Weakly Supervised Graph Convolutional Neural Network for Human Action Localization

Daisuke Miki[1,2] Shi Chen[1] Kazuyuki Demachi[1]
[1]The University of Tokyo, Japan
[2]Tokyo Metropolitan Industrial Technology Research Institute, Japan
miki.daisuke@iri-tokyo.jp, shichen@g.ecc.u-tokyo.ac.jp, demachi@nuclear.jp

Abstract

Skeleton-based human action recognition from video sequences is currently an active topic of research. Conventionally, human action recognition is performed after conducting feature extraction on a given spatial-temporal representation of a human pose by using statistical methods or deep learning methods. The spatial and temporal features are globally evaluated by a classifier and used to determine which action is closest. However, the conventional methodology does not identify the temporal location of the action that determines the classification. To address this problem, we propose a skeleton-based human action recognition and localization method using weakly supervised graph convolutional neural networks, which are both spatially and temporally connected. In this method, human action localization is accomplished using time series data of human joint positions as input and then applying regression to find an expected value for each action at each time frame. Our weakly supervised training is based on multiple-instance learning inspired by deep ranking, and we devise a loss function so that high scores can be spontaneously learned for temporally important time frames. In this paper, we first explain the network architecture and then present a multiple-instance learning method for its optimization. In the experiment, we performed localization and classification of human actions by using this method and confirmed the temporal localization efficacy of the method.

1. Introduction

Human action recognition from video sequences is currently being actively studied, and it has attracted increasing attention in computer vision fields such as video surveillance, human-computer interaction, and entertainment. To recognize human action, various methods using RGB images, depth images, and human skeletons have been proposed. For RGB images, methods for extracting spatial-temporal features such as optical flow [25, 27, 31] and silhouettes [11, 10] have been proposed. Although these methods can be applied to various situations, they struggle to yield robust results in the presence of a background, noise, and other disturbances. The use of a depth image captured by a stereo camera or an infrared camera can easily separate the background and is advantageous for the analysis of human action. Recently, with the emergence of hardware such as Microsoft Kinect, skeleton-based human action recognition techniques have been actively researched. When applied to human action analysis and compared with RGB and depth images, skeletal information is a high-level feature, is the simplest in terms of information, and is lightweight. In addition, the skeletal information is more robust to subject rotation as compared with RGB or depth image based methods.

The skeleton-based human action analysis method is focused on human action that can be expressed as a combination of two- or three-dimensional spatial time-series data of human poses. Conventionally, after performing feature extraction on a given spatial-temporal representation of a human pose, by using statistical methods or deep learning methods, the spatial and temporal features are globally evaluated by a classifier, which classifies the action using an a priori dataset of actions. However, not all temporal locations of a human action are necessarily important for identifying it. For example, when recognizing the “make a call on a mobile phone” action, the “move hand close to ear” action is important for determining the action. However, the “take mobile phone out of pocket” action is common to other actions, so it is insufficient to determine the action. If it is possible to localize where the important action is included, it would be useful for advancing video surveillance or other applications. Thus far, various human action datasets [15, 34, 29, 2, 21, 7, 16] have been proposed for research in human action analysis. Each of these is composed of tens to hundreds of frames of video and skeletal information, each with a single label to explain its action. However, it has been difficult to perform temporal action localization using such datasets, because when annotating an instance in high-dimensional time series data, such as human action, it is difficult to quantify what information is necessary in a given frame to determine whether the action is occurring.

To solve these problems, we propose a weakly supervised graph convolutional network (WST-GCN) that enables temporal human action localization that recognizes actions and localizes important time frames. We devise a loss function to optimize the network using a singly labeled human action dataset so that high scores can be learned spontaneously for temporally important video frames. The loss function is able to recognize and localize multiple classes of actions. We also adopt multiple-instance learning inspired by learning to rank [26].

In this paper, we first explain the network architecture and then explain a weakly supervised learning method for training the network. We then apply single and multiple-class action classification.

2. Related works

2.1. Skeleton-based human action recognition

In recent years, many human action analysis methods using human pose time series data have been proposed. Hand-crafted feature quantities [34, 8, 20, 28, 33] and methods using deep learning [25, 27, 31, 19, 4, 18, 35, 13, 32] have been proposed for human action recognition. Hand-crafted feature-based methods include temporal covariance matrices of skeletal joint [8], modeling of human behavior as a curve in a Lie group [28], spatial-temporal Naive-Bayes Nearest-Neighbor [33], etc. Deep learning based methods include convolutional neural networks (CNNs), recurrent neural networks (RNNs), long short-term memory (LSTM), and GCNs. Liu *et al*. [19] proposed representing joint positions by using RGB image maps and processing them with a CNN-based model to extract and fuse deep features. Du *et al*. [4] introduced an end-to-end hierarchical RNN model to represent the temporal dynamics of human structures and joints. Liu *et al*. [18] designed a 2D spatial-temporal LSTM framework to simultaneously explore hidden sources of behavioral context information in the spatial and temporal domains. They also introduced a trust gate mechanism [17] to handle inaccurate 3D coordinates provided by depth sensors for skeletal joints.

More recent studies focus on spatial and temporal features of the skeleton sequence, Yan *et al*. [35] and Li *et al*. [13] proposed a human action analysis method using spatially and temporally connected GCNs (ST-GCNs). Furthermore, several works demonstrating the improved performance of ST-GCN have been reported [14, 24, 23, 22]. Li *et al*. [14] proposed an A-link inference module and an encoder-decoder structure called actional-structural graph convolution network (AS-GCN), which combines actional-structural graph convolution and temporal convolution as a basic building block. Si *et al*. [24] improved ST-GCN including an attention enhanced graph convolutional LSTM (AGC-LSTM) layer and improved its classification accuracy. These ST-GCN based methods achieve state-of-the art performance in human action recognition, and all methods use hundreds of frames as input but provide only a single score for each action; localization is not considered.

2.2. Multiple-instance learning for ranking

Multiple-instance learning inspired by learning to rank can be used to estimate the relative score, rather than the absolute score, by using weakly labeled data. Joachims *et al*. [9] proposed a rank-SVM and report improvements in search engines. Recently, deep ranking has been used in computer vision applications, and it has reported leading edge performance in various fields: feature extraction [30], image generation [6], person identification [3], place recognition [1], and video summarization [5].

Similarly, Sultani [26] proposed an anomaly detection method inspired by learning to rank and degree of anomaly that applied a multiple-instance learning model to video sequences in which it is difficult to annotate a ground truth value. These methods are similar in nature to human action localization in dealing with time series data that is difficult to annotate. By imitating the dataset for anomaly detection, each instance in the dataset has two values, positive or negative, and it can perform single-class action classification by training on a dataset including or not including the specific action. However, in this method, the loss function is only applicable for binary classification (positive or negative), and it cannot be applied as-is to multiple-class classification problems. In our research, this idea is applied to multiple-class human action recognition and localization by improving the loss function.

3. Action localization ST-GCN

In this section, we first describe an overview of the proposed human action recognition method and the structure of the GCN. We then explain how to train it with weakly supervised learning. After describing the method applied to single-class human action recognition, we apply it to multiple-class action recognition.

3.1. Overview of proposed method

An overview of the proposed method is depicted in Figure 1. In this method, we first perform feature extraction on the human-pose time series data using a GCN. Next, human action recognition and localization are performed by a one dimensional CNN, which outputs human action localization as an expected value for each time frame. In action recognition using pose information, three-dimensional data for

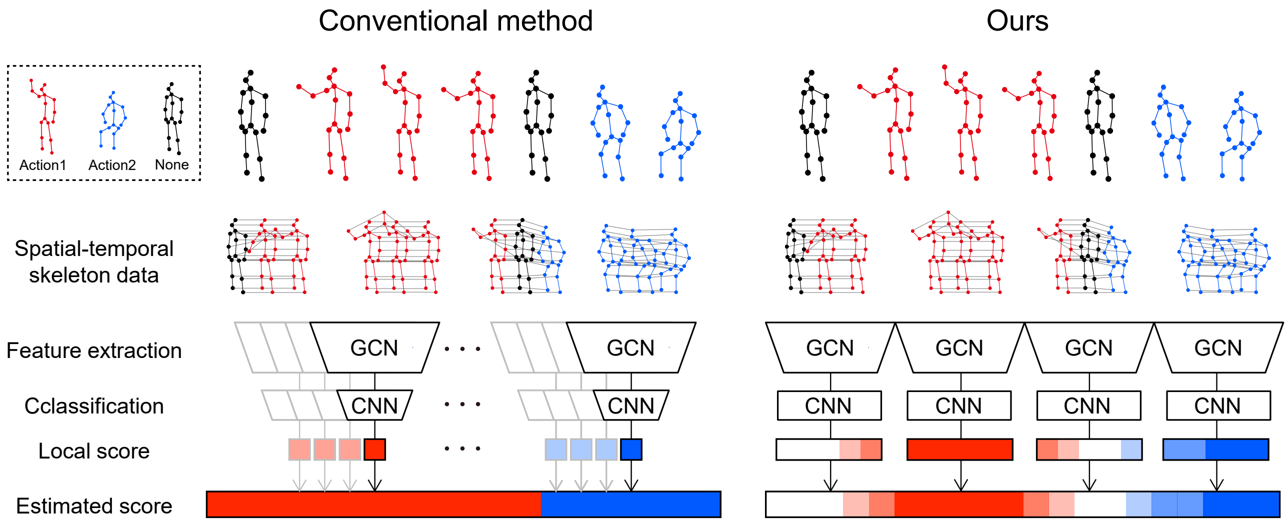

Figure 1. Visualization of human action recognition method. (left) Conventional method using ST-GCN: multiple frames are input to the network and output a single frame score. (right) Proposed method: multiple input frames are input and output multiple frame scores.

the pose is provided in a time series along with spatial relationships between different joints in the same frame and between different frames. The temporal relationships between the same joints are both important in human action analysis using skeletal information. Therefore, in this research, we adopted an ST-GCN for feature extraction. The output of a general ST-GCN [35] is an $N \times T$ dimensional matrix. Here, N is the number of action classes and T is the time length. However, the classifier CNN, installed after the ST-GCN, transforms the outputs into an N dimensional vector, because it is activated with a softmax function and trained on single-class labels with cross-entropy loss. Meanwhile, the proposed network outputs an $N \times T$ dimensional matrix directly through weakly supervised learning based on ranking loss. It enables us to use regression to find the expected importance of each time frame for identifying the desired action.

3.2. Spatial-temporal graph convolution

The idea of using ST-GCNs for spatial-temporal feature extraction from human action was based on the method of [35]. The input to an ST-GCN expresses a human pose by using a spatial-temporal graph, in which each node corresponds to a human joint at each time frame, and each edge corresponds to a spatial-temporal connection between nodes. Spatial connections constitute graphs that are represented by human joints in a single frame. Here, the spatial connections of the graph represent the natural connections between joints of the human body. The temporal connections are configured by connecting corresponding joints across a series of frames. A graph having T frames with a skeletal graph having I nodes in a single frame is represented by $G = (V, E)$. Here, $V = \{v_{ti} | t = 1, ..., T, i = 1, ..., I\}$. For the spatial domain, the graph convolution is expressed as

$$f_{out}(v_{ti}) = \sum_{v_{tj} \in \mathcal{B}(v_{ti})} \frac{1}{Z_{ti}(v_{tj})} f_{in}(v_{tj}) \cdot \mathbf{w}(l_{ti}(v_{tj})). \quad (1)$$

Here, f_{in} and f_{out} are input and output features, respectively. $\mathcal{B}(v_{ti})$ represents the set of neighboring nodes for v_{ti}, where one distance neighborhood of the object node v_{ti} is considered. The weight for computing the inner product using the input features is w. Because the number of weight vectors is fixed, $\mathcal{B}(v_{ti})$ is divided into three subsets: (1) the target node, (2) nodes that are closer to the center of gravity (centripetal nodes), and (3) the remaining nodes (centrifugal nodes). l_{ti} is a function that maps each node in the vicinity of v_{ti} to its subset label. The subset radix $Z_{ti}(v_{tj})$ is used as a normalization term to ensure that different subsets do not break the output balance. In the ST-GCN in Yan's implementation [35] of the graph convolution described in Kiph *et al.* [12],

$$\mathbf{f_{out}} = \sum_j \mathbf{\Lambda}_j^{-\frac{1}{2}} \mathbf{A}_j \mathbf{\Lambda}_j^{-\frac{1}{2}} \mathbf{f_{in}} W_j \quad (2)$$

is adopted. $\mathbf{A}$ is an $N \times N$ adjacency matrix. To implement the ST-GCN, equation (1) is converted to the feature $\mathbf{f_{in}}$ and $\mathbf{f_{out}}$, and the input feature $\mathbf{f_{in}}$ is expressed as a tensor with dimension (N, T, C), where C is the number of input channels. The adjacency matrix is divided into three matrices: $\mathbf{A_0}$, $\mathbf{A_1}$, and $\mathbf{A_2}$. These represent self-connections, t centripetal-node connections, and centrifugal-node connections respectively. Each matrix represents a subset of the connections. W_j represents a weight matrix, and the weight vectors for a plurality of the output channels are stacked.

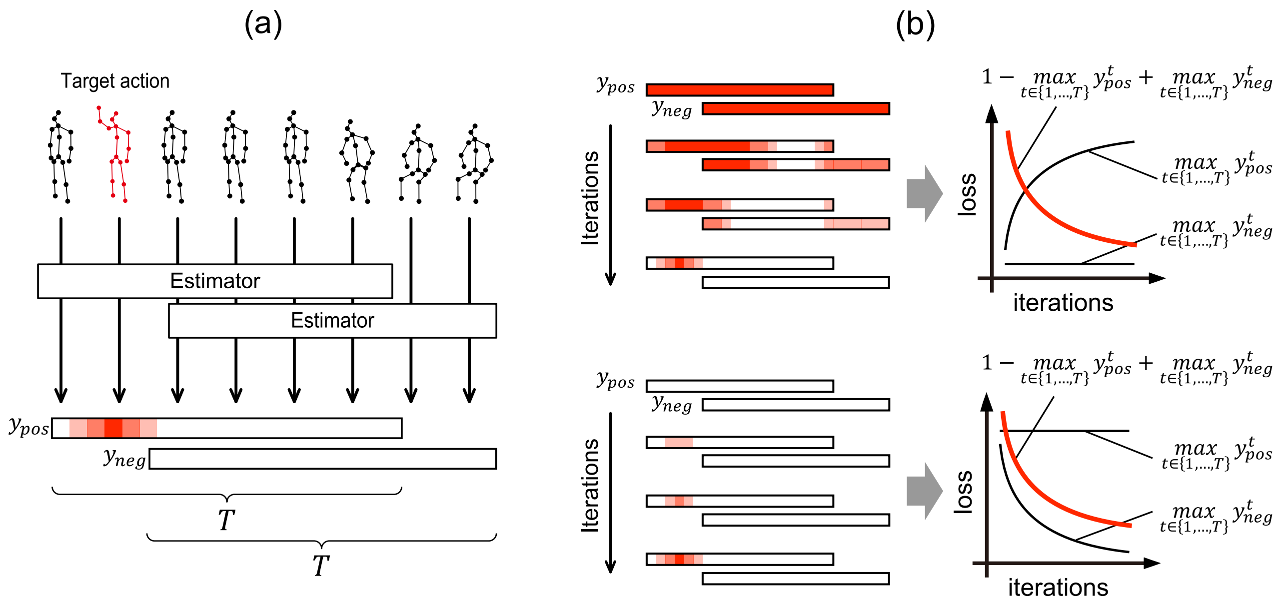

Figure 2. Proposed network optimization method for action localization. (a) Action localization for time frames that include and do not include a target action. (b) Minimization of loss function $\mathcal{L}$, when initial overall estimated scores are (top) high and (bottom) low.

3.3. Action localization

3.3.1 Single-class action localization

The proposed model estimates the importance score for determining the action from human pose time series data as a regression problem, as it is difficult to label the human pose data quantitatively for each time frame. Figure 2 is an overview of the model optimization method with our weakly supervised training method. First, the human pose data is classified for each class according to whether the target action is included (positive) or not included (negative). Next, the frames containing a specific action are given a higher score than those that do not contain the specific action. Here, we use ranking loss to encourage a higher score for positive frames than for negative ones:

$$\max_{t\in\{1,\dots,T\}} y^t_{pos} > \max_{t\in\{1,\dots,T\}} y^t_{neg} \tag{3}$$

where y^t_{pos} and y^t_{neg} are the predicted positive and negative scores of time frame t. It is unknown which part of the sequence contains important information for determining the action, so we use only the two frames that have the highest score from each of the positive and negative data:

$$\mathcal{L} = \max(0, 1 - \max_{t\in\{1,\dots,T\}} y^t_{pos} + \max_{t\in\{1,\dots,T\}} y^t_{neg}) + \lambda \tag{4}$$

where λ is a regularization term to stabilize training,

$$\lambda = \mu_1 \sum_{t=1}^{T-1} (y^t_{pos} - y^{t+1}_{pos}) + \mu_2 \sum_{t=1}^{T} y^t_{pos}. \tag{5}$$

The two terms are a smoothness term and a sparsity term, and μ_1 and μ_2 are parameters for controlling the strength of each type of regularization.

3.3.2 Multiple-class action localization

By training multiple-class human action localizers and using them in parallel, multiple-class action localization becomes possible. However, this is not preferable owing to the memory requirements and the calculation time. Using the expressive power of GCNs of the same size as those used for single-class localization, it is possible to localize multiple actions. To utilize GCNs for multi-class action localization, we expanded the output dimension of our model to $N \times T$ and proposed a new loss function

$$\mathcal{L} = \max\left(0, \sum_{k=1}^{2} \sum_{n=1}^{N} (\phi_{kn} - \psi_{kn} Y_{kn})\right) + \lambda. \tag{6}$$

Here, Y_{kn} denotes the maximum value of output score from the tth frame, and the kth indexed and nth action data randomly selected pair of indexes included in the training dataset

$$Y_{kn} = \max_{t\in\{1,\dots,T\}} y^t_{kn}. \tag{7}$$

ϕ_{kn} and ψ_{kn} are N dimensional labels indicating whether each instance includes the action. We define ϕ_{kn} and ψ_{kn} as

$$\phi_{kn} = \begin{cases} 1 & \text{if } n\text{th action is included,} \\ 0 & \text{otherwise,} \end{cases} \quad (8)$$

$$\psi_{kn} = \begin{cases} 1 & \text{if } n\text{th action is included,} \\ -1 & \text{otherwise.} \end{cases} \quad (9)$$

We use λ as a regularization term:

$$\lambda = \lambda_1 + \lambda_2 + \lambda_3, \quad (10)$$

$$\lambda_1 = \mu_1 \sum_{k=1}^{2} \sum_{k=1}^{N} \sum_{t=1}^{T} (y_{kn}^{t} - y_{kn}^{t+1}), \quad (11)$$

$$\lambda_2 = \mu_2 \sum_{k=1}^{2} \sum_{k=1}^{N} \sum_{t=1}^{T} y_{kn}^{t}, \quad (12)$$

$$\lambda_3 = -\mu_3 \sum_{k=1}^{2} \sum_{k=1}^{N} \phi_{kn}^{t} \log \frac{\exp(Y_{kn})}{\sum_m \exp(Y_{km})}. \quad (13)$$

It is represented similarly as in the single-class case, where the first term represents smoothing and the second term represents sparsity. The third term represents cross entropy loss, which prevents the score for the negative sample from becoming too large.

3.4. Model optimization

For the architecture of the ST-GCN, we adopted the method of Yan *et al*. [35]. The ST-GCN has 9 layers. The first 3 layers have 64 channels. The next 3 layers have 128 channels and the last 3 layers have 256 channels. Pooling was performed so that the width in the temporal domain was pooled in the fourth and seventh layers. After each GCN layer, we perform dropout with a probability of 0.1. In the original ST-GCN, $F = 256$ dimensional feature vectors are extracted temporally and spatially with respect to GCN via average pooling to obtain a score for N dimensional score as output. In the proposed method, spatial information is preserved. After extracting the feature quantity of $T \times F$ dimensions and applying one dimensional convolution, the $T \times N$ dimensional score is output. These networks were optimized using stochastic gradient descent with a learning rate of 10^{-4}. The regularization term μ_1 is 10^{-5}, μ_2 is 10^{-2}, and μ_3 is 10^{-2}. After 20 epochs, the learning rate is reduced by multiplying with 10^{-1}. For data augmentation, the skeletal data are rotated $-30°$ to $30°$ around an axis perpendicular to the floor surface. In addition, to make our model robust to differences in body size, we performed a scale transformation in the range of 0.9 to 1.1 in three-dimensional space. Furthermore, Gaussian noise was added to the data to simulate measurement noise for the joint positions. To verify frame rate change robustness, 0 to 10% of the frames were randomly removed. The TITAN V GPU was used for training and experiments.

4. Experiments

In this section, qualitative and quantitative evaluations were performed on publicly available human action datasets to evaluate our method. In the experiment, the ability to perform action localization and classification was compared with ST-GCN and other related methods.

4.1. Dataset

Three datasets were utilized for evaluating our method. These datasets include not only human skeleton data but also RGB images and depth images. However, in this experiment, only skeleton information was used.

UTD-MHAD dataset: The UTD MHAD dataset [2] was captured with Microsoft Kinect. In this dataset, 27 class actions were performed four different times by eight subjects. Each skeleton is represented by three-dimensional coordinates of 20 points of human joints. In the evaluation, according to the method of [2], the data were divided into 1, 3, 5, and 7, and the data of subjects were divided into 2, 4, 6, and 8; the former was used for training and the latter was used for testing. Zero padding was applied so that all data contained 128 frames.

SYSU datasets The SYSU [7] datasets were also captured with the Kinect, and 12 class actions were performed by 40 subjects. This dataset was utilized to represent data that are not included in UTD-MHAD, that is, negative data, and was used to confirm the behavior of this method to negative data.

NTU RGB+D datasets To conduct experiments on a larger dataset, we used the NTU RGB+D [21] dataset. This dataset was captured by the Kinect V2 and consists of more than 56,000 frames of video data. It includes 60 action classes performed by 40 subjects, and each human pose is represented by 25 points of human joints. The providers of this dataset recommend two evaluation methods as benchmarks. The first was the cross-subject benchmark, where the training and test data included 40,320 and 16,560 instances, respectively. In this evaluation method, learning and testing were performed in subsets. The second recommendation was the cross-view benchmark, where the training and test data were divided to contain 37,920 and 18,960 instances, respectively; those captured by two cameras in the same subset were training data, and the rest were testing data.

4.2. Qualitative results

The action localization results are presented in Figure 3. In the experiment, training of the WST-GCN was performed using the UTD-MHAD dataset and it was confirmed whether the actions were included in the testing set. To confirm the ability of temporal action localization, annotations were added to frames that were important for determining

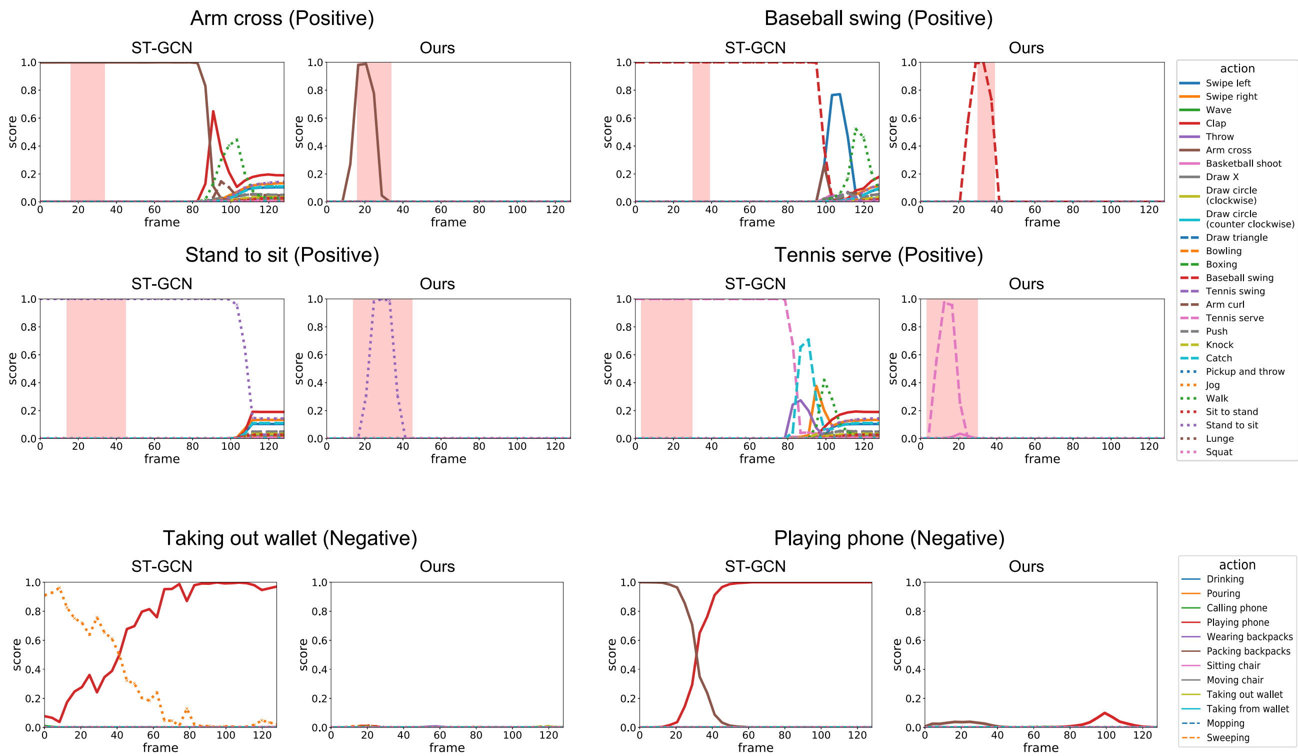

Figure 3. Estimatied score according to softmax classifier, and the proposed method: (upper) Skeleton sequences of “Arm cross”, “Baseball swing”, “Tennis serve”, and “Stand to sit” in UTD-MHAD datasets are used as positive test data. (bottom) and “Taking out wallet” and “Playing phone” in SYSU are used as negative test data.

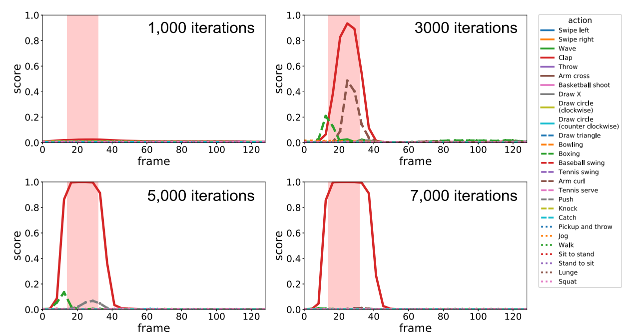

Figure 4. Evolution of estimated score over iterations. Colored windows represent ground truth.

the behavior of each test data in the UTD-MHAD dataset (not given to training data). This annotation was done manually. Although it required approximately 6 hours, the main feature of the proposed method is that learning can be localized automatically without the need for the annotation work. As shown in Figure 3, our method yielded high scores for the frames that are important for determining the action and low scores in the other frames. Furthermore, the desired response to negative data: a sufficiently low value was output. Figure 4 shows the relationship between the number of iterations used for training and the output value of the proposed model. With 3,000 iterations, a high score was produced for the negative class actions, but as the number of iterations was increased, the score for negative class actions declined, while a high score was maintained for the positive class actions.

4.3. Quantitative result on action classification

To evaluate our method, we first confirmed ability to classify multiple-class actions by using the UTD-MHAD, and NTU RGB+D datasets.

4.3.1 Experiments on the UTD-MHAD dataset

When calculating action recognition accuracy, the expected score in each frame output is summed over the entire skeleton sequence. The detected action is calculated as

$$\text{detected action} = \underset{n \in \{1,\ldots,N\}}{\arg\max} \sum_{t=1}^{T} y_n^t. \tag{14}$$

A comparison of recognition accuracy with the latest method and a confusion matrix are shown in Table 1 and Figure 5.

Here, the ST-GCN paper [35] was not evaluated by UTD-MHAD, so it was newly implemented and tested. In

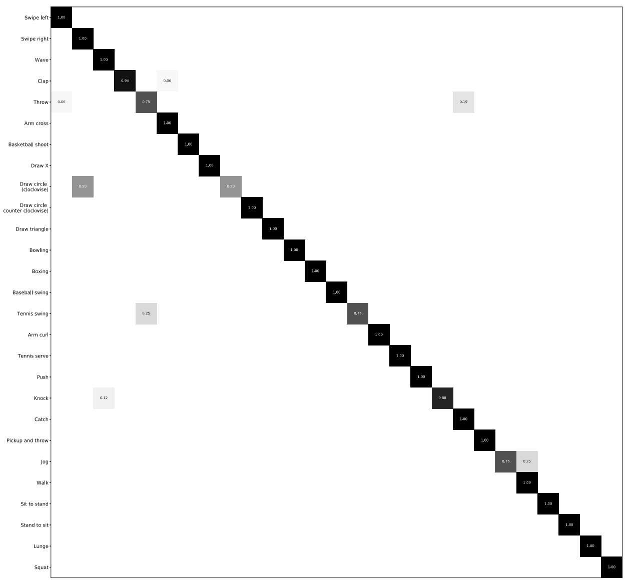

Figure 5. Confusion matrix on UTD-MHAD dataset.

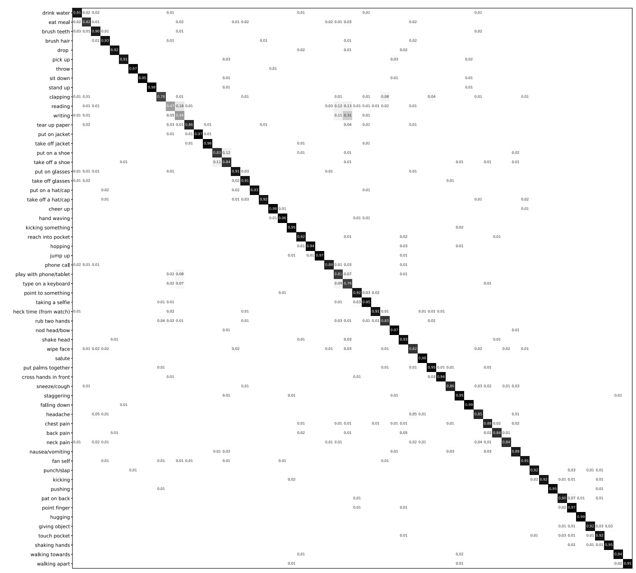

Figure 6. Confusion matrix on NTU RGB+D dataset.

particular, when compared with ST-GCN, which is the comparison that is the crux of this study, equivalent classification accuracy was confirmed.

Table 1. Comparison of classification accuracy on Two datasets.

Methods	UTD-MHAD	NTU RGB+D	
		CS	CV
T-GCN [35]	–	81.5	88.3
ST-GCN (Our implementation)	94.2	79.5	87.3
WST-GCN (Ours)	94.6	79.9	89.8

4.3.2 Experiments on the NTU RGB+D Dataset

To conduct experiments on a larger dataset, we used the NTU RGB+D dataset. NTU RGB+D dataset. The proposed method yielded an equivalent classification accuracy to that of ST-GCN, demonstrating that human action can be properly classified even in a larger dataset. The ST-GCN that we implemented was slightly lower than that in the literature [35], but almost the same result was achieved, and the appropriateness of the evaluation was confirmed.

4.4. Quantitative result on action classification

To confirm the accuracy related to temporal action localization, the manually annotated UTD-MHAD dataset was utilized. The recognition accuracy was evaluated by mean average precision (mAP), varying the intersection over union (IoU) threshold over the range of $0.1 - 0.5$.

Figure 7 shows part of the localization results and ground truth annotation in the UTD-MHAD dataset. Table 2 shows the quantitative evaluation results of recognition accuracy regarding localization. As shown in Figure 7, ST-GCN was able to recognize the appropriate action but was unable to localize each action. This failure is explained by the fact that softmax activation is installed before the output of ST-GCN, produces some high scores even for information that is not necessary to determine the action. On the other hand, in the proposed method, localization was performed appropriately. The two methods yielded almost identical results in the classification task, but the proposed method was advantageous in terms of the action localization task.

Table 2. Comparison of action localization result on UTD-MHAD dataset measured by mAP(%) at different IoU thresholds.

	mAP@IoU				
IoU threshold	0.1	0.2	0.3	0.4	0.5
ST-GCN (Our implementation)	24.2	20.5	12.4	4.0	0.7
WST-GCN (Ours)	72.3	69.5	59.7	44.1	25.0

5. Conclusions

In this study, skeleton-based human action localization was achieved through weakly supervised training using ranking loss inspired by deep ranking. By devising loss function, our method localized multiple classes of human actions. In the evaluation, it was possible to classify human

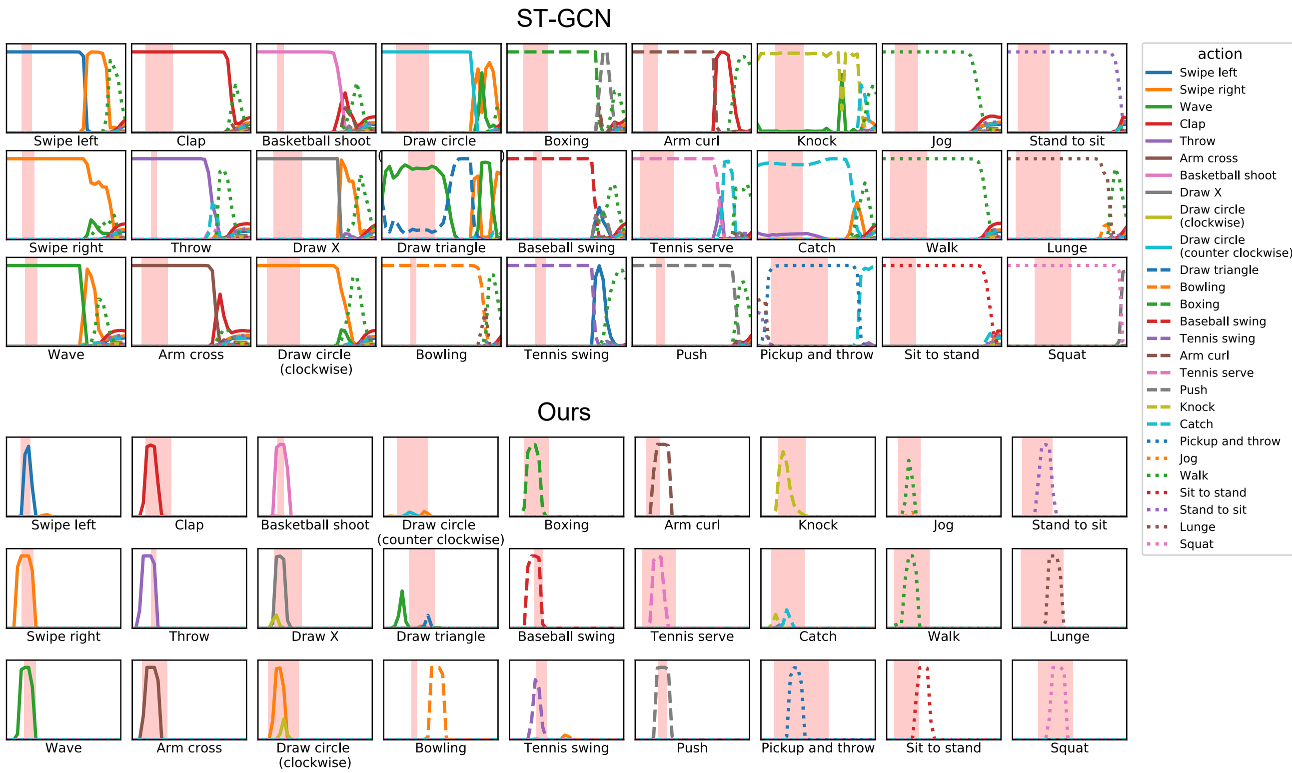

Figure 7. Example of action localization on the UTD-MHAD dataset estimated by (a) ST-GCN and (b) our method.

action by training the WST-GCN that estimates the degree to each action of the subjects. Temporal localization of human action is possible while still maintaining the equivalent classification accuracy. These results suggest the proposed method is effective in enhancing video surveillance. In addition, because training does not require information on the temporal location and the degree of action, it is possible to detect actions with an unclear definition, such as unnatural human behavior.

Acknowledgments

This work was supported by Japan Society for the Promotion of Science (JSPS) Grants-in-Aid for Scientific Research Grant Numbers JP19K20310.

References

[1] R. Arandjelovic, P. Gronat, A. Torii, T. Pajdla, and J. Sivic. NetVLAD: CNN Architecture for Weakly Supervised Place Recognition. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 40(6):1437–1451, 2018.

[2] 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 *2015 IEEE International conference on image processing (ICIP)*, pages 168–172, 2015.

[3] S. Ding, L. Lin, G. Wang, and H. Chao. Deep feature learning with relative distance comparison for person re-identification. *Pattern Recognition*, 48(10):2993–3003, 2015.

[4] Y. Du, W. Wang, and L. Wang. Hierarchical recurrent neural network for skeleton based action recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 1110–1118, 2015.

[5] M. Elfeki and A. Borji. Video Summarization Via Actionness Ranking. In *2019 IEEE Winter Conference on Applications of Computer Vision (WACV)*, pages 754–763, 2019.

[6] M. Gygli, Y. Song, and L. Cao. Video2GIF: Automatic Generation of Animated GIFs from Video. In *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 1001–1009, 2016.

[7] J.-f. Hu, W.-s. Zheng, J. Lai, and J. Zhang. Jointly learning heterogeneous features for rgb-d activity recognition. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 39(11):2186–2200, 2017.

[8] M. E. Hussein, M. Torki, M. A. Gowayyed, and M. El-Saban. Human action recognition using a temporal hierarchy of covariance descriptors on 3D joint locations. *IJCAI International Joint Conference on Artificial Intelligence*, pages 2466–2472, 2013.

[9] T. Joachims. Optimizing search engines using clickthrough data. In *Proceedings of the eighth ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 133–142, 2002.

[10] W. Khan, Zafar Aliand Sohn. A hierarchical abnormal human activity recognition system based on r-transform and kernel discriminant analysis for elderly health care. *Computing*, 95(2):109–127, 2013.

[11] Z. A. Khan and W. Sohn. Abnormal human activity recognition system based on r-transform and kernel discriminant technique for elderly home care. *IEEE Transactions on Consumer Electronics*, 57(4):1843–1850, 2011.

[12] T. N. Kipf and M. Welling. Semi-supervised classification with graph convolutional networks. *arXiv preprint arXiv:1609.02907*, 2016.

[13] C. Li, Z. Cui, W. Zheng, C. Xu, and J. Yang. Spatio-Temporal Graph Convolution for Skeleton Based Action Recognition. *Thirty-Second AAAI Conference on Artificial Intelligence*, pages 3482–3489, 2018.

[14] M. Li, S. Chen, X. Chen, Y. Zhang, Y. Wang, and Q. Tian. Actional-structural graph convolutional networks for skeleton-based action recognition. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 3595–3603, 2019.

[15] W. Li, Z. Zhang, and Z. Liu. Action recognition based on a bag of 3d points. In *2010 IEEE Computer Society Conference on Computer Vision and Pattern Recognition-Workshops*, pages 9–14, 2010.

[16] J. Liu, A. Shahroudy, M. L. Perez, G. Wang, L.-Y. Duan, and A. K. Chichung. Ntu rgb+ d 120: A large-scale benchmark for 3d human activity understanding. *IEEE transactions on pattern analysis and machine intelligence*, 2019.

[17] J. Liu, A. Shahroudy, D. Xu, A. C. Kot, and G. Wang. Skeleton-Based Action Recognition Using Spatio-Temporal LSTM Network with Trust Gates. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 40(12):3007–3021, 2018.

[18] J. Liu, G. Wang, L.-Y. Duan, K. Abdiyeva, and A. C. Kot. Skeleton-Based Human Action Recognition With Global Context-Aware Attention LSTM Networks. *IEEE Transactions on Image Processing*, 27(4):1586–1599, 2018.

[19] M. Liu, H. Liu, and C. Chen. Enhanced skeleton visualization for view invariant human action recognition. *Pattern Recognition*, 68:346–362, 2017.

[20] F. Ofli, R. Chaudhry, G. Kurillo, R. Vidal, and R. Bajcsy. Sequence of the most informative joints (smij): A new representation for human skeletal action recognition. *Journal of Visual Communication and Image Representation*, 25(1):24–38, 2014.

[21] A. Shahroudy, J. Liu, T.-t. Ng, and G. Wang. NTU RGB+D: A Large Scale Dataset for 3D Human Activity Analysis. In *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 1010–1019, 2016.

[22] L. Shi, Y. Zhang, J. Cheng, and H. Lu. Skeleton-based action recognition with directed graph neural networks. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 7912–7921, 2019.

[23] L. Shi, Y. Zhang, J. Cheng, and H. Lu. Two-stream adaptive graph convolutional networks for skeleton-based action recognition. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 12026–12035, 2019.

[24] C. Si, W. Chen, W. Wang, L. Wang, and T. Tan. An attention enhanced graph convolutional lstm network for skeleton-based action recognition. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 1227–1236, 2019.

[25] K. Simonyan and A. Zisserman. Two-stream convolutional networks for action recognition in videos. In *Advances in neural information processing systems*, pages 568–576, 2014.

[26] W. Sultani, C. Chen, and M. Shah. Real-World Anomaly Detection in Surveillance Videos. In *2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 6479–6488, 2018.

[27] D. Tran, L. Bourdev, R. Fergus, L. Torresani, and M. Paluri. Learning spatiotemporal features with 3d convolutional networks. In *Proceedings of the IEEE international conference on computer vision*, pages 4489–4497, 2015.

[28] R. Vemulapalli, F. Arrate, and R. Chellappa. Human Action Recognition by Representing 3D Skeletons as Points in a Lie Group. In *2014 IEEE Conference on Computer Vision and Pattern Recognition*, pages 588–595, 2014.

[29] J. Wang, X. Nie, Y. Xia, Y. Wu, and S.-C. Zhu. Cross-view action modeling, learning and recognition. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 2649–2656, 2014.

[30] J. Wang, Y. Song, T. Leung, C. Rosenberg, J. Wang, J. Philbin, B. Chen, and Y. Wu. Learning Fine-Grained Image Similarity with Deep Ranking. In *2014 IEEE Conference on Computer Vision and Pattern Recognition*, pages 1386–1393, 2014.

[31] L. Wang, Y. Xiong, Z. Wang, Y. Qiao, D. Lin, X. Tang, and L. Van Gool. Temporal segment networks: Towards good practices for deep action recognition. In *European conference on computer vision*, pages 20–36. Springer, 2016.

[32] P. Wang, W. Li, P. Ogunbona, J. Wan, and S. Escalera. Rgb-d-based human motion recognition with deep learning: A survey. *Computer Vision and Image Understanding*, 171:118–139, 2018.

[33] J. Weng, C. Weng, and J. Yuan. Spatio-Temporal Naive-Bayes Nearest-Neighbor (ST-NBNN) for Skeleton-Based Action Recognition. In *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 445–454, 2017.

[34] L. Xia, C.-C. Chen, and J. K. Aggarwal. View invariant human action recognition using histograms of 3d joints. In *2012 IEEE Computer Society Conference on Computer Vision and Pattern Recognition Workshops*, pages 20–27, 2012.

[35] S. Yan, Y. Xiong, and D. Lin. Spatial Temporal Graph Convolutional Networks for Skeleton-Based Action Recognition. *Thirty-Second AAAI Conference on Artificial Intelligence*, 2018.