Multi-Modal Human Action Recognition Using Deep Neural Networks Fusing Image and Inertial Sensor Data

Inhwan Hwang, Geonho Cha, and Songhwai Oh

Abstract— **Human action recognition has been studied in many fields including computer vision and sensor networks using inertial sensors. However, there are limitations such as spatial constraints, occlusions in images, sensor unreliability, and the inconvenience of users. In order to solve these problems we suggest a sensor fusion method for human action recognition exploiting RGB images from a single fixed camera and a single wrist mounted inertial sensor. These two different domain information can complement each other to fill the deficiencies that exist in both image based and inertial sensor based human action recognition methods. We propose two convolutional neural network (CNN) based feature extraction networks for image and inertial sensor data and a recurrent neural network (RNN) based classification network with long short term memory (LSTM) units. Training of deep neural networks and testing are done with synchronized images and sensor data collected from five individuals. The proposed method results in better performance compared to single sensor-based methods with an accuracy of 86.9 % in cross-validation. We also verify that the proposed algorithm robustly classifies the target action when there are failures in detecting body joints from images.**

I. Introduction

Human action recognition has been studied in computer vision [1], [2], and inertial measurement unit (IMU) based sensor networks [3], [4]. Both methods have accomplished high performance, but there exist several limitations. Camera based action recognition ensures high accuracy only in certain situations where a fixed camera is stabilized without disturbance such as occlusion, noise, and illumination of the target human. If the target is out of the searching area or occluded, it is difficult to accurately recognize the action.

There is also the problem of privacy. Let us assume that a patient is in a nursing hospital who needs a full time surveillance. It may be possible to install surveillance cameras all around the hospital. However, lots of the urgent situations may happen in private spaces such as a toilet, a ward, and a shower room. Image based action recognition requires a sequence of RGB images which can depict the patient's behavior which result in the invasion of privacy. Unlike computer vision based action recognition, IMU-based action recognition requires only a pile of sensor data which cannot harm the patient's privacy. However, the IMU-based action recognition shows meaningful performance only with multiple sensors mounted on the whole body which hinders the patients' convenience [3]. When the limited number of IMU sensors are available, only few actions which have distinctive features can be classified. For this reason, IMU-based action recognition is limited to certain applications such as the motion capturing system with multiple IMU sensors attached on different parts of a body and classification of broad and vague action grouping which is closer to activity recognition than action recognition [5]. To solve these problems, we propose a human action recognition system which effectively combines these two methodologies to complement each other's deficiencies.

For action recognition using a sequence of images, the most important part is extracting features out of each image. Among the diverse features from the images, human pose is the most efficient and effective feature for action recognition [1]. For this reason, we focus on extracting human poses out of images rather than extracting other visual features such as optical flow, image gradients, and silhouettes. For human pose estimation, we have adopted deep neural networks which show great performance in many areas of machine learning tasks recently. Since AlexNet [6] which utilizes convolutional neural networks (CNNs) won the ImageNet contest in 2012 with overwhelming performance, CNNs have not only been exploited in the field of image processing but also have been used as a core technology to replace the technology for may conventional artificial intelligence tasks.

Unlike estimating a human pose from a single image, action recognition through videos and IMU sensors has temporal information so that it is necessary to utilize an algorithm considering the temporal flow. For this reason, we choose a recurrent neural network (RNN) which is widely applied in speech recognition and natural language processing. RNNs have an additional connection from the $(t-1)$th hidden layer to the tth hidden layer. This enables the hidden layer at time t to contain the information at time $t-1$. A traditional RNN connects two successive hidden layers with weights and biases with a non-linear activation function. Although a simple connection can transfer the information from time $t-1$ to t, there are critical issues, such as vanishing gradients and exploding gradient problems, which prevent the flow of information from a distant past. In order to suppress above problems, several methods are applied to the connection between two successive hidden layers. Long short term memory (LSTM) [7] and a gated recurrent unit (GRU) [8] are the most popular methods and we choose LSTM for overcoming the deficiencies of a vanilla RNN.

The proposed method assumes a situation in which a user is wearing a watch-type smart device, such as a smart watch or a fitness band. At the same time, the user's motion is captured by a single RGB camera from a fixed position. We

I. Hwang, G. Cha, and S. Oh are with the Department of Electrical and Computer Engineering and ASRI, Seoul National University, Seoul, Korea (e-mail: {inhwan.hwang, geonho.cha}@cpslab.snu.ac.kr, songhwai@snu.ac.kr).

compare the action recognition performance in three different input data types, image alone, inertial sensor data alone, and combined. Ahead of performing each action recognition, feature extraction should be performed on both camera images and inertial sensor measurements. Once features are extracted from each sensor domain, they are fed into the RNN based classifier for action recognition.

II. Related Work

There have been studies on human pose estimation from RGB images [9], [10], [11], [12]. In earlier work, the feature extraction step was not jointly optimized with the pose inference step, and a human pose was inferred using a structural support vector machine (SSVM) [13]. In order to utilize SSVMs, one needs a unary term which represents the detection score of each body joint and a pairwise term which models the relative relation between two joints. In [9], a histogram of oriented gradient (HOG) feature [14] was utilized for part detection and the squared distance between joints was used for modeling the pairwise term.

With the application of CNNs, the performances of many computer vision applications including human pose estimation has improved significantly. With CNNs, the feature extraction step and the pose estimation step can be jointly optimized, resulting in better performance. In [10], all joints from an RGB image are directly regressed in 2D Euclidean coordinates. They simply apply the AlexNet structure to design a pose regression network. After that, there have been researches which show that the score from the regression result gives better performance and makes the training easier [11], [12]. In [11], several convolutional networks were sequentially incorporated to implicitly model long-range dependencies between variables in a human pose estimation task. In [12], features were extracted from all scales using repeated bottom-up and top-down processing.

Action recognition using sensors other than cameras has been studied using diverse sensors. Most approaches are focused on using IMU sensors for action recognition and they usually exploit multiple sensors that are attached on a human body. The Opportunity dataset is one of the most popular human activity recognition dataset with 72 IMU sensors attached on a whole body [15]. With the Opportunity dataset, several works have been done with proper feature extraction and classification methods [3]. Furthermore, a neural network is exploited for better classification without putting the effort of humans for designing proper features [4]. While human action recognition with multiple inertial sensors shows reasonably high performance with diverse actions in detail categories [4], human action recognition with a single sensor mounted on the wrist as if he or she wears a fitness band or a smart watch only distinguish limited actions [16], [5]. It is more convenient in actual uses if sensors are attached on few certain parts of a body but it is hard to track user's actions considering diverse everyday actions. In addition to approaches using multiple sensors attached to a body, diverse sensors including eight motion capture camera, two stereo cameras, two quad cameras, two Kinects, six accelerometers, and four microphones to recognize human actions [2]. It is more practical to exploits IMU sensors in smartphones [17] and wearable device [18] for action recognition. However, performance is not as good as the systems using a full body sensor network and there are fewer actions that can be classified. Even sensors other than inertial sensors and cameras such as a muscle electrocardiogram (ECG) sensor [19], an IMU sensor conjunction with a microphone [16] and indoor radio signal [20] are utilized for action recognition.

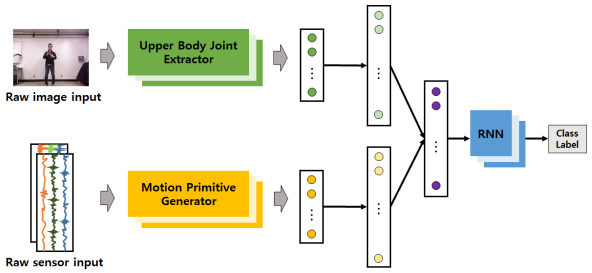


Fig. 1. An overview of the proposed action recognition system using images and IMU sensor data.

III. Proposed Method

In this section, we propose a deep neural network based action recognition using a wrist-worn IMU sensor and a single RGB camera. For both the IMU-based and the camera-based action recognition, we use neural network-based feature extraction and action classification. The overall structure of the proposed network is shown in Figure 1. Both the upper body joint extractor and the motion primitive generator are designed with a CNN and they are trained separately. For classification, we adopt an RNN with LSTM units. The detail of each network is described in following sections.

A. Image Feature Extraction

For action recognition, the proposed method extracts features from RGB images in a form of human pose. Let I_i be the input image at the ith frame, and the jth joint at the ith frame is represented with 2D coordinate x_{ij} where there are n target joints. It is possible to directly regress $\{x_{ij}\}$ from the input image and it is easier to train the network to regress the score map of each joint. In order to train the network, we first need to define the ground truth of the score map $S(x_{ij})$ from the input image I_i. We assume a square score map, and the mapping function is defined as follows:

$$s^{uv}(x_{ij}) = \frac{|\Sigma|^{-\frac{1}{2}}}{2\pi} e^{-\left([u,v]^T - x_{ij}\right)^T |\Sigma|^{-\frac{1}{2}} \left([u,v]^T - x_{ij}\right)/2}, \quad (1)$$

where $s^{uv}(x_{ij})$ is the value of $S(x_{ij})$ at (u, v), and Σ is the variance of the multivariate Gaussian distribution. To sum up, the role of the human pose estimator $f(\cdot)$ can be represented as

$$f_j(I_i) = S(x_{ij}). \quad (2)$$

Our proposed model for joint estimation, $f(\cdot)$, is inspired by [12]. The input of the network is an RGB image and the output is a concatenated score map of each joint. In this way,

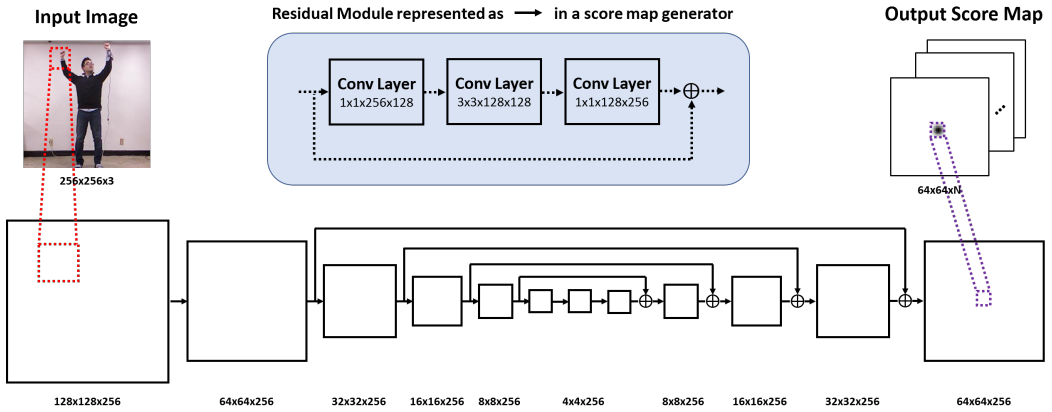


Fig. 2. A network structure of estimating human pose from a single image

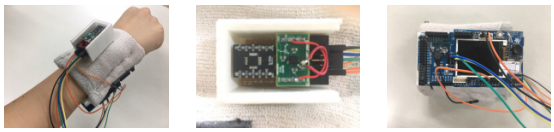
the network can utilize the global configuration of all joints so that training becomes easier. Before an image is fed into the network, normalization and resizing should be preceded. The input image is scaled down to have the pixel values ranging from 0 to 1 and resized to be $256 \times 256 \times 3$. Figure 2 shows the overall structure of human pose estimation used in this paper. When the pre-processed input passes through a convolution layer of which the filter size, stride, and output channel dimension are 7×7, 2, and 256, respectively, the input become a tensor whose size is $128 \times 128 \times 256$. Then it passes into residual modules [21] repeatedly with down-sizing and up-sizing, which are done with max pooling layers and nearest neighbor up-sampling layers. Here, a residual module consists of three convolutional layers with skip connections, and the filter sizes of the convolutional layers are $1 \times 1 \times 256 \times 128$, $3 \times 3 \times 128 \times 256$, and $1 \times 1 \times 128 \times 256$, respectively. The residual modules are represented as black solid arrows in Figure 2. Finally this output is fed into a 1×1 convolutional filter to predict the output score map whose size is $64 \times 64 \times 13$ where 13 is the number of joints in the human upper body. The estimation of joint position $\hat{x}_{ij}$ is obtained as follows:

$$\hat{x}_{ij} = \arg\max_{(u,v)} f_j^{uv}(I_i), \tag{3}$$

where f_j^{uv} is the value of estimated score map at (u, v).

B. IMU Feature Extraction

In this section, we propose a feature extraction method for action recognition using an IMU sensor mounted on the user's wrist as if he or she is wearing a smart watch. The sampling rate of an IMU sensor inside the commercial smart watches are not sufficiently high so that we manufactured a 6-DOF prototype sensor module with an accelerometer and a gyroscope both having maximum sampling rate of 1 kHz Figure 3. The input data is normalized before it is fed into the feature extractor. For normalization, we collected arbitrary sensor data form each sensor, accelerometer and gyroscope. We compute the mean and standard deviation of each sensor and normalize the input data by subtracting mean and divide it with standard deviation of a corresponding sensor.



(a) The prototype mounted on a wrist (b) A sensor module (c) A sensor board for data collection

Fig. 3. A hardware prototype of the wearable device with an IMU sensor

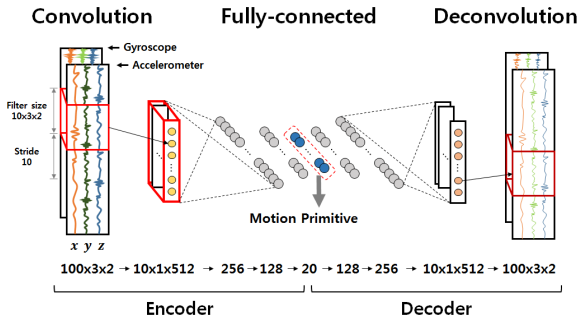


Fig. 4. A convolutional autoencoder network for extracting features from IMU sensor data.

We then simply reorganize the 6-DOF temporal data into three dimensional tensor to be fed into the feature extracting network. Transforming the temporal data into frequency domain like the other neural network based action recognition using IMU sensors is not performed due to the limitation of the human movement speed. Our prototype sensor has 1 kHz of sampling rate so that it can express the movement of having maximum frequency of 500 Hz which is abundant for human action recognition. Most of the human actions lie on the low frequency band so that the high frequency information is more likely to be noisy information. Also the transformation into other than time domain prevents to exploit the high capability of the sensor with high sampling rate.

Our proposed feature extractor for an IMU sensor is based on a convolutional autoencoder so that we can separate the proposed network into two parts, an encoder and a decoder. As shown in Figure 4, the encoder part consists of con-

volution layers and fully-connected layers and the decoder part consists of fully-connected layers and the deconvolution layers. The encoder generates the compressed representation of the input tensor and we call this representation as a motion primitive. It is the output of the motion primitive generator in Figure 1. In order to train the proposed network, we need to minimize the difference between reconstructed input and input data with l_2-norm.

We designed the feature extracting network in a way that the motion primitive contains not only the temporal information but also the axis-wise and sensor-wise correlation. In order to do so, we re-organized sensor data to be a shape of $100 \times 3 \times 2$ where 100 is the length of one motion segment, 3 is the number of the axis (x, y, z), and 2 is the number of sensors in an IMU (an accelerometer and a gyroscope). The first convolution layer has 512 filters whose size is $10 \times 3 \times 2$. After passing the first convolution layer, the input data becomes a tensor whose size is $10 \times 1 \times 512$. It is then connected to a fully-connected layer whose output size is 256, 128, and 20 respectively. In this way, the proposed network can extract a compressed motion primitive with 20 data points while the input data has 600 data points.

C. Action Recognition

Let X_t be the input vector at time t in a single action sequence, a whole sequence $\mathbf{X}$ can be represented as $\mathbf{X} = \{X_t | t = 1, 2, \ldots, T\}$ where T is the number of elements in a sequence which can be referred to as a sequence length. The upper body joints extracted from images and motion primitives extracted from the wrist-worn IMU sensor are the elements of sequential input respectively. For action recognition with a set of images and IMU data, a series of concatenated vectors of joint vectors and motion primitives are fed into an RNN as a sequence of vectors $\mathbf{X}$. The proposed classifier with an RNN which consists of LSTM units can effectively interpret the aspects of temporal change through several gates inside LSTM units without loosing information while time passes.

Ahead of feeding the input to the RNN, the joint vectors need a pre-processing. When the joint vectors are extracted, they are expressed in an absolute coordinate system so that it is necessary to be converted into a relative coordinate system. In this way we can get the joint features regardless of the target position. We set the hip to be the new origin of the coordinate system and scaled it down with a scaling factor which is the length of the spline. The length of the spline is defined as the distance between a hip joint and a neck joint.

As shown in Figure 5, the proposed action classifying RNN has two stacked layers with LSTM units. We set the size of a search window for action recognition to two seconds and the time interval between two successive input vectors in a sequence to 0.1 second so that there are 20 vectors in a single action search window. Instead of finding the start and end points of the actions, we successively derive output with 0.1 second of sliding window. In this way, we can get the classification result at every 0.1 second after two seconds which is the time for the first decision. In Figure 5,

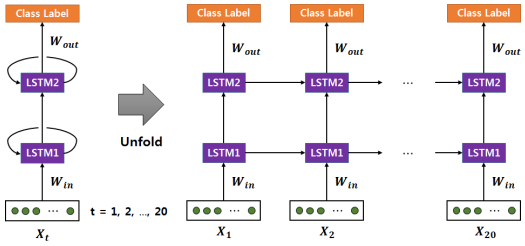


Fig. 5. A network structure of RNN based action classifier

an RNN returns a class label at every time step t in a single sequence $X_{t=1,\ldots,20}$ so that it is possible to get the class label at desired time, i.e., it is possible to handle an input with varying sequence lengths without making multiple networks. Although the proposed classification method can deal with the sequential data with varying sequence lengths, we fix the sequence length because of the difficulty of finding the exact start and end points of a sequence. This fixed length RNN for action classification is trained and tested on three different settings, *camera only*, *IMU only*, and *combined*. The network structure of these three setting is identical but the input vectors of the RNN so that a different size of fully-connected layer is added ahead of the input layer of the RNN in order to adjust the input vector size of the RNN.

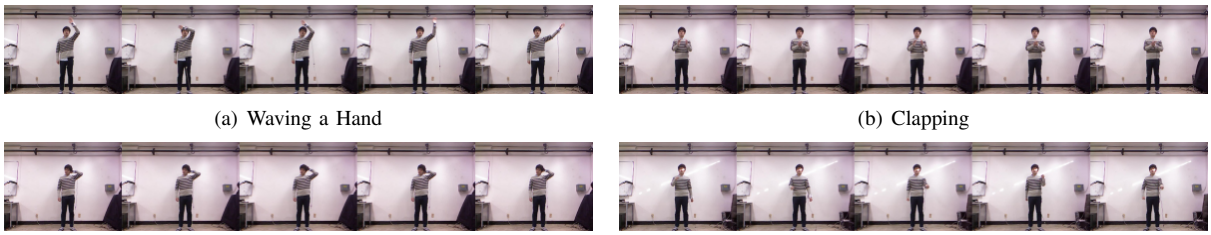
In order to share the same RNN for fair comparison among different types of inputs with varying size, we added an one-layered fully-connected layer whose output size is 128 ahead of the first LSTM layers to unify the input size of the RNN. The size of the motion primitive is 20 and that of joint feature is 26 so that the dimension of the fully-connected layer is 20×128 and 26×128 respectively. In case of the *combined* setting, size unifying using a fully-connected layer provides not only the fair comparison but also the balancing the influence of two different types of inputs. The concatenated vector of two vectors after a fully-connected layer is 256 which should be down-sized in order to be fed into the RNN. For this reason, an additional fully-connected layer whose dimension is 256×128 is added after the concatenation of two vectors. Fully-connected layers are illustrated as black solid lines in Figure 1.

IV. Experimental Results

In this section, we show the results of three different types of networks based on the input data type: images alone, IMU sensor data alone, and combined data with images and IMU sensor measurements. Each of them is trained and tested separately. Feature extraction methods for both images and inertial sensors are explained in previous sections. Extracted features are stacked through time with the fixed length and interval.

A. Data

We collect 20 different types of actions from five different individuals, three males and two females. Each action is performed for 50 repetitions with an IMU sensor at 1 kHz sampling rate on the right wrist and RGB images are taken



(a) Waving a Hand (b) Clapping

(c) Scratching a Head (d) Swinging Arms

Fig. 6. Image sequences of action collected at 0.1 second intervals

Fig. 7. Motion primitives from the collected actions (duration: two seconds). Each row is a motion primitive and 20 samples are stacked vertically.

every 0.1 second with a Kinect RGB-D camera. For training the upper body joint extracting network, we exploit the skeleton from the Kinect camera as the ground-truth. The MPU-6050 is used for the IMU module and it is attached to the Arduino DUE board (Figure 3). Figure 6 shows some sample motions of 20 actions. All 20 actions are everyday-life based actions focused on upper-body movement including waving a hand, pointing, yawning, clapping, stretching arms, scratching a head, shaking a bottle, drinking, putting hands in pockets, hitting a chest, picking up a stuff, crossing arms, fanning a face with a hand, making a hand shade, putting on a hood, picking up a phone, beckoning with a hand, and swinging arms with three different ways (to back and forth and to left and right with two different hand rotations). Figure 7 shows sequences of motion primitives of six actions from Figure 6. We can see that each action is separable with compressed motion primitives.

B. Evaluation

We examine the result from three different input data types: *camera only*, *IMU only*, and *combined* with the corresponding classifying network. The data from five individuals are separated into two groups as data for feature extraction and action classification. We separate remaining three individuals into two groups, two for training and one for testing. For training the upper body joint extracting network, we use the mean squared error as the loss function, and the network is trained based on RMSProp [22] with a learning rate of 2.5×10^{-4}. For training the motion primitive generator, we use the Adam Optimizer with a learning rate of 1.0×10^{-3}. An RNN for classification is also trained with the Adam Optimizer and the learning rate is set to 1.0×10^{-3} identically for three different input settings. As mentioned in Section , we slide a recognition window by 0.1 second on continuously collected dataset and it helps to augment the data. We have trained for 50 epochs and the batch size is set to 128. The result is obtained with cross-validation by

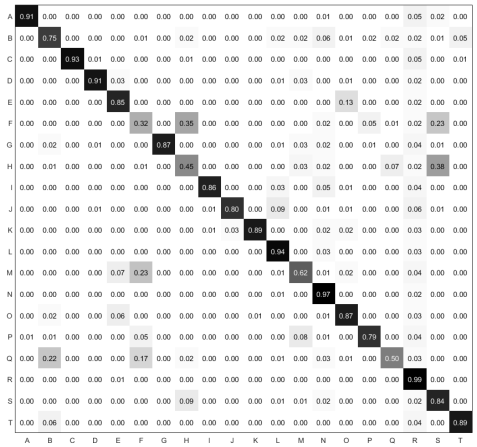


Fig. 8. The confusion matrix of sensor fused setting

* **A**: Waving a hand, **B**: Pointing, **C**: Yawning, **D**: Clapping, **E**: Stretching arms, **F**: Scratching a head, **G**: Shaking a bottle, **H**: Drinking, **I**: Swinging arms1, **J**: Swinging arms2, **K**: Swinging arms3, **L**: Putting hands in pockets, **M**: Hitting a chest, **N**: Picking up a stuff, **O**: Crossing arms, **P**: Fanning a face, **Q**: Making a hand shade, **R**: Putting on a hood, **S**: Picking up a phone, **T**: Beckoning with a hand.

setting two individuals as a training set and the other as a testing set.

The cross validated accuracy for action classification is 85.3 %, 67.1 %, and 86.9 % according to three different settings, *camera only*, *IMU only*, and *combined*, respectively. The result of the combined setting is the best, however, the gap between *image only* and *combined* is only 1.6 % which cannot be considered as a big improvement. Figure 8 shows the resulting confusion matrix when both images and sensor data are used for action recognition.

The reason for proposing a new method to combine the information from a camera and an IMU sensor is to alleviate the performance degeneration. The performance degeneration is mainly caused by the failure of joint estimation due to an occlusion and a noise. In order to verify that our algorithm can be a solution, we emulate the situation when there are some missing joints or highly inaccurate joint estimation.

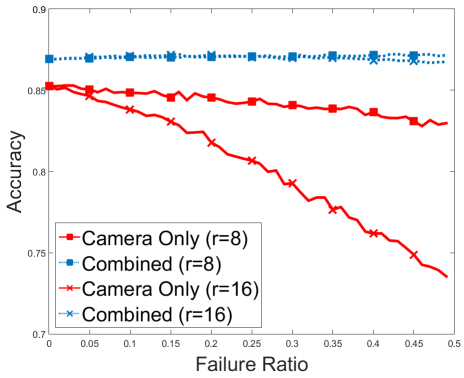


Fig. 9. The accuracy when the failure at joint estimation occurred. γ is the maximum pixel noise added to randomly selected joints.

We randomly select some joints and corrupt the 2D joint coordinates by adding some noise. The 2D joint coordinates are ranging from 0 to 63 and the noise is added with uniform random integer values except 0 in $[-4, 4]$ and $[-8, 8]$ (each noise level is represented as $\gamma = \{4, 8\}$ in Figure 9).

Figure 9 shows the accuracy comparison between the image alone case and sensor fusion case when there are some failures in joint estimation. As shown in Figure 9, the *camera only* shows performance drop as the failure ratio increases. However, the proposed method shows steady performance at each noise level and even a small performance improvement when the noise within 8 pixels is added. When the noise is added within 16 pixels which is the quarter of the input image size, our proposed sensor fusion method shows only 0.19 % of performance degeneration whereas the case that only images are used shows 11.72 % of a performance drop. The gap between these two cases gets bigger as the noise level increases. From the result, we can conclude that a single IMU sensor mounted on a wrist is not a solid device to recognize diverse actions as accurate as a camera but it can assist in a situation when a camera-based action recognition method fails to estimate the exact human pose.

V. Conclusion

In this paper, we have proposed an action recognition algorithm utilizing both images and inertial sensor data. We efficiently extract feature vectors using a CNN and perform the classification using an RNN. We have proposed a simple and powerful method to combine two different domain data by adding a fully-connected layers and derive good performance compared to cases when a single sensor is used. We have also verified that the proposed method shows robust performance against joint estimation failures. For future work, we plan to further improve the algorithm to detect the start and end points of an action. Moreover, we plan to apply the proposed method to more actions and individuals.

ACKNOWLEDGMENT

This work was supported by SNU-Samsung Smart Campus Research Center.

References

[1] H. Jhuang, J. Gall, S. Zuffi, C. Schmid, and M. J. Black, "Towards understanding action recognition," in *Proc. of the IEEE International Conference on Computer Vision (ECCV)*, 2013, pp. 3192–3199.

[2] F. Ofli, R. Chaudhry, G. Kurillo, R. Vidal, and R. Bajcsy, "Berkeley mhad: A comprehensive multimodal human action database," in *Proc. of the IEEE Workshop on Applications of Computer Vision (WACV)*. IEEE, 2013, pp. 53–60.

[3] R. Chavarriaga, H. Sagha, A. Calatroni, S. T. Digumarti, G. Tröster, J. d. R. Millán, and D. Roggen, "The opportunity challenge: A benchmark database for on-body sensor-based activity recognition," *Pattern Recognition Letters*, vol. 34, no. 15, pp. 2033–2042, 2013.

[4] F. J. Ordóñez and D. Roggen, "Deep convolutional and lstm recurrent neural networks for multimodal wearable activity recognition," *Sensors*, vol. 16, no. 1, p. 115, 2016.

[5] D. Morris, T. S. Saponas, A. Guillory, and I. Kelner, "Recofit: using a wearable sensor to find, recognize, and count repetitive exercises," in *Proc. of the ACM Conference on Human Factors in Computing Systems*, 2014.

[6] A. Krizhevsky, I. Sutskever, and G. E. Hinton, "Imagenet classification with deep convolutional neural networks," in *Proc. of the Advances in Neural Information Processing Systems*, 2012.

[7] S. Hochreiter and J. Schmidhuber, "Long short-term memory," *Neural Computation*, vol. 9, no. 8, pp. 1735–1780, 1997.

[8] K. Cho, B. Van Merriënboer, D. Bahdanau, and Y. Bengio, "On the properties of neural machine translation: Encoder-decoder approaches," *arXiv preprint arXiv:1409.1259*, 2014.

[9] Y. Yang and D. Ramanan, "Articulated pose estimation with flexible mixtures-of-parts," in *Proc. of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2011.

[10] A. Toshev and C. Szegedy, "Deeppose: Human pose estimation via deep neural networks," in *Proc. of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2014.

[11] S.-E. Wei, V. Ramakrishna, T. Kanade, and Y. Sheikh, "Convolutional pose machines," in *Proc. of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2016.

[12] A. Newell, K. Yang, and J. Deng, "Stacked hourglass networks for human pose estimation," in *Proc. of the European Conference on Computer Vision (ECCV)*, 2016.

[13] H. Xue, S. Chen, and Q. Yang, "Structural support vector machine," *Advances in Neural Networks*, pp. 501–511, 2008.

[14] N. Dalal and B. Triggs, "Histograms of oriented gradients for human detection," in *Proc. of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2005.

[15] D. Roggen, A. Calatroni, M. Rossi, T. Holleczek, K. Förster, G. Tröster, P. Lukowicz, D. Bannach, G. Pirkl, A. Ferscha, *et al.*, "Collecting complex activity datasets in highly rich networked sensor environments," in *Proc. of the IEEE International Conference on Networked Sensing Systems (INSS)*, 2010.

[16] J. A. Ward, P. Lukowicz, G. Troster, and T. E. Starner, "Activity recognition of assembly tasks using body-worn microphones and accelerometers," *IEEE transactions on Pattern Analysis and Machine Intelligence*, vol. 28, no. 10, pp. 1553–1567, 2006.

[17] X. Su, H. Tong, and P. Ji, "Activity recognition with smartphone sensors," *Tsinghua Science and Technology*, vol. 19, no. 3, pp. 235–249, 2014.

[18] O. D. Lara and M. A. Labrador, "A survey on human activity recognition using wearable sensors." *IEEE Communications Surveys and Tutorials*, vol. 15, no. 3, pp. 1192–1209, 2013.

[19] L. C. Jatoba, U. Grossmann, C. Kunze, J. Ottenbacher, and W. Stork, "Context-aware mobile health monitoring: Evaluation of different pattern recognition methods for classification of physical activity," in *Proc. of the IEEE Conference on Engineering in Medicine and Biology Society (EMBS)*, 2008.

[20] A. Alvarez-Alvarez, J. M. Alonso, and G. Trivino, "Human activity recognition in indoor environments by means of fusing information extracted from intensity of wifi signal and accelerations," *Information Sciences*, vol. 233, pp. 162–182, 2013.

[21] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *Proc. of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2016.

[22] T. Tieleman and G. Hinton, "Lecture 6.5-rmsprop: Divide the gradient by a running average of its recent magnitude," *COURSERA: Neural networks for machine learning*, vol. 4, no. 2, pp. 26–31, 2012.