorithm 2.

\begin{algorithm}
\caption{Iteratively Estimate \( \{W^i\}_{i=1}^V \)}
\begin{algorithmic}
\Require Multiview feature matrix \( D = \{X^i|1 \leq i \leq V\} \), the tradeoff parameter \( r \), the weights \( \alpha^0 \), the scale parameter \( c \), and the unified space \( Y \).
\Ensure Converged \( \{W^i\}_{i=1}^V \)
\State Set \( i = 1 \)
\Repeat
\State Compute weight vector function \( h \) through (12)
\State Compute (13) to update the estimate \( W^i \)
\State Update the residuals \( \{r^t_i\}_{i=1}^V \)
\State Update \( t = t + 1 \)
\Until the estimate \( W^i \) convergence
\State Update \( i = i + 1 \)
\Until \( |O_{t+1} - O_t| < \xi \)
\end{algorithmic}
\end{algorithm}

3) Optimize \( \alpha \): Given fixed unified space \( Y \) and projection matrix \( \{W^i\}_{i=1}^V \), (3) can be minimized over weights \( \alpha \),\n
\[ \arg \min_{\alpha} \sum_{i=1}^V (\alpha)^r||W^i||^2, s.t. \sum_{i=1}^V \alpha_i = 1, \alpha > 0. \]  

(14)

The Lagrangian function of (14) is

\[ L(\alpha, \eta) = \sum_{i=1}^V (\alpha)^r||W^i||^2 - \eta(\sum_{i=1}^V \alpha_i - 1), \]  

(15)

where \( \eta \) is the Lagrangian multiplier. Setting the gradient of \( L(\alpha, \eta) \) with respect to \( \alpha^r \) to 0, we have

\[ r(\alpha)^r - 1 ||W^i||^2 - \eta = 0, \]  

(16)

which can be rewritten as

\[ \alpha_i = \left( \frac{\eta}{r||W^i||^2} \right)^{1/(r-1)}. \]  

(17)

By considering the constraint \( \sum_{i=1}^V \alpha_i = 1 \) into (17), we have

\[ \alpha_i = \frac{(r||W^i||^2)^{1/(r-1)}}{\sum_{i=1}^V (r||W^i||^2)^{1/(r-1)}}. \]  

(18)

The proposed efficient MCEFE optimization procedure is summarized in Algorithm 3.

Given a test sample with \( V \) views \( \{t^i\}_{i=1}^V \), the corresponding data \( y \) in the unified space can be obtained by solving the following problem

\[ \min_{y} L = \frac{1}{V} \sum_{i=1}^V \log(1 + ||t^i - W^i y||^2). \]  

(19)

where \( \{W^i\}_{i=1}^V \) are the optimal projection matrices obtained from Algorithm 3.

\begin{algorithm}
\caption{Efficient MCEFE optimization algorithm}
\begin{algorithmic}
\Require Multiview feature matrix \( D = \{X^i|1 \leq i \leq V\} \), the tradeoff parameter \( r \), the initial weights \( \alpha^0 \), the scale parameter \( c \), the initial projection matrices \( W^0 \), and the unified space \( Y^0 \), and the threshold \( \xi \).
\Ensure Converged \( \{W^i\}_{i=1}^V \) and \( Y^t \)
\State Set \( t = 1 \)
\Repeat
\State Update the estimate \( Y^t \) by algorithm 1
\State Update the estimate \( \{W^i\}_{i=1}^V \) by algorithm 2
\State Update the weights \( \alpha^t \) by (18)
\State Compute \( O_t \) by (3)
\State Update \( t = t + 1 \)
\Until \( |O_{t+1} - O_t| < \xi \)
\end{algorithmic}
\end{algorithm}

IV. EXPERIMENTS

Given that the actions in existing multimodal human action databases are relative fixed (that is, subjects are generally on
Fig. 1. Body placement of the 17 inertial sensors in CAS-YNU-MHAD.

the same spot when performing movements), we collected human action samples to create CAS-YNU-MHAD by utilizing one Kinect camera and 17 inertial sensors. For each sample, the FFT coefficients and the STIPs are extracted from the Kinect camera and inertial sensors to represent the raw data. MCEFE performance is evaluated using the average recognition rates for each human action. Confusion matrices are also reported to better understand the data. Further details of the dataset, experimental setup, and baseline methods are given below.

A. Dataset

One Kinect camera and 17 inertial sensors were used in CAS-YNU-MHAD. The Kinect camera was placed in front of the individual and the 17 inertial sensor locations and corresponding accelerometers were hips (A1), right up leg (A2), right leg (A3), right foot (A4), left up leg (A5), left leg (A6), left foot (A7), right shoulder (A8), right arm (A9), right forearm (A10), right hand (A11), left shoulder (A12), left arm (A13), left forearm (A14), left hand (A15), head (A16), and spine (A17) (The number in parentheses is the corresponding accelerometer name). Fig. 1 shows where the 17 inertial sensors were placed in the body. Both types of sensor are low cost, widely available, computationally inexpensive, and provided relatively easy manipulation of the generated samples. The Kinect camera can capture a 16-bit depth image with a resolution of 512 × 424 pixels and an RGB image with a resolution of 968 × 544 pixels. Moreover, the frame rate is about 25 frames per second.

CAS-YNU-MHAD contains 10 actions performed by 10 subjects between 20 and 30 years of age. All subjects performed about 10 repetitions of each action, yielding about 1086 action sequences. These human actions and the corresponding definition are shown in Table I. where typing, sitting down, standing up, and lying down are light intensity actions, walking, walking quickly, and walking S are moderate intensity actions, and jumping forward, jumping up, and running are vigorous actions. In most actions, the subject moves from far away to near. Fig. 2 shows example 3D-acceleration data, depth images (with background removed), depth images, and RGB images. CAS-YNU-MHAD is, therefore, extremely useful for human action research.

B. Feature extraction and parameter setting

For each 3D-acceleration data point generated from inertial sensors, FFT features were extracted according to the following procedure: 1) each 3D-acceleration data point was processed by subtracting its mean and dividing by its standard deviation. A moving average filter of length 5 was then imposed on each axis; 2) FFT coefficients were extracted corresponding to each axis using a 250-point window with a 125-point overlap between consecutive windows; 3) the first 64 FFT coefficients were retained for each sliding window; and 4) all the obtained FFT coefficients were concatenated as the FFT feature to represent a human action. Note that the 3D-acceleration data on each axis contains 1024 sample points and we can obtain 7 sampling windows and 7 × 64 × 3 = 1344-dimensional coefficients. For depth images generated from the Microsoft Kinect camera, we first removed the background of each depth image and then extracted STCP features to represent the actions; the dimensionality of the extracted STCP features was 81648. Since the FFT features and the STCP features were of high dimensionality, we used PCA for feature preprocessing to avoid the dimensional curve problem and speed up computation.

C. Baselines and performance measures

We compared MCEFE’s effectiveness with four representative methods: PCA [40], LPP [41], LDA [42], and DLA [43]. Specifically, multiview features were first concatenated to a long feature before projecting the obtained long feature to find the optimal space. These methods represent state-of-the-art embedding algorithms that have proven advantages in many practical problems. PCA and LPP are classical unsupervised methods, while LDA and DLA are representative supervised methods. Furthermore, a baseline was also provided by directly projecting the obtained long feature on the classifier.

Inspired by leave-one-out cross-validation, samples of one subject from CAS-YNU-MHAD were used for testing, while the remaining samples were used as training data. The training set was used to learn the projection matrices and the test data were used to evaluate robustness of MCEFE for
Fig. 2. Example 3D-acceleration data, depth images (the background was removed), depth images, and RGB images (from top to bottom). There are two paired samples in this figure and each pair corresponding to one human action, i.e., Jumping forward and Walking.
human action recognition. Therefore, CAS-YNU-MHAD was divided into 10 splits. To unify the classification stage, the support vector machine (SVM) [44] classifier was employed for compatibility with the various embedding methods. In our experiments, LIBSVM [45] was used for classification.

D. Experimental results and analysis

In order to find the optimal sensor locations, the recognition rate of depth and inertial sensors at different sensor locations were analyzed, where D refers to the depth sensor (see Table II). To examine whether the simultaneous use of 3D accelerometer signals and depth images was an improvement over the use of the accelerometers or Kinect camera alone, the recognition rates of MCEFE on multimodal sensors were analyzed (Table III). Using the two inertial sensors A8 and A14 in the fusion approach did not result in an improvement in recognition over using a single inertial sensor A8 or A14. However, the overall recognition rate was improved compared to using the depth sensor or the inertial sensor alone, and the overall recognition rate was further improved using the depth sensor with several inertial sensors. For example, it can be seen from Table III that the ideal inertial sensor location was the spine (A17). The depth sensor had lower recognition accuracies compared to inertial sensors and it failed to recognize

TABLE II
CLASSIFICATION RESULTS ON DEPTH SENSOR AND INERTIAL SENSORS WITH DIFFERENT SENSOR LOCATION RESPECTIVELY.

| Classes          | Depth(D) | Head(A16) | Left up leg(A5) | Left foot(A7) | Right leg(A3) | Right arm(A9) | Right foot(A4) | Left leg(A6) | Right hand(A11) |
|------------------|----------|-----------|-----------------|---------------|---------------|---------------|----------------|---------------|-----------------|
| jumping forward  | 0.9455   | 0.9355    | 0.9168          | 0.8089        | 0.9709        |                |                |               |                 |
| jumping up       | 0.94     | 0.9118    | 0.9226          | 0.8089        | 0.9118        |                |                |               |                 |
| lying down       | 0.9035   | 0.8495    | 0.8833          | 0.89          | 0.9161        |                |                |               |                 |
| running          | 0.7618   | 0.82      | 0.85            | 0.8809        | 0.8118        |                |                |               | 0.9227          |
| sitting down     | 0.9136   | 0.8711    | 0.7744          | 0.6424        | 0.8548        |                |                |               | 0.7148          |
| standing up      | 0.9253   | 0.7942    | 0.7342          | 0.6598        | 0.6745        |                |                |               | 0.7361          |
| typing           | 0.7222   | 0.9592    | 0.9621          | 0.9369        | 0.9866        |                |                |               | 0.9007          |
| walking          | 0.8367   | 0.875     | 0.7783          | 0.8871        | 0.925         |                |                |               | 0.8057          |
| walking quickly  | 0.9678   | 0.9763    | 0.9755          | 0.9475        | 0.9753        |                |                |               | 0.9755          |
| walking S        | 0.2208   | 0.83      | 0.83            | 0.83          | 0.835         |                |                |               | 0.7617          |
| Average accuracy | 0.7945   | 0.8729    | 0.8604          | 0.8325        | 0.9008        |                |                |               | 0.8851          |

TABLE III
CLASSIFICATION RESULTS ON THE CAS-YNU-MHAD WITH SERVERAL SENSORS.

| Classifier      | A8,A14 | D,A8 | D,A14 | D,A8,A14 | D,A11 | D,A17 | D,A11,A17 | D,A8,A14,A15,A17 |
|-----------------|--------|------|-------|----------|-------|-------|-----------|------------------|
| jumping forward | 0.9709 | 0.9545 | 0.9545 | 0.9545 | 0.9545 | 0.9545 | 0.9545 | 0.9636           |
| jumping up      | 0.9618 | 0.95 | 0.95 | 0.95 | 0.95 | 0.95 | 0.95 | 0.95 | 0.95 | 0.9636 |
| lying down      | 0.9055 | 0.99 | 0.99 | 0.99 | 0.99 | 0.99 | 0.99 | 0.99 | 0.99 | 0.99 |
| running         | 0.99 | 0.9218 | 0.9409 | 0.96 | 0.9509 | 0.9318 | 0.96 | 0.97 | 1 |
| sitting down    | 0.6979 | 0.9318 | 0.9418 | 0.9409 | 0.9409 | 0.9409 | 0.9409 | 0.9409 | 0.965 |
| standing up     | 0.9696 | 0.98 | 0.98 | 0.98 | 0.98 | 0.98 | 0.98 | 0.98 |
| typing          | 0.9763 | 0.97 | 0.97 | 0.97 | 0.97 | 0.97 | 0.97 | 0.97 | 1 |
| walking         | 0.9283 | 0.8767 | 0.7917 | 0.9417 | 0.84 | 0.895 | 0.895 | 0.895 | 0.867 | 0.965 |
| walking quickly | 0.9023 | 0.9923 | 0.9923 | 0.9846 | 0.9923 | 0.9769 | 0.9769 | 0.9769 | 0.9769 | 1 |
| walking S       | 0.875 | 0.7617 | 0.7567 | 0.8433 | 0.795 | 0.7333 | 0.8433 | 0.8433 | 0.875 | 1 |
| average recognition | 0.8891 | 0.9404 | 0.9292 | 0.9591 | 0.9404 | 0.9354 | 0.952 | 0.9676 |

average recognition | 0.8891 | 0.9404 | 0.9292 | 0.9591 | 0.9404 | 0.9354 | 0.952 | 0.9676 |
TABLE IV

| Classifier | MCEFE | PCA | LPP | LDA | DLA | baseline |
|------------|-------|-----|-----|-----|-----|---------|
| Average recognition | 0.9794(400) | 0.9279(50) | 0.8714(150) | 0.9654(200) | 0.9659(200) | 0.8683  |

walking S and typing. The inertial sensor on the left forearm (A14) produced higher recognition accuracies with walking S, and an inertial sensor on the right shoulder (A8) produced higher recognition accuracies for typing. Hence, the recognition accuracy of (D,A14) and (D,A8) was an improvement over single modality sensors D, A8, and A14. (D,A8,D14) further improved over (D,A14) and (D,A8), since A14 or A8 provided additional useful information to distinguish actions. (D,A8,D14) further improved over (D,A14) and (D,A8), since A14 or A8 provided additional useful information to distinguish actions.

MCEFE was next compared with MCEFE, PCA, LPP, LDA, DLA, and baseline on CAS-YNU-MHAD (Fig. 3). The x-coordinate represents the number of embedded dimensions, and the y-coordinate represents the corresponding average accuracy. The best average recognition rates of the five methods, baseline, and the corresponding dimension (the number in parentheses is the embedding dimension) are shown in Table IV. LDA and DLA are promising solutions and significantly outperformed PCA and LPP because they are supervised methods that utilize label information. MCEFE, although an unsupervised method, outperformed the others, because it finds the optimal unified space and is robust to outliers, which is important for improving human action recognition performance.

Fig. 3, Fig. 6, and Fig. 7 show three confusion matrices corresponding to using depth sensor alone, inertial sensor A11 alone, and depth sensor and inertial sensor A11 fused. Fig. 3 shows that the misclassifications occurred between typing and lying down, walking S, and walking. Fig. 6 shows that the misclassifications occurred between sitting down and standing up. By integrating the depth sensor and inertial sensor A11, the fused features extracted from MCEFE were more effective and improved recognition accuracies over the depth sensor or inertial sensor A11 alone. Fig. 7 shows that the recognition rates when using one modality sensor improved when the depth sensor and the inertial sensor were used together. This is due to the complementary nature of the data generated from the depth sensor and the inertial sensor. More importantly, the confused actions shown in Fig. 3 and Fig. 6 could be separated correctly (Fig. 7). For example, the recognition rate for typing and walking S was improved by 52% and 58% respectively over the depth sensor alone, and the recognition rate for sitting down and standing up was improved by 26% and 24% respectively over the inertial sensor alone.

The convergence curves of MCEFE are presented in Fig. 4 to discover different dimensional unified spaces. The error rate quickly decreased after only a few rounds of iteration (less than 5) and became stable within 10 iterations. Higher dimensional unified spaces led to even faster convergence. Thus, MCEFE is effective and converges rapidly.

V. CONCLUSION

In this paper, a novel robust multiview embedding method - termed MCEFE - is presented for depth and inertial sensor-based human action recognition. MCEFE finds the optimal unified space and projection matrices by minimizing empirical risk through the Cauchy estimator and enforcing ensemble manifold regularization on the projection matrices. To further facilitate depth and inertial sensor fusion for human action recognition research and demonstrate the effectiveness and robustness of MCEFE, a multimodal human action database (CAS-YNU-MHAD) is presented. Data for CAS-YNU-MHAD was collected from a Kinect camera and 17 inertial sensors and most actions of CAS-YNU-MHAD do the movements from far away to near.
jumping forward
jumping up
lying down
running
sitting down
standing up
typing
walking
walking quickly
walking S

Jumping forward
jumping up
lying down
running
sitting down
standing up
typing
walking
walking quickly
walking S

Jumping forward
jumping up
lying down
running
sitting down
standing up
typing
walking
walking quickly
walking S

Jumping forward
jumping up
lying down
running
sitting down
standing up
typing
walking
walking quickly
walking S

Jumping forward
jumping up
lying down
running
sitting down
standing up
typing
walking
walking quickly
walking S

Compared to the classical embedding methods such as PCA, LPP, LDA, and DLA, MCEFE shows many attractive properties that help distinguish different actions. MCEFE is superior to other embedding methods in terms of recognition accuracy for depth and inertial sensor-based human action recognition.

REFERENCES

[1] J. Zheng, Z. Jiang, and R. Chellappa, “Cross-view action recognition via transferable dictionary learning,” IEEE Transactions on Image Processing, vol. 25, no. 6, pp. 2542–2556, 2016.

[2] W. Ross, A. Gorod, and M. Uliieru, “A socio-physical approach to systemic risk reduction in emergency response and preparedness,” IEEE Transactions on Systems, Man, and Cybernetics: Systems, vol. 45, no. 8, pp. 1125–1137, 2015.

[3] J. L. Coyle, D. Stirling, H. Du, and M. Ros, “Seated whole-body vibration analysis, technologies, and modeling: A survey,” IEEE Transactions on Systems, Man, and Cybernetics: Systems, vol. 46, no. 6, pp. 725–739, 2016.

[4] D. Tosato, M. Spera, M. Cristani, and V. Murino, “Characterizing humans on riemannian manifolds,” IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 35, no. 8, pp. 1972–1984, 2013.

[5] A. Jalal, M. Z. Uddin, and T.-S. Kim, “Depth video-based human activity recognition system using translation and scaling invariant features for life logging at smart home,” IEEE Transactions on Consumer Electronics, vol. 58, no. 3, pp. 863–871, 2012.

[6] M. Tenatori and J. Favela, “Activity-aware computing for healthcare,” Pervasive Computing, vol. 7, no. 2, pp. 51–57, 2008.

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

[8] F. Santoso, “A new framework for rapid wireless tracking verifications based on optimized trajectories in received signal strength measurements,” IEEE Transactions on Systems, Man, and Cybernetics: Systems, vol. 45, no. 11, pp. 1424–1436, 2015.

[9] X. Ji, J. Cheng, and W. Feng, “Spatio-temporal cuboid pyramid for action recognition using depth motion sequences,” in Proceedings of the International Conference on Advanced Computational Intelligence, pp. 208–213, IEEE, 2016.

[10] J. Du, C. Mouser, and W. Sheng, “Design and evaluation of a teleoperated robotic 3-d mapping system using an rgb-d sensor,” IEEE Transactions on Systems, Man, and Cybernetics: Systems, vol. 46, no. 5, pp. 718–724, 2016.

[11] P. Gupta and T. Dallas, “Feature selection and activity recognition system using a single triaxial accelerometer,” IEEE Transactions on Biomedical Engineering, vol. 61, no. 6, pp. 1780–1786, 2014.

[12] L. Atallah, B. Lo, R. King, and G.-Z. Yang, “Sensor placement for activity detection using wearable accelerometers,” in Proceedings of the International Con-
ference on Body Sensor Networks, pp. 24–29, IEEE, 2010.

[13] S. Wan and J. Aggarwal, “Mining discriminative states of hands and objects to recognize egocentric actions with a wearable rgbd camera,” in Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops, pp. 36–43, 2015.

[14] M. Ermes, J. Parkka, J. Mantyjarvi, and I. Korhonen, “Detection of daily activities and sports with wearable sensors in controlled and uncontrolled conditions,” IEEE Transactions on Information Technology in Biomedicine, vol. 12, no. 1, pp. 20–26, 2008.

[15] A. M. Khan, Y.-K. Lee, S. Y. Lee, and T.-S. Kim, “A triaxial accelerometer-based physical-activity recognition via augmented-signal features and a hierarchical recognizer,” IEEE Transactions on Information Technology in Biomedicine, vol. 14, no. 5, pp. 1166–1172, 2010.

[16] C. Chen, R. Jafari, and N. Kehtarnavaz, “Improving human action recognition using fusion of depth camera and inertial sensors,” IEEE Transactions on Human-Machine Systems, vol. 45, no. 1, pp. 51–61, 2015.

[17] Y. Xue and L. Jin, “A naturalistic 3d acceleration-based activity dataset & benchmark evaluations,” in Proceedings of the IEEE International Conference on Systems Man and Cybernetics, pp. 4081–4085, IEEE, 2010.

[18] D. Tao, L. Jin, Y. Yuan, and Y. Xue, “Ensemble manifold rank preserving for acceleration-based human activity recognition,” IEEE Transactions on Neural Networks and Learning Systems, vol. 27, no. 6, pp. 1392–1404, 2016.

[19] K. Huang, C. Wang, and D. Tao, “High-order topology modeling of visual words for image classification,” IEEE Transactions on Image Processing, vol. 24, no. 11, pp. 3598–3608, 2015.

[20] A. F. Bobick and J. W. Davis, “The recognition of human movement using temporal templates,” IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 23, no. 3, pp. 257–267, 2001.

[21] I. Laptev, “On space-time interest points,” International Journal of Computer Vision, vol. 64, no. 2-3, pp. 107–123, 2005.

[22] X. Yang, C. Zhang, and Y. Tian, “Recognizing actions using depth motion maps-based histograms of oriented gradients,” in Proceedings of the ACM international conference on Multimedia, pp. 1057–1060, ACM, 2012.

[23] L. Xia and J. Aggarwal, “Spatio-temporal depth cuboid similarity feature for activity recognition using depth camera,” in Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 2834–2841, 2013.

[24] C. Xu, D. Tao, and C. Xu, “Multi-view intact space learning,” IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 37, no. 12, pp. 2531–2544, 2015.

[25] A. Shrivastava, V. M. Patel, J. K. Pillai, and R. Chellappa, “Generalized dictionaries for multiple instance learning,” International Journal of Computer Vision, vol. 114, no. 2-3, pp. 288–305, 2015.

[26] W. Hu, W. Li, X. Zhang, and S. Maybank, “Single and multiple object tracking using a multi-feature joint sparse representation,” IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 37, no. 4, pp. 816–833, 2015.

[27] W. Ren, K. Huang, D. Tao, and T. Tan, “Weakly supervised large scale object localization with multiple instance learning and bag splitting,” IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 38, no. 2, pp. 405–416, 2016.

[28] F. Oflı, R. Chaudhry, G. Kurillo, R. Vidal, and R. Bajcsy, “Berkeley mhad: A comprehensive multimodal human action database,” in Proceedings of the IEEE Workshop on Applications of Computer Vision, pp. 53–60, IEEE, 2013.

[29] C. Chen, R. Jafari, and N. Kehtarnavaz, “Utd-mhad: a multimodal dataset for human action recognition utilizing a depth camera and a wearable inertial sensor,” in Proceedings of the IEEE International Conference on Image Processing, pp. 168–172, IEEE, 2015.

[30] B. Kwolek and M. Kepski, “Human fall detection on embedded platform using depth maps and wireless accelerometer,” Computer methods and programs in biomedicine, vol. 117, no. 3, pp. 489–501, 2014.

[31] K. Liu, C. Chen, R. Jafari, and N. Kehtarnavaz, “Fusion of inertial and depth sensor data for robust hand gesture recognition,” IEEE Sensors Journal, vol. 14, no. 6, pp. 1898–1903, 2014.

[32] C. Cao, Y. Zhang, and H. Lu, “Multi-modal learning for gesture recognition,” in Proceedings of the IEEE International Conference on Multimedia and Expo, pp. 1–6, IEEE, 2015.

[33] J. Wu and J. Cheng, “Bayesian co-boosting for multi-modal gesture recognition,” The Journal of Machine Learning Research, vol. 15, no. 1, pp. 3013–3036, 2014.

[34] P. Gehler and S. Nowozin, “On feature combination for multiclass object classification,” in Proceedings of the IEEE International Conference on Computer Vision, pp. 221–228, IEEE, 2009.

[35] G. Shafer et al., A mathematical theory of evidence, vol. 1. Princeton university press Princeton, 1976.

[36] W. Rey, Introduction to robust and quasi-robust statistical methods. Springer Science & Business Media, 2012.

[37] I. Mizera and C. H. Müller, “Breakdown points of cauchy regression-scale estimators,” Statistics & probability letters, vol. 57, no. 1, pp. 79–89, 2002.

[38] B. Geng, D. Tao, C. Xu, L. Yang, and X.-S. Hua, “Ensemble manifold regularization,” IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 34, no. 6, pp. 1227–1233, 2012.

[39] H. Vöß and U. Eckhardt, “Linear convergence of generalized weiszfeld’s method,” Computing, vol. 25, no. 3, pp. 243–251, 1980.

[40] H. Hotelling, “Analysis of a complex of statistical variables into principal components,” Journal of Educational Psychology, vol. 24, no. 6, p. 417, 1933.

[41] X. Niyogi, “Locality preserving projections,” in Neural Information Processing Systems, vol. 16, p. 153, MIT, 2004.
[42] R. A. Fisher, “The use of multiple measurements in taxonomic problems,” *Annals of Eugenics*, vol. 7, no. 2, pp. 179–188, 1936.

[43] T. Zhang, D. Tao, X. Li, and J. Yang, “Patch alignment for dimensionality reduction,” *IEEE Transactions on Knowledge and Data Engineering*, vol. 21, no. 9, pp. 1299–1313, 2009.

[44] V. N. Vapnik and V. Vapnik, *Statistical learning theory*, vol. 1. Wiley New York, 1998.

[45] C.-C. Chang and C.-J. Lin, “Libsvm: a library for support vector machines,” *ACM Transactions on Intelligent Systems and Technology*, vol. 2, no. 3, p. 27, 2011.