# Improving Human Action Recognition Using Fusion of Depth Camera and Inertial Sensors

Chen Chen, *Student Member, IEEE*, Roozbeh Jafari, *Senior Member, IEEE*, and Nasser Kehtarnavaz, *Fellow, IEEE*

***Abstract*—This paper presents a fusion approach for improving human action recognition based on two differing modality sensors consisting of a depth camera and an inertial body sensor. Computationally efficient action features are extracted from depth images provided by the depth camera and from accelerometer signals provided by the inertial body sensor. These features consist of depth motion maps and statistical signal attributes. For action recognition, both feature-level fusion and decision-level fusion are examined by using a collaborative representation classifier. In the feature-level fusion, features generated from the two differing modality sensors are merged before classification, while in the decision-level fusion, the Dempster–Shafer theory is used to combine the classification outcomes from two classifiers, each corresponding to one sensor. The introduced fusion framework is evaluated using the Berkeley multimodal human action database. The results indicate that because of the complementary aspect of the data from these sensors, the introduced fusion approaches lead to 2% to 23% recognition rate improvements depending on the action over the situations when each sensor is used individually.**

***Index Terms*—Depth motion map (DMM), fusion of depth camera and inertial sensor, human action recognition, wearable inertial sensor.**

## I. Introduction

Human action recognition is used in human–computer interaction (HCI) applications, including gaming, sports annotation, content-based video retrieval, health monitoring, visual surveillance, and robotics. For example, game consoles such as Nintendo Wii or Microsoft Kinect rely on the recognition of gestures or full-body movements for gaming interactions. Human action recognition is also a part of fitness training and rehabilitation, e.g., [1], [2]. Some human action recognition approaches are based on a depth camera or wearable inertial sensors, e.g., [3]–[5].

Since the release of Microsoft Kinect depth cameras, research has been conducted regarding human action recognition using them. Depth images generated by a structured light depth sensor, in particular the Kinect depth camera, are insensitive to changes in lighting conditions and provide 3-D information toward distinguishing actions that are difficult to characterize using intensity images. For example, an action graph was employed in [6] to model the dynamics of actions, and a collection of 3-D points from depth images was used to characterize postures. In [7], a depth motion map (DMM)-based histogram of oriented gradients was utilized to compactly represent body shape and movement information followed by a linear support vector machine (SVM) to recognize human actions. In [8], the so-called random occupancy pattern features were extracted from depth images using a weighted sampling scheme and used for action recognition. In [9], a 4-D histogram overdepth, time, and spatial coordinates were used to encode the distribution of the surface normal orientation, which was then used for action recognition. In [10], a filtering method extracted the spatiotemporal interest points, followed by a depth cuboid similarity feature for action recognition.

Several action recognition systems involve wearable inertial sensors. For example, in [5], wearable inertial sensors were employed to recognize daily activities and sports in unsupervised settings by using artificial neural networks within a tree structure. In [11], a sparse representation classifier (SRC) was introduced for human daily activity modeling and recognition using a single-wearable inertial sensor. In [12], a hierarchical-recognition scheme was proposed to extract features based on linear discriminant analysis from a single triaxial accelerometer. Artificial neural networks were then used for human activity classification. In [13], a wireless body area network composed of multiple wearable inertial sensors monitored position and activity of upper and lower extremities for computer-assisted physical rehabilitation. In [14], a fall detection system was presented based on wearable inertial sensors.

Depth sensors and wearable inertial sensors have been used individually for human action recognition. However, simultaneous utilization of both depth and wearable inertial sensors for human action recognition are less common [15]–[18]. In [15], an inertial sensor and a Kinect were used to monitor a person's intake gesture. The position and angular displacement of arm gestures captured by the Kinect and the acceleration of arm gestures captured by the inertial sensor were analyzed separately. No information was published about how the data from the two sensors were fused together to achieve more accurate monitoring. Moreover, the application involved intake gestures not human action recognition. In [16], a Kinect depth sensor and a sensor consisting of an accelerometer and a gyroscope were used together to detect falls using a fuzzy inference approach. More specifically, the acceleration data from the accelerometer, the angular velocity data from the gyroscope, and the center of gravity data of a moving person from the Kinect were used as inputs into a fuzzy inference module to generate alarms when falls occurred. However, in the paper, only one action (falling) was considered and no distinction between different actions was considered. In [17], a Kinect depth sensor and five three-axis

Manuscript received February 2, 2014; revised September 8, 2014; accepted October 5, 2014. Date of publication October 23, 2014; date of current version January 13, 2015. This paper was recommended by Associate Editor H. Liu.

The authors are with Department of Electrical Engineering, University of Texas at Dallas, Richardson, TX 75080 USA (e-mail: chenchen870713@gmail.com; rjafari@utdallas.edu; kehtar@utdallas.edu).

Color versions of one or more of the figures in this paper are available online at http://ieeexplore.ieee.org.

Digital Object Identifier 10.1109/THMS.2014.2362520

accelerometers were used for indoor activity recognition. The acceleration data from the accelerometers and the position data from the Kinect were merged as the input to an ensemble of binary neural network classifiers. However, only feature-level fusion was performed, and the input signals to the classifiers were raw acceleration and position data without feature extraction. In [18], a Hidden Markov Model (HMM) classifier was used for hand gesture recognition with raw data from both a Kinect depth camera and an inertial body sensor (position data of the hand joint from a Kinect depth camera, as well as acceleration data and angular velocity data from an inertial body sensor). No feature extraction was conducted, and only feature-level fusion was used.

Depth and wearable inertial sensors are used to achieve improved human action recognition, compared with the sensors when used individually, while each of these sensors has its own limitations when operating under realistic conditions, utilizing them together provides synergy. In addition, our recognition solution is devised to be computationally efficient, so as to run in real time on desktop platforms.

In this paper, both feature-level and decision-level fusion are considered. The decision-level fusion is performed via the Dempster–Shafer theory. The introduced fusion approach is evaluated using a publicly available multimodal human action database (MHAD), the Berkeley MHAD [19]. Performance is compared in situations when using each modality sensor individually. Depth and wearable inertial sensors are low cost, easy to operate, and can be used in darkness. These attributes make their joint utilization practical in many HCI applications.

The rest of the paper is organized as follows. In Section II mathematical techniques used in our fusion approach are stated. In Section III, the MHAD is described. The details of our fusion approach are presented in Section IV. The results are reported in Section V. The conclusion is drawn in Section VI.

## II. Mathematical Techniques

### A. Sparse Representation Classifier

Sparse representation (or sparse coding) has received attention due to success in face recognition [20], [21]. The idea is to represent a test sample according to a small number of atoms sparsely chosen out of an overcomplete dictionary formed by all available training samples. Let us consider $C$ distinct classes and a matrix $\mathbf{X} = \{\mathbf{x}_i\}_{i=1}^{n} \in \mathbb{R}^{d \times n}$ formed by $n$ $d$-dimensional training samples arranged column wise to form the overcomplete dictionary. For a test sample $\mathbf{y} \in \mathbb{R}^d$, let us express $\mathbf{y}$ as a sparse representation in terms of matrix $\mathbf{X}$ as follows:

$$\mathbf{y} = \mathbf{X}\boldsymbol{\alpha} \tag{1}$$

where $\boldsymbol{\alpha}$ is a $n \times 1$ vector of coefficients corresponding to all training samples from the $C$ classes. One cannot directly solve for $\boldsymbol{\alpha}$, since (1) is typically underdetermined [21]. However, a solution can be obtained by solving the following $\ell_1$-regularized minimization problem:

$$\hat{\boldsymbol{\alpha}} = \underset{\alpha}{\operatorname{argmin}} \|\mathbf{y} - \mathbf{X}\boldsymbol{\alpha}\|_2^2 + \lambda\|\boldsymbol{\alpha}\|_1 \tag{2}$$

where $\lambda$ is a regularization parameter, which balances the influence of the residual and the sparsity term. According to the class labels of the training samples, $\hat{\boldsymbol{\alpha}}$ can be partitioned into $C$ subsets $\hat{\boldsymbol{\alpha}} = [\hat{\boldsymbol{\alpha}}_1, \hat{\boldsymbol{\alpha}}_2, \ldots, \hat{\boldsymbol{\alpha}}_C]$ with $\hat{\boldsymbol{\alpha}}_j (j \in 1, 2, \ldots, C)$ denoting the subset of the coefficients associated with the training samples from the $j$th class, i.e., $\mathbf{X}_j$. After coefficients partitioning, a class-specific representation $\widetilde{\mathbf{y}}_j$ is computed as follows:

$$\widetilde{\mathbf{y}}_j = \mathbf{X}_j \hat{\boldsymbol{\alpha}}_j. \tag{3}$$

The class label of $\mathbf{y}$ can be identified by comparing the closeness between $\mathbf{y}$ and $\widetilde{\mathbf{y}}_j$ via

$$\operatorname{class}(\mathbf{y}) = \underset{j \in \{1,2,\ldots,C\}}{\operatorname{argmin}} \; r_j(\mathbf{y}) \tag{4}$$

where $r_j(\mathbf{y}) = \|\mathbf{y} - \widetilde{\mathbf{y}}_j\|_2$ indicates the residual error.; see Algorithm 1.

**Algorithm 1** The SRC Algorithm

```
Input: Training samples $\mathbf{X} = \{\mathbf{x}_i\}_{i=1}^{n} \in \mathbb{R}^{d \times n}$, class label
$\omega_i$ (used for class partitioning), test sample $\mathbf{y} \in \mathbb{R}^d$, $\lambda$, $C$
(number of classes)
Calculate $\hat{\boldsymbol{\alpha}}$ via $\ell_1$-minimization of (2)
for all $j \in \{1, 2, \ldots, C\}$ do
    Partition $\mathbf{X}_j$, $\boldsymbol{\alpha}_j$
    Calculate $r_j(\mathbf{y}) = \|\mathbf{y} - \widetilde{\mathbf{y}}_j\|_2 = \|\mathbf{y} - \mathbf{X}_j\hat{\boldsymbol{\alpha}}_j\|_2$
end for
Decide class($\mathbf{y}$) via (4)
Output: class($\mathbf{y}$)
```

### B. Collaborative Representation Classifier

As suggested in [22], it is the collaborative representation, i.e., the use of all the training samples as a dictionary, but not the $\ell_1$-norm sparsity constraint, that improves classification accuracy. The $\ell_2$-regularization generates comparable results, but with significantly lower computational complexity [22]. The collaborative representation classifier (CRC) [22] swapped the $\ell_1$ penalty in (2) with an $\ell_2$ penalty, i.e.,

$$\hat{\boldsymbol{\alpha}} = \underset{\alpha}{\operatorname{argmin}} \|\mathbf{y} - \mathbf{X}\boldsymbol{\alpha}\|_2^2 + \theta\|\boldsymbol{\alpha}\|_2^2. \tag{5}$$

The $\ell_2$-regularized minimization of (5) is in the form of the Tikhonov regularization [23] leading to the following closed form solution:

$$\hat{\boldsymbol{\alpha}} = (\mathbf{X}^T\mathbf{X} + \theta\mathbf{I})^{-1}\mathbf{X}^T\mathbf{y} \tag{6}$$

where $\mathbf{I} \in \mathbb{R}^{n \times n}$ denotes an identity matrix. The general form of the Tikhonov regularization involves a Tikhonov regularization matrix $\Gamma$. As a result, (5) can be expressed as

$$\hat{\boldsymbol{\alpha}} = \underset{\alpha}{\operatorname{argmin}} \|\mathbf{y} - \mathbf{X}\boldsymbol{\alpha}\|_2^2 + \theta\|\Gamma\boldsymbol{\alpha}\|_2^2. \tag{7}$$

The term $\Gamma$ allows the imposition of prior knowledge on the solution using the approach in [24]–[26], where the training samples that are most dissimilar from a test sample are given less weight than the training samples that are most similar. Specifically, the following diagonal matrix $\Gamma \in \mathbb{R}^{n\times n}$ is considered:

$$\Gamma = \begin{bmatrix} \|\mathbf{y}-\mathbf{x}_1\|_2 & & 0 \\ & \ddots & \\ 0 & & \|\mathbf{y}-\mathbf{x}_n\|_2 \end{bmatrix}. \tag{8}$$

The coefficient vector $\hat{\boldsymbol{\alpha}}$ is then calculated as follows:

$$\hat{\boldsymbol{\alpha}} = (\mathbf{X}^T\mathbf{X} + \theta\Gamma^T\Gamma)^{-1}\mathbf{X}^T\mathbf{y}. \tag{9}$$

### C. Dempster–Shafer Theory

DST introduced by Demspter was later extended by Shafer [27]. DST is able to represent uncertainty and imprecision and can effectively deal with any union of classes and has been applied to many data fusion applications, e.g., [28], [29].

Let $\Theta$ be a finite universal set of mutually exclusive and exhaustive hypotheses, which is called a frame of discernment. In classification applications, $\Theta$ corresponds to a set of classes. The power set $2^\Theta$ is the set of all possible subsets of $\Theta$. A mass function or basic probability assignment (BPA) is a function $m: 2^\Theta \to [0,1]$, which satisfies the following properties:

$$\sum_{A\subseteq\Theta} m(A) = 1 \tag{10}$$

$$m(\emptyset) = 0 \tag{11}$$

where $\emptyset$ is the empty set. A subset $A$ with nonzero BPA is called a focal element. The value of $m(A)$ is a measure of the belief that is assigned to set $A$, not to subsets of $A$. Two common evidential measures, belief, and plausibility functions, respectively, are defined as follows ($A \subseteq \Theta$, $B \subseteq \Theta$):

$$\text{Bel}(A) = \sum_{B\subseteq A} m(B) \tag{12}$$

$$\text{Pl}(A) = \sum_{B\cap A\neq\emptyset} m(B), \text{Pl}(\emptyset) = 0. \tag{13}$$

These two measures have the following properties:

$$\text{Bel}(A) \leq \text{Pl}(A) \tag{14}$$

$$\text{Pl}(A) = 1 - \text{Bel}(\bar{A}) \tag{15}$$

where $\bar{A}$ is the complementary set of $A$: $\bar{A} = \Theta - A$.

For combining the measures of evidence from two independent sources, the Dempster's rule for combining two BPAs $m_1$ and $m_2$ is given by

$$m_{1,2}(\emptyset) = 0 \tag{16}$$

$$m_{1,2}(A) = \frac{1}{1-K}\sum_{B\cap C=A\neq\emptyset} m_1(B)m_2(C) \tag{17}$$

$$K = \sum_{B\cap C=\emptyset} m_1(B)m_2(C). \tag{18}$$

The normalization factor $K$ provides a measure of conflict between the two sources to be combined. This rule is commutative and associative. If there are more than two sources, the combination rule can be generalized by iteration. A joint decision is made based on the combined BPA by choosing the class with the maximum Bel or Pl [27].

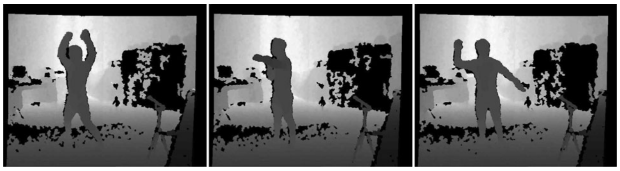


Fig. 1. Example of depth images of the actions (left to right) *jumping jacks*, *punching*, and *throwing a ball*.

## III. Human Action Database

The Berkeley MHAD [19] contains temporally synchronized and geometrically calibrated data from a motion capture system, stereo cameras, Kinect depth cameras, wireless wearable accelerometers, and microphones. After removing one erroneous sequence, it consists of 659 data sequences from 11 actions performed five times by seven male and five female subjects (11 aged 23–30 years and one elderly). The 11 actions are: *jumping in place* (jump), *jumping jacks* (jack), *bending-hands up all the way down* (bend), *punching* (punch), *waving two hands* (wave2), *waving right hand* (wave1), *clapping hands* (clap), *throwing a ball* (throw), *sit down and stand up* (sit+stand), *sit down* (sit), *stand up* (stand). The database incorporates the intraclass variations. For example, the speed of an action was different for different subjects.

There are five sensor modalities in the Berkeley MHAD, from which only the depth and inertial data are considered here. Furthermore, only the data from the Kinect camera placed in front of the subject are considered.

## IV. Fusion Approach

### A. Feature Extraction From Depth Data

Fig. 1 shows three example depth images of the actions *jumping jacks*, *punching*, and *throwing a ball*. A depth image can be used to capture the 3-D structure and shape information. Yang *et al.* [7] proposed to project depth frames onto three orthogonal Cartesian planes for the purpose of characterizing an action. In [30], we considered the same approach to achieve human action recognition based on depth images. Before performing depth image projections, first the foreground that contains the moving human subject needs to be extracted. Most of the dynamic background subtraction algorithms involve background modeling techniques [31], or spatiotemporal filtering to extract the spatiotemporal interest points corresponding to an action [32]. To make this task computationally efficient, with the consideration that a human subject is expected to be in front of the camera at a certain distance range, the mean depth value $\mu$ for each $M_0 \times N_0$ depth image is computed and, then, the foreground region is selected according to

$$d_{a,b} = \begin{cases} d_{a,b}, & \text{if } |d_{a,b}-\mu| \leq \epsilon \\ 0, & \text{otherwise} \end{cases} \tag{19}$$

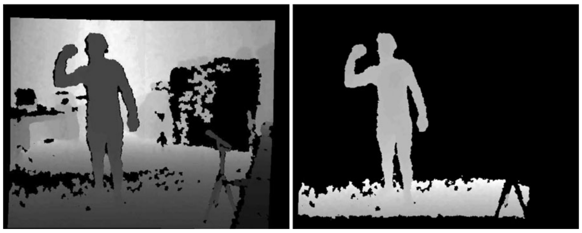


Fig. 2. Depth image foreground extraction. Original depth image (left). Foreground extracted depth image (right).

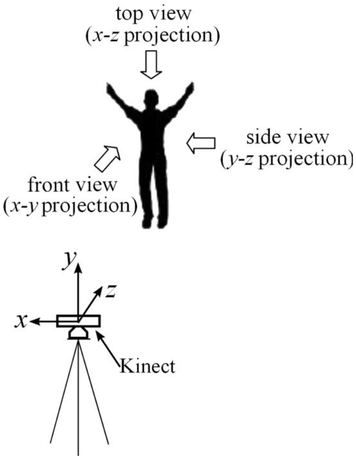


Fig. 3. Three projection views of a depth image.

where $d_{a,b}(a = 1, 2, \ldots, M_0, b = 1, 2, \ldots, N_0)$ is the depth value (indicating the distance between the Kinect camera and the object) of the pixel in the $a$th row and $b$th column of the depth image, $\epsilon$ is a threshold for the depth value with unit millimeter. We examined all depth video sequences in the Berkeley MHAD and found that the foreground can be removed by setting $\epsilon \in [800, 900]$. In our experiments, $\epsilon = 850$ was chosen. An example of the foreground extracted depth image is shown in Fig. 2.

Each foreground extracted depth image is then used to generate three 2-D projected maps corresponding to the front, side, and top views (see Fig. 3), denoted by $map_v$, where $v \in \{f, s, t\}$. For a point $(x, y, z)$ in the depth image with $z$ denoting the depth value in a right-handed coordinate system, the pixel values in the three projected maps $(map_f, map_s, map_t)$ are indicated by $z$, $x$, and $y$, respectively. For each projection view, the absolute difference between two consecutive projected maps is accumulated through an entire depth video sequence forming the so-called DMM [30]. Specifically, for each projected map, the motion energy is calculated as the absolute difference between two consecutive maps. For a depth video sequence with $N$ frames, the $\mathrm{DMM}_v$ is obtained by stacking the motion energy across an entire depth video sequence as follows:

$$\mathrm{DMM}_v = \sum_{q=1}^{N-1} \left| \mathrm{map}_v^{q+1} - \mathrm{map}_v^{q} \right| \tag{20}$$

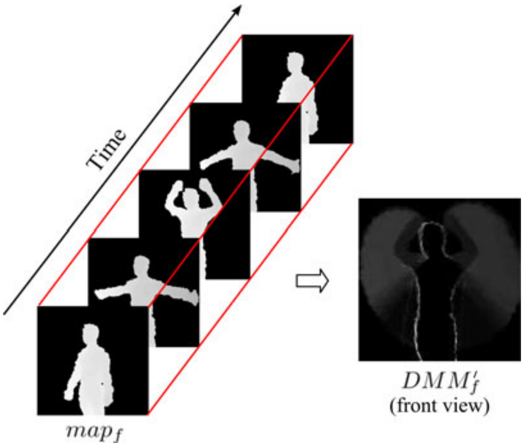


Fig. 4. $\mathrm{DMM}'_f$ generated from a *waving two hands* depth video sequence.

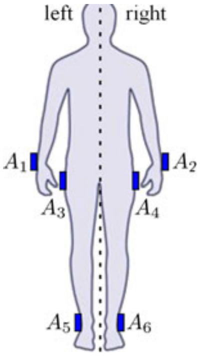


Fig. 5. Body placement of the six accelerometers in the Berkeley MHAD.

where $q$ represents the frame index, and $\mathrm{map}_v^q$ is the projected map of the $q$th frame for the projection view $v$. To keep the computational cost low, only the DMM generated from the front view, i.e., $\mathrm{DMM}_f$ is used as the feature in our case.

A bounding box is set to extract the nonzero region, as the region of interest (ROI) in each $\mathrm{DMM}_f$. Let the ROI extracted $\mathrm{DMM}_f$ be denoted by $\mathrm{DMM}'_f$. Fig. 4 shows an example $\mathrm{DMM}'_f$ generated from a *waving two hands* depth video sequence. As seen here, DMM is able to capture the characteristics of the motion. Since $\mathrm{DMM}'_f$ of different action video sequences may have different sizes, bicubic interpolation is used to resize all $\mathrm{DMM}'_f$ to a fixed size in order to reduce the intraclass variations.

### B. Feature Extraction From Acceleration Data

In the Berkeley MHAD, six three-axis wireless accelerometers $A_1, \ldots, A_6$ were placed on the subjects (see Fig. 5) to measure movements at the wrists, ankles, and hips. The accelerometers captured the motion data with the frequency of about 30 Hz. Here, each accelerometer sequence is partitioned into $N_s$ temporal windows as suggested in [19]. Statistical measures including *mean*, *variance*, *standard deviation*, and *root mean square* are computationally efficient and useful for capturing structural patterns in motion data. Therefore, these four measures are computed here along each direction in each

TABLE I
RECOGNITION RATES (%) WHEN USING DIFFERENT ACCELEROMETERS

| Accelerometer | Recognition rate (%) |
|---|---|
| $A_1$ | 86.67 |
| $A_2$ | 85.15 |
| $A_3$ | 71.49 |
| $A_4$ | 72.42 |
| $A_5$ | 56.43 |
| $A_6$ | 57.88 |

temporal window. For each accelerometer, concatenating all measures from $N_s$ windows results in a column feature vector of dimensionality $4 \times 3 \times N_s$.

Although six accelerometers were used in the Berkeley MHAD, we consider only two accelerometers because of practicality. To select the two accelerometers, an analysis was conducted by using the first six subjects for training and the remainder for testing. We set $N_s = 15$ (an analysis of choosing the number of segments is provided in Section V and employed SVM to classify the 11 actions. Based on the recognition performance and the positions of the accelerometers, the accelerometers $A_1$ and $A_4$ were found to be the most effective for the human actions in the database (see Table I). Note that $A_1$ and $A_4$ are placed on the left wrist and right hip, respectively, where people may wear a watch and a cell phone pouch in a nonintrusive manner. Neither $A_5$ nor $A_6$ were chosen because they were placed on the ankles, and were not able to generate useful information because of the relatively static foot movements in the actions.

### C. Feature-Level Fusion

Feature-level fusion involves fusing feature sets of different modality sensors. Let $\mathbf{U} = \{\mathbf{u}_l\}_{l=1}^{n}$ in $\mathbb{R}^{d_1}$ ($d_1$-dimensional feature space) and $\mathbf{V} = \{\mathbf{v}_l\}_{l=1}^{n}$ in $\mathbb{R}^{d_2}$ ($d_2$-dimensional feature space) represent the feature sets generated, respectively, from the Kinect depth camera and the accelerometer for $n$ training action samples. Column vectors $\mathbf{u}_l$ and $\mathbf{v}_l$ are normalized to have the unit length. Then, the fused feature set is represented by $\mathbf{F} = \{\mathbf{f}_l\}_{l=1}^{n}$ in $\mathbb{R}^{d_1+d_2}$ with each column vector being $\mathbf{f}_l = [\mathbf{u}_l^T, \mathbf{v}_l^T]^T$. The fused feature set is then fed into a classifier.

### D. Decision-Level Fusion

As noted earlier, for the $C$ action classes and a test sample $\mathbf{y}$, the frame of discernment is given by $\Theta = \{H_1, H_2, \ldots, H_C\}$, where $H_j : class(\mathbf{y}) = j, j \in \{1, 2, \ldots, C\}$. The classification decision of the classifiers SRC or CRC is based on the residual error with respect to class $j$, $r_j(\mathbf{y})$ using (4). Each class-specific representation $\widetilde{\mathbf{y}}_j$ and its corresponding class label $j$ constitute a distinct item of evidence regarding the class membership of $\mathbf{y}$. If $\mathbf{y}$ is close to $\widetilde{\mathbf{y}}_j$ according to the Euclidean distance, for small $r_j(\mathbf{y})$, it is most likely that $H_j$ is true. If $r_j(\mathbf{y})$ is large, the class of $\widetilde{\mathbf{y}}_j$ will provide little or no information about the class of $\mathbf{y}$. As demonstrated in [33] and [34], this item of evidence may be represented by a BPA over $\Theta$ defined as follows:

$$m(H_j|\widetilde{\mathbf{y}}_j) = \beta\phi_j(r_j(\mathbf{y})) \tag{21}$$

$$m(\Theta|\widetilde{\mathbf{y}}_j) = 1 - \beta\phi_j(r_j(\mathbf{y})) \tag{22}$$

$$m(D|\widetilde{\mathbf{y}}_j) = 0, \forall D \in 2^{\Theta} \setminus \{\Theta, H_j\} \tag{23}$$

where $\beta$ is a parameter such that $0 < \beta < 1$, and $\phi_j$ is a decreasing function satisfying these two conditions:

$$\phi_j(0) = 0 \tag{24}$$

$$\lim_{r(\mathbf{y}_j)\to\infty} \phi_j(r_j(\mathbf{y})) = 0. \tag{25}$$

However, as there exist many decreasing functions satisfying the two conditions, Denoeux [33] suggests to choose this $\phi_j$

$$\phi_j(r_j(\mathbf{y})) = e^{-\gamma_j r_j(\mathbf{y})^2} \tag{26}$$

with $\phi_j$ being a positive parameter associated with class $j$. In [34], a method for tuning the parameter $\gamma_j$ was proposed. To gain computational efficiency, $\gamma_j$ is set to 1 in our case, which makes $\phi_j$ a Gaussian function

$$\phi_j(r_j(\mathbf{y})) = e^{-r_j(\mathbf{y})^2}. \tag{27}$$

Since there are $C$ class-specific representations $\widetilde{\mathbf{y}}_j$'s, the final belief regarding the class label of $\mathbf{y}$ is obtained by combining the $C$ BPAs using the Dempster's rule of combination. The resulting global BPA $m_g$ was shown in [33] to be

$$\begin{aligned} m_g(H_j) = {} & \frac{1}{K_0}(1 - \{1 - \beta\phi_j(r_j(\mathbf{y}))\}) \cdot \prod_{p \neq j}\{1 - \beta\phi_p \\ & \times (r_p(\mathbf{y}))\} \quad p \in \{1, \ldots, C\} \end{aligned} \tag{28}$$

$$m_g(\Theta) = \frac{1}{K_0}\prod_{j=1}^{C}\{1 - \beta\phi_j(r_j(\mathbf{y}))\} \tag{29}$$

where $K_0$ is a normalization factor

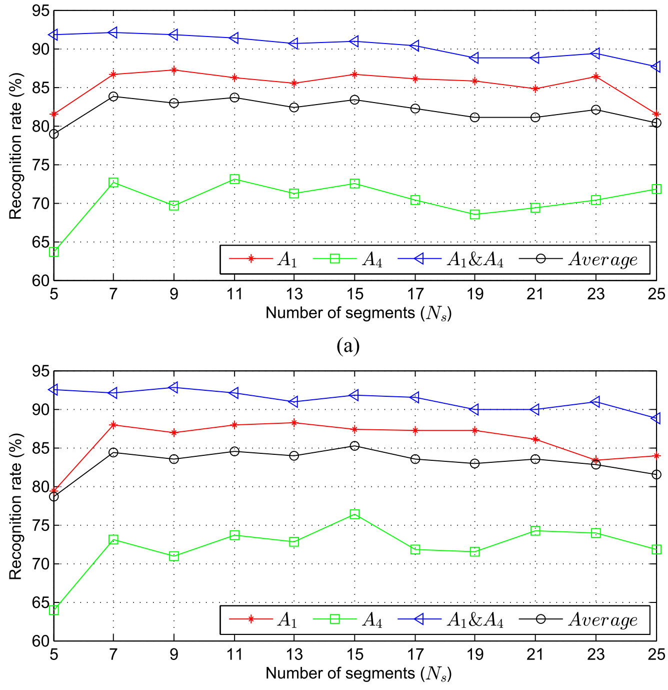
$$\begin{aligned} K_0 = {} & \sum_{j=1}^{C}\prod_{p \neq j}\{1 - \beta\phi_p(r_p(\mathbf{y}))\} \\ & + (1 - C)\prod_{j=1}^{C}\{1 - \beta\phi_j(r_j(\mathbf{y}))\}. \end{aligned} \tag{30}$$

In our decision-level fusion here, SRC or CRC is first applied to the depth feature set $\mathbf{U}$ and acceleration feature set $\mathbf{V}$, respectively. Therefore, two corresponding global BPAs $m_{g,1}$ and $m_{g,2}$ are generated. The combined BPA from $m_{g,1}$ and $m_{g,2}$ is then obtained via (17). The class label of a new test sample is determined, which corresponds to the maximum value of $\mathrm{Bel}(H_j)$, i.e., $max(\mathrm{Bel}(H_j))$.

## V. RESULTS

### A. Experimental Setup

The size of the depth images in the database is $480 \times 640$ pixels. After the foreground extraction from each depth image, the foreground extracted image was downsampled to $1/4$ of the original size, i.e., $120 \times 160$, to reduce the dimensionality and,



(a)

(b)

Fig. 6. Recognition rates (%) using different number of segments for accelerometer features. (a) SVM classifier. (b) CRC classifier.

thus, the computational complexity. Then, the DMM generation was performed on the reduced size images. To have a fixed size for the $\mathrm{DMM}'_f$, the sizes of these maps for all action samples in the database were found. The fixed size of each $\mathrm{DMM}'_f$ was set to the mean value of all of the sizes, which was $65 \times 50$. Therefore, each feature vector $\mathbf{u}_l$ had a dimensionality of 3250.

The number of segments $N_s$ for the acceleration data was determined via experimentation using the first six subjects for training and the rest for testing. SVM and CRC were employed as the classifiers, and the performance was tested using different $N_s$ (see Fig. 6). In this figure, $A_1$ denotes only using the accelerometer $A_1$, $A_4$ denotes only using the accelerometer $A_4$, and $A_1\&A_4$ denote using both of the accelerometers $A_1$ and $A_4$ together, where the features from the two accelerometers are stacked. Average denotes the mean accuracy of using the three accelerometer settings: $A_1$, $A_4$, and $A_1\&A_4$. The setting $N_s \in [13, 17]$ produced a consistent recognition performance under three accelerometer settings. Thus, $N_s = 15$ was chosen for the experiments. Each feature vector $\mathbf{v}_l$ had the dimension of 180 and 360 for the single-accelerometer setting and the two-accelerometer setting, respectively.

Although downsampling was used to reduce the dimensionality of the features generated from the depth images, the dimensionality of $\mathbf{u}_l$ and the fused feature $\mathbf{f}_l$ was greater than 3000. To gain computational efficiency, principal component analysis (PCA) was applied to $\mathbf{f}_l$ to reduce the dimensionality. The PCA transform matrix was calculated using the training feature set and, then, applied to the test feature set. The principal components that accounted for 95% of the total variation of the training feature set were considered.

## B. *Recognition Outcome*

For evaluation purposes, the leave-one-subject-out cross validation test (CV test) was considered. The recognition outcome was found for each subject as the left-out subject, and the final recognition outcome was averaged over all subjects to remove any bias. Five classifiers consisting of SVM, SRC, CRC, $k$-nearest neighbor ($k$-NN), and HMM were employed to evaluate the effectiveness of the proposed fusion approach. SVM was implemented using the LIBSVM toolbox[1] with an RBF kernel. Additionally, the package solver `l1_ls`[2] was used to calculate the sparse approximations for SRC. The optimal parameters for SVM and the regularization parameters, $\lambda$ and $\theta$, for SRC and CRC were assigned to be those that maximized the training accuracy via a fivefold cross validation. The parameter $k = 3$ was used in $k$-NN, as it generated the best outcome among different $k$s. The left-to-right topology with eight states [18] was used for HMM.

We compared the recognition performance of our feature-level fusion framework with the performance of each individual modality sensor (see Table II). By combining the features from the two differing modality sensors, the overall recognition rate was improved over the Kinect camera alone and over the accelerometer alone. This improved performance was consistent for all five classifiers. The overall recognition rate of accelerometer $A_1$ was found to be higher than that of accelerometer $A_4$, mainly due to the type of actions in the database consisting of hand movements. Fusing the Kinect data with $A_1$ data achieved similar recognition rates as fusing the Kinect data with $A_4$ data (except for the case when the $k$-NN classifier was used) due to the complementary nature of the data from the two differing modality sensors. For example, the accelerometer $A_4$ was not able to capture the hand movement of the action *waving two hands*; however, the $\mathrm{DMM}'_f$ generated from the depth images as shown in Fig. 4 could capture the characteristics of this action. As seen in Table II, using the two accelerometers $A_1$ and $A_4$ in the fusion approach did not lead to a substantial recognition improvement over the situation when using a single accelerometer $A_1$ or $A_4$. For the five classifiers, the recognition accuracy of Kinect + $A_1\&A_4$ came close to that of Kinect + $A_1$ (less than 1%), as the accelerometer $A_4$ did not provide any additionally useful data to distinguish certain actions, in particular, the actions that involved moving hands and arms.

Tables III–V show three recognition confusion matrices corresponding to using Kinect only, using accelerometer $A_1$ only, and using Kinect and $A_1$ fusion, respectively, with the SVM classifier. As it can be seen from Table III, the misclassifications mostly occurred among the actions *sit down and stand up*, *sit down*, and *stand up*. As illustrated in Fig. 7, the DMMs (representing shape and motion) of these actions appeared quite similar; however, the shape and motion of the actions sit down and stand up occurred in different temporal orders. The action *sit down and stand up* is a complex movement composed of *sit down* and *stand up*. The failure of the DMM to distinguish the shape and motion cues occurred in different temporal orders

[1] http://www.csie.ntu.edu.tw/ ~cjlin/libsvm/

[2] http://www.stanford.edu/ ~boyd/l1_ls

TABLE II
RECOGNITION RATES (%) FOR THE LEAVE-ONE-SUBJECT-OUT CVT

| Method | Kinect | $A_1$ | Kinect+$A_1$ | $A_4$ | Kinect+$A_4$ | $A_1$ & $A_4$ | Kinect+$A_1$ & $A_4$ |
|---|---|---|---|---|---|---|---|
| SVM | 92.39 | 91.77 | 98.48 | 79.03 | 98.18 | 94.20 | **99.24** |
| SRC | 84.93 | 92.38 | 98.79 | 72.03 | 97.57 | 95.73 | **99.54** |
| CRC | 87.52 | 93.00 | 98.18 | 82.19 | 97.11 | 96.81 | **99.13** |
| $k$-NN | 65.04 | 86.91 | 91.17 | 65.65 | 82.57 | 89.08 | **91.85** |
| HMM | 84.80 | 90.43 | 97.57 | 78.12 | 96.50 | 93.77 | **98.18** |

TABLE III
CONFUSION MATRIX WHEN USING KINECT ONLY FOR THE LEAVE-ONE-SUBJECT-OUT CVT

| Action | jump | jack | bend | punch | wave2 | wave1 | clap | throw | sit+stand | sit | stand |
|---|---|---|---|---|---|---|---|---|---|---|---|
| jump | 98.33 | - | - | - | - | - | - | 1.67 | - | - | - |
| jack | - | 95 | - | - | - | - | - | 5 | - | - | - |
| bend | - | - | 100 | - | - | - | - | - | - | - | - |
| punch | - | - | - | 100 | - | - | - | - | - | - | - |
| wave2 | - | 8.33 | - | - | 91.67 | - | - | - | - | - | - |
| wave1 | - | - | - | - | - | 100 | - | - | - | - | - |
| clap | - | - | - | 11.67 | - | - | 86.67 | 1.67 | - | - | - |
| throw | - | - | - | 1.69 | 1.69 | - | - | 96.61 | - | - | - |
| sit+stand | - | - | - | - | - | - | - | - | 88.33 | 3.33 | 8.33 |
| sit | - | - | - | - | - | - | - | - | 8.33 | 86.67 | 5 |
| stand | - | - | - | - | - | - | - | - | 13.33 | 13.33 | 73.33 |

TABLE IV
CONFUSION MATRIX WHEN USING ACCELEROMETER $A_1$ ONLY FOR THE LEAVE-ONE-SUBJECT-OUT CVT

| *Action* | jump | jack | bend | punch | wave2 | wave1 | clap | throw | sit+stand | sit | stand |
|---|---|---|---|---|---|---|---|---|---|---|---|
| jump | 93.33 | 1.67 | - | 1.67 | - | - | 3.33 | - | - | - | - |
| jack | 11.67 | 88.33 | - | - | - | - | - | - | - | - | - |
| bend | - | - | 100 | - | - | - | - | - | - | - | - |
| punch | 6.67 | - | - | 75 | - | - | 18.33 | - | - | - | - |
| wave2 | - | - | - | - | 100 | - | - | - | - | - | - |
| wave1 | - | - | - | - | - | 100 | - | - | - | - | - |
| clap | 1.69 | - | - | 3.39 | - | - | 93.22 | 1.69 | - | - | - |
| throw | 1.69 | - | - | - | - | 10.17 | 1.69 | 76.27 | - | - | 10.17 |
| sit+stand | - | - | - | - | - | 1.67 | - | 1.67 | 96.67 | - | - |
| sit | - | - | - | - | - | 1.67 | - | 1.67 | - | 96.67 | - |
| stand | - | - | - | - | - | 3.33 | - | 1.67 | - | 5 | 90 |

TABLE V
CONFUSION MATRIX WHEN USING KINECT AND ACCELEROMETER $A_1$ FUSION FOR THE LEAVE-ONE-SUBJECT-OUT CVT

| Action | jump | jack | bend | punch | wave2 | wave1 | clap | throw | sit+stand | sit | stand |
|---|---|---|---|---|---|---|---|---|---|---|---|
| jump | 100 | - | - | - | - | - | - | - | - | - | - |
| jack | 1.67 | 98.33 | - | - | - | - | - | - | - | - | - |
| bend | - | - | 100 | - | - | - | - | - | - | - | - |
| punch | - | - | - | 98.33 | - | - | 1.67 | - | - | - | - |
| wave2 | - | 1.67 | - | - | 98.33 | - | - | - | - | - | - |
| wave1 | - | - | - | - | - | 100 | - | - | - | - | - |
| clap | 1.69 | - | - | - | - | - | 98.31 | - | - | - | - |
| throw | - | - | - | 1.69 | - | - | - | 98.31 | - | - | - |
| sit+stand | - | - | - | - | - | - | - | - | 100 | - | - |
| sit | - | - | - | - | - | - | - | - | - | 100 | - |
| stand | - | - | - | - | - | - | - | - | 8.33 | - | 91.67 |

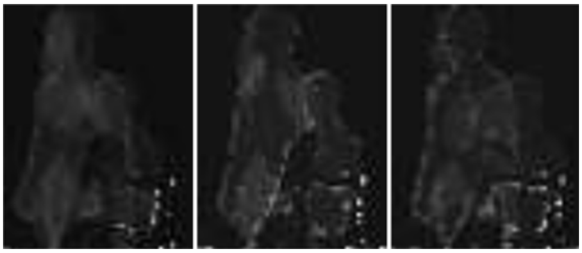


Fig. 7. DMMs for the actions (left to right) *sit down and stand up*, *sit down*, and *stand up*.

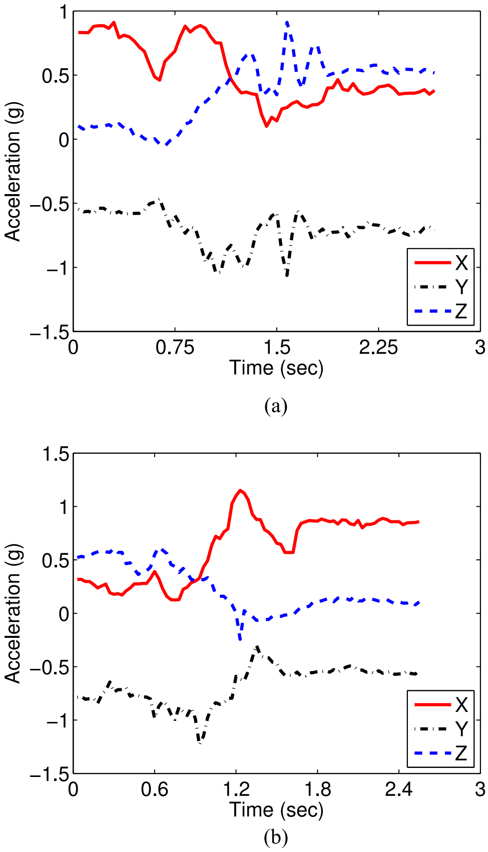


Fig. 8. Three-axis acceleration signals corresponding to the actions. (a) *Sit down*. (b) *Stand up*.

of these actions, which demonstrated a disadvantage of using the Kinect alone.Table IV shows the confusion matrix associated with using accelerometer $A_1$ alone. The accuracies of the actions *sit down and stand up*, *sit down*, and *stand up* were improved noticeably (over 10% for these three actions) as compared with the Kinect only situation. The three-axis acceleration data were able to distinguish similar motions that occurred in different temporal orders, since the trend of the three-axis acceleration data for the action *sit down* was opposite to that for the action *stand up* as illustrated in Fig. 8. However, some of the actions, e.g., *punch*, produced much lower accuracy than using the Kinect alone. The action punch was mostly misclassified with the action *clap*. From Fig. 9, one sees that the features generated from the three-axis acceleration data for the two actions were similar. By integrating the Kinect depth images with the acceleration data, the fused features were more discriminatory leading to the improved recognition rates over the Kinect alone and the accelerometer alone situations. Table V shows that the low-recognition rates for those actions when using one modality sensing improved when the Kinect and accelerometer data were used together due to the complementary nature of the data from these two differing modality sensors. For example, the overall recognition rate for the action *sit* was improved by 13% over the Kinect alone, and the accuracy for the action *punch* was improved by 23% over the accelerometer alone.

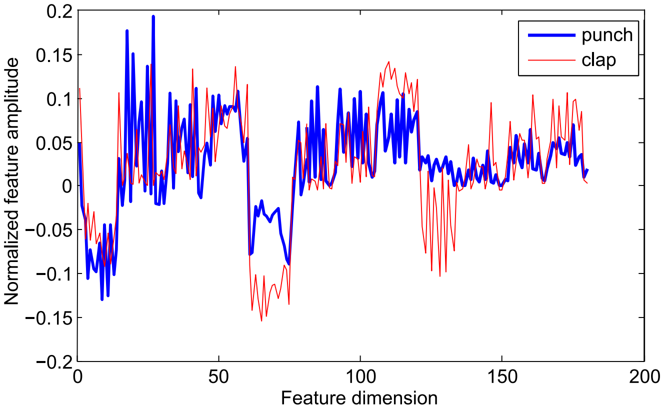


Fig. 9. Features generated from three-axis acceleration data for the actions *punch* and *clap*.

To investigate training data size, we conducted a random test experiment by randomly choosing half of the subjects for training and the remaining subjects for testing. Each test was repeated 20 times, and the mean performance (mean recognition rate ± standard deviation) was computed. As can be seen from Table VI, our fusion approach produced the same performance as the CV test. Again, the overall recognition rate of the fusion approach was improved over the Kinect alone and the accelerometer alone (the improvement was even greater than that of the CV test). This trend was consistent for the four different classifiers.

We also tested the effectiveness of our decision-level fusion approach. We used CRC rather than SRC because of its computational efficiency. As suggested in [33], we set $\beta = 0.95$ for the BPA in (21). Table VII shows that the feature-level fusion outperformed the decision-level fusion in most cases. However, the decision-level fusion involving the Kinect camera and accelerometer still achieved better performance than each individual modality sensor. One disadvantage of the decision-level fusion is that CRC needs to be applied to both the depth feature and the acceleration feature. In other words, CRC has to be run twice.

We conducted a comparison of our fusion approach with the one described in [19], where multiple kernel learning (MKL) was employed to fuse information from different modality sensors. In [19], each depth video was first divided into eight disjoint depth-layered multichannel (DLMC) videos by dividing

TABLE VI
RECOGNITION RATES (%) FOR RANDOM TEST

| Method | Kinect | $A_1$ | Kinect+$A_1$ | $A_4$ | Kinect+$A_4$ | $A_1$ & $A_4$ | Kinect+$A_1$ & $A_4$ |
|---|---|---|---|---|---|---|---|
| SVM | 86.34±1.92 | 87.69±2.95 | 97.02±1.33 | 70.52±2.98 | 96.41±1.43 | 90.30±2.27 | **98.23**±0.70 |
| SRC | 79.20±2.01 | 87.05±2.82 | 97.41±0.94 | 65.65±3.24 | 94.64±1.97 | 90.64±2.64 | **98.14**±0.83 |
| CRC | 81.61±2.00 | 88.09±3.12 | 96.87±1.09 | 73.59±3.43 | 94.89±2.01 | 92.01±2.84 | **97.94**±0.93 |
| $k$-NN | 63.81±2.06 | 84.02±3.71 | 90.65±2.08 | 62.71±3.31 | 81.52±3.16 | 86.16±3.26 | **90.93**±1.64 |
| HMM | 78.83±2.24 | 86.12±2.47 | 95.70±1.38 | 69.13±2.75 | 93.42±1.82 | 89.62±2.06 | **96.68**±1.14 |

TABLE VII
RECOGNITION RATES (%) COMPARISON BETWEEN FEATURE-LEVEL FUSION (CRC) AND DECISION-LEVEL FUSION (CRC)

| *Method* | Kinect+$A_1$ | Kinect+$A_4$ | Kinect+$A_1$ & $A_4$ |
|---|---|---|---|
| | | CV test | |
| Feature-level fusion | 98.18 | 97.11 | 99.13 |
| Decision-level fusion | 98.05 | 97.38 | 98.97 |
| | | Random test | |
| Feature-level fusion | 96.87 | 94.89 | 97.94 |
| Decision-level fusion | 96.04 | 95.36 | 97.31 |

TABLE VIII
COMPARISON OF RECOGNITION RATES (%) BETWEEN OUR FEATURE-LEVEL FUSION AND THE MULTIPLE KERNEL LEARNING METHOD IN [19]

| Method | Kinect+$A_1$ | Kinect+$A_4$ | Kinect+$A_1$ & $A_4$ |
|---|---|---|---|
| | | CV test | |
| Ours | 98.48 | 98.18 | 99.24 |
| [19] | 92.65 | 91.93 | 93.77 |
| | | Random test | |
| Ours | 97.02 | 96.41 | 98.23 |
| [19] | 90.59 | 88.87 | 91.43 |

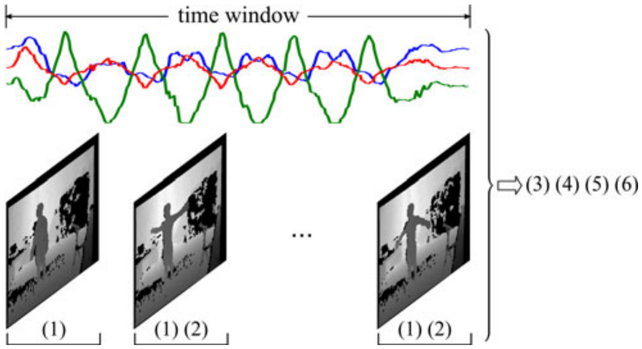


Fig. 10. Real-time action recognition timeline of our fusion framework.

TABLE IX
AVERAGE AND STANDARD DEVIATION OF PROCESSING TIME OF THE COMPONENTS OF OUR FUSION APPROACH

| Component | Processing time (ms) |
|---|---|
| 1 | 2.1±0.3/frame |
| 2 | 2.5±0.1/frame |
| 3 | 1.4 ±0.3/action sequence |
| 4 | 2.4 ±0.6/action sequence |
| 5 | 1.3 ±0.2/action sequence |
| 6 | 3.2 ±0.4/action sequence |

the depth range into eight equal depth layers and by keeping the pixels within the depth range of the corresponding depth layer. The first two depth layers and the last depth layer were discarded due to a lack of depth information. Histogram of gradients (HOGs) and histogram of flow (HOF) features [35] were then extracted from each DLMC video. Then, the bag-of-features representation in [36] was employed to code the DLMC videos into histograms to serve as the features. Note that the type of fusion in [19] was a feature-level fusion and SVM was employed in MKL. Therefore, our feature-level fusion with the SVM classifier is compared with the approach in [19]. As listed in Table VIII, our approach led to higher recognition rates. For the acceleration data, only variance was utilized to extract features from the temporal windows as described in [19]. For the depth videos, HOG/HOF features were computed at the space-time interest points (STIPs). Due to the noise in the depth videos, the detected STIPs contained many points that were not related to the actions. In addition, the feature extraction method in [19] calculates the HOG/HOG descriptors for each DLMC video, which is computationally expensive and poses real-time implementation challenges.

Finally, the computational aspect of our solution is considered (see Fig. 10). An action is normally completed approximately within a 2-s time duration. The numbers in Fig. 10 indicate the main components in our fusion approach. More specifically, the components are as follows. 1) Depth image foreground extraction and image downsampling. 2) $\mathrm{DMM}_f = \mathrm{DMM}_f + |\mathrm{map}_f^{q+1} - \mathrm{map}_f^q|$ computation. 3) Acceleration feature extraction captured within a time window. 4) ROI extraction from the $\mathrm{DMM}_f$ and resizing the $\mathrm{DMM}'_f$ to a fixed size via bicubic interpolation. 5) Applying PCA dimensionality reduction on the fused feature vector. 6) Performing classification using SVM. The components (1) and (2) are executed right after each depth frame is captured, while the components (3)–(6) are performed after an action sequence completes. Since the PCA transform matrix is calculated using the training feature set, it can be directly applied to the feature vector of a test sample. Our code is written in MATLAB, and the processing time reported is for an Intel i7 Quadcore 2.67-GHz PC platform with 8-GB RAM. The average processing time of each component is listed in Table IX.

## VI. Conclusion

In this paper, a fusion framework was introduced that utilizes data from two differing modality sensors [a Kinect camera and a wearable inertial sensor (accelerometer)] for the purpose of achieving human action recognition. Using data from the Berkeley multimodality human action database, improved recognition rates were achieved by using these two differing modality sensors together compared with the situations when each sensor was used individually. This was found to be due to the complementary aspect of data from these two differing modality sensors.

## References

[1] Y.-J. Chang, S.-F. Chen, and J.-D. Huang, "A kinect-based system for physical rehabilitation: A pilot study for young adults with motor disabilities," *Res. Develop. Disabilities*, vol. 32, no. 6, pp. 2566–2570, Nov. 2011.

[2] C. Chen, K. Liu, R. Jafari, and N. Kehtarnavaz, "Home-based senior fitness test measurement system using collaborative inertial and depth sensors," in *Proc. IEEE 36th Annu. Int. Conf. Eng. Med. Biol. Soc.*, Chicago, IL, USA, Aug. 2014, pp. 4135–4138.

[3] 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 Trans. Consum. Electron.*, vol. 58, no. 3, pp. 863–871, Aug. 2012.

[4] C. Chen, N. Kehtarnavaz, and R. Jafari, "A medication adherence monitoring system for pill bottles based on a wearable inertial sensor," in *Proc. IEEE 36th Annu. Int. Conf. Eng. Med. Biol. Soc.*, Chicago, IL, USA, Aug. 2014, pp. 4983–4986.

[5] M. Ermes, J. Pärkkä, J. Mäntyjärvi, and I. Korhonen, "Detection of daily activities and sports with wearable sensors in controlled and uncontrolled conditions," *IEEE Trans. Inf. Technol. Biomed.*, vol. 12, no. 1, pp. 20–26, Jan. 2008.

[6] W. Li, Z. Zhang, and Z. Liu, "Action recognition based on a bag of 3D points," in *Proc. IEEE Conf. Comput. Vis. Pattern Recog. Workshops*, San Francisco, CA, USA, Jun. 2010, pp. 9–14.

[7] X. Yang, C. Zhang, and Y. Tian, "Recognizing actions using depth motion maps-based histograms of oriented gradients," in *Proc. ACM Conf. Multimedia*, Nara, Japan, Oct. 2012, pp. 1057–1060.

[8] J. Wang, Z. Liu, J. Chorowski, Z. Chen, and Y. Wu, "Robust 3D action recognition with random occupancy patterns," in *Proc. Eur. Conf. Comput. Vis.*, Florence, Italy, Oct. 2012, pp. 872–885.

[9] O. Oreifej and Z. Liu, "HON4D: Histogram of oriented 4D normals for activity recognition from depth sequences," in *Proc. IEEE Conf. Comput. Vis. Pattern Recog.*, Portland, OR, USA, Jun. 2013, pp. 716–723.

[10] L. Xia and J. Aggarwal, "Spatio-temporal depth cuboid similarity feature for activity recognition using depth camera," in *Proc. IEEE Conf. Comput. Vis. Pattern Recog.*, Portland, OR, USA, Jun. 2013, pp. 2834–2841.

[11] M. Zhang and A. A. Sawchuk, "Human daily activity recognition with sparse representation using wearable sensors," *IEEE Trans. Inf. Technol. Biomed.*, vol. 17, no. 3, pp. 553–560, May 2013.

[12] 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 Trans. Inf. Technol. Biomed.*, vol. 14, no. 5, pp. 1166–1172, Sep. 2010.

[13] E. Jovanov, A. Milenkovic, C. Otto, and P. C. de Groen, "A wireless body area network of intelligent motion sensors for computer assisted physical rehabilitation," *J. NeuroEng. Rehabil.*, vol. 2, no. 6, pp. 1–10, 2005.

[14] J. Chen, K. Kwong, D. Chang, J. Luk, and R. Bajcsy, "Wearable sensors for reliable fall detection," in *Proc. IEEE Conf. Eng. Med. Biol. Soc.*, Shanghai, China, Sep. 2005, pp. 3551–3554.

[15] H. M. Hondori, M. Khademi, and C. V. Lopes, "Monitoring intake gestures using sensor fusion (Microsoft Kinect and inertial sensors) for smart home tele-rehab setting," presented at the IEEE 1st Annu. Healthcare Innovation Conf., Houston, TX, USA, Nov. 2012.

[16] M. Kepski, B. Kwolek, and I. Austvoll, "Fuzzy inference-based reliable fall detection using kinect and accelerometer," in *Proc. 11th Int. Conf. Artif. Intell. Soft Comput.*, Zakopane, Poland, 2012, pp. 266–273.

[17] B. Delachaux, J. Rebetez, A. Perez-Uribe, and H. F. S. Mejia, "Indoor activity recognition by combining one-vs.-all neural network classifiers exploiting wearable and depth sensors," in *Proc. 12th Int. Conf. Artif. Neural Netw., Adv. Comput. Intell.*, Puerto de la Cruz, Spain, Jun. 2013, pp. 216–223.

[18] K. Liu, C. Chen, R. Jafari, and N. Kehtarnavaz, "Fusion of inertial and depth sensor data for robust hand gesture recognition," *IEEE Sensors J.*, vol. 14, no. 6, pp. 1898–1903, Jun. 2014.

[19] F. Ofli, R. Chaudhry, G. Kurillo, R. Vidal, and R. Bajcsy, "Berkeley MHAD: A comprehensive multimodal human action database," in *Proc. IEEE Workshop Appl. Comput. Vis.*, Clearwater Beach, FL, USA, Jan. 2013, pp. 53–60.

[20] J. Wright, A. Y. Yang, A. Ganesh, S. S. Sastry, and Y. Ma, "Robust face recognition via sparse representation," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 31, no. 2, pp. 210–227, Feb. 2009.

[21] J. Wright, Y. Ma, J. Mairal, G. Sapiro, T. S. Huang, and S. Yan, "Sparse representation for computer vision and pattern recognition," *Proc. IEEE*, vol. 98, no. 6, pp. 1031–1044, Jun. 2010.

[22] L. Zhang, M. Yang, and X. Feng, "Sparse representation or collaborative representation: Which helps face recognition?" in *Proc. IEEE Int. Conf. Comput. Vis.*, Barcelona, Spain, Nov. 2011, pp. 471–478.

[23] A. N. Tikhonov and V. Y. Arsenin, *Solutions of Ill-Posed Problems*. Washington, DC, USA: V. H. Winston & Sons, 1977.

[24] C. Chen, E. W. Tramel, and J. E. Fowler, "Compressed-sensing recovery of images and video using multihypothesis predictions," in *Proc. 45th Asilomar Conf. Signals, Syst. Comput.*, Pacific Grove, CA, USA, Nov. 2011, pp. 1193–1198.

[25] C. Chen, W. Li, E. W. Tramel, and J. E. Fowler, "Reconstruction of hyperspectral imagery from random projections using multihypothesis prediction," *IEEE Trans. Geosci. Remote Sens.*, vol. 52, no. 1, pp. 365–374, Jan. 2014.

[26] C. Chen and J. E. Fowler, "Single-image super-resolution using multihypothesis prediction," in *Proc. 46th Asilomar Conf. Signals, Syst. Comput.*, Pacific Grove, CA, USA, Nov. 2012, pp. 608–612.

[27] G. Shafer, *A Mathematical Theory of Evidence*. Princeton, NJ, USA: Princeton Univ. Press, 1976.

[28] M. Rombaut and Y. M. Zhu, "Study of dempster-shafer theory for image segmentation applications," *Image Vis. Comput.*, vol. 20, no. 1, pp. 15–23, Jan. 2002.

[29] O. Basir, F. Karray, and H. Zhu, "Connectionist-based Dempster–Shafer evidential reasoning for data fusion," *IEEE Trans. Neural Netw.*, vol. 16, no. 6, pp. 1513–1530, Nov. 2005.

[30] C. Chen, K. Liu, and N. Kehtarnavaz, "Real time human action recognition based on depth motion maps," J. Real-Time Image Process., (2013, Aug). [Online]. Available at: http://link.springer.com/article/10.1007%2Fs11554-013-0370-1, doi: 10.1007/s11554-013-0370-1.

[31] C. Stauffer and W. E. L. Grimson, "Learning patterns of activity using real-time tracking," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 22, no. 8, pp. 747–757, Aug. 2000.

[32] I. Laptev, "On space-time interest points," *Int. J. Comput. Vis.*, vol. 64, no. 2/3, pp. 107–123, Jun 2005.

[33] T. Denoeux, "A $k$-nearest neighbor classification rule based on Dempster–Shafer theory," *IEEE Trans. Syst., Man, Cybern.*, vol. 25, no. 5, pp. 804–813, May 1995.

[34] L. M. Zouhal and T. Denoeux, "An evidence-theoretic $k$-NN rule with parameter optimization," *IEEE Trans. Syst., Man, Cybern. C, Appl. Rev.*, vol. 28, no. 2, pp. 263–271, May 1998.

[35] I. Laptev, M. Marszalek, C. Schmid, and B. Rozenfeld, "Learning realistic human actions from movies," in *Proc. IEEE Conf. Comput. Vis. Pattern Recog.*, Anchorage, AK, USA, Jun. 2008, pp. 1–8.

[36] C. Schüldt, I. Laptev, and B. Caputo, "Recognizing human actions: A local SVM approach," in *Proc. 17th Int. Conf. Pattern Recog.*, Cambridge, U.K., Aug. 2004, pp. 32–36.

**Chen Chen** (S'10) received the B.E. degree in automation from Beijing Forestry University, Beijing, China, in 2009, and the M.S. degree in electrical engineering from Mississippi State University, Starkville, MS, USA, in 2012. He is currently working toward the Ph.D. degree with the Department of Electrical Engineering, University of Texas at Dallas, Richardson, TX, USA.

His research interests include compressed sensing, signal and image processing, pattern recognition, computer vision, and hyperspectral image analysis.

**Roozbeh Jafari** (SM'12) received the Ph.D. degree in computer science from the University of California Los Angeles (UCLA), Los Angeles, CA, USA.

He completed the Postdoctoral Fellowship with UC-Berkeley. He is currently an Associate Professor with the University of Texas at Dallas, Richardson, TX, USA. His research interests include wearable computer design and signal processing. His research has been funded by the NSF, NIH, DoD (TATRC), AFRL, AFOSR, DARPA, SRC, and industry (Texas Instruments, Tektronix, Samsung & Telecom Italia).

**Nasser Kehtarnavaz** (S'82–M'86–SM'92–F'12) received the Ph.D. degree in electrical and computer engineering from Rice University, Houston, TX, USA, in 1987.

He is currently a Professor of electrical engineering and the Director with the Signal and Image Processing Laboratory, University of Texas at Dallas, Richardson, TX, USA. His research interests include signal and image processing, real-time implementation on embedded processors, biomedical image analysis, and pattern recognition.