View Adaptive Neural Networks for High Performance Skeleton-based Human Action Recognition

Hüseyin Temiz

17.02.2021

1 Introduction

Human action recognition is one of the most important problems in computer vision research since action recognition has many applications in daily life. Regarding the type of input given to the algorithm, action recognition has many sub-versions. In this report, we study the method [17] proposed by Zhang *et al.* in TPAMI 2019. The method takes 3D skeleton sequences as an input to predict the action class performed. RGB-image based action recognition methods utilize the image sequences to learn action classes. Skeleton-based methods process skeleton representations extracted from image sequences or sensor data.

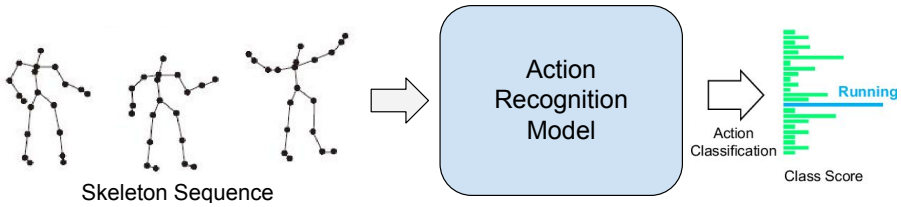

Figure 1: General pipeline of skeleton-based action recognition method.

Regardless of input type, humans in an action can be seen from different viewpoints. Therefore, action recognition methods should be robust to viewpoint variation. In the case of view bias in a method, there will be misclassification when the viewpoint is different from the model trained on. In [17], Zhang *et al.* proposes a view adaptive method to handle view variance in skeleton sequence for higher recognition performance.

In [17], the authors propose two different view-adaptation module architectures: VA-RNN and VA-CNN. Since VA-CNN performs significantly better than VA-RNN, we choose the VA-CNN as an inspiration in our project. As a novelty, we propose an alternative design for view adaptation module by utilizing equivariant steerable CNNs [15]. In experiments, we also make use of emsembling different VA modules to get further classification performance.

2 Related Works

By nature, human action contents can be collected from varying camera viewpoints. Arbitrary camera views make the learning algorithms difficult to generalize action dynamics. In the context of skeleton-based action recognition, there are a few studies proposing different methods to be view-invariant. Pre-processing the skeletons frame by frame is the common strategy [2, 6, 9]. In this strategy, skeletons in each frame are moved to the body center with upper body alignment. But, this strategy causes partial loss in relative motion between consequent frames, which can be highly informative for some action classes.

Unlike frame by frame pre-processing strategy, sequence level pre-processing applies the same transformation to all frames with the same parameters derived from the initial frame [14]. So, the sequence-level

strategy can preserve the motion information. However, sequence-level pre-processing can cause undesirable transformation for some action classes containing body bending.

In [17], Zhang *et al.* propose a content-dependent-view adaptation module, which has an automated inferring mechanism providing a suitable viewpoint. This automatic mechanism can handle the static view transformation methods (frame-based and sequence-based).

Convolutional neural networks (CNN) are known as powerful in classification. In the action recognition literature, there are some studies [4,20,23], converting a skeleton sequence to 2D images and then utilizing CNNs to classify the action. [4,23] uses three channels in the image to encode x,y,z axes in 3D coordinates. Also, each frame is encoded in a row or a column. Another common trick in this approach is the normalization of coordinate values to the range 0-255.

In the literature, CNN-based action recognition methods do not focus on handling the view variation. So, [17] propose a CNN-based view adaptation module to the CNN-based classifier. In our study, we propose a different view adaptation module architecture. In our view adaptation module, we utilize equivariant-steerable CNNs (ES-CNN) [15], which are designed to be more robust to transformations in the image domain such as translations, rotations, and reflections. Even though conventional CNNs have also some equivariance property, their generalization capability is not strong as ES-CNNs [15].

3 Methods

In the paper [17], Zhang *et al.* proposes two different adaptation modules: VA-CNN and VA-RNN. In the benchmarks provided in [17], the CNN-based view adaptation module is better than the RNN-based view adaptive module. In the paper, fusion-based architecture VA-fusion, which combines VA-CNN and VA-RNN, is proposed. In the benchmarks, VA-fusion is slightly better than VA-CNN.

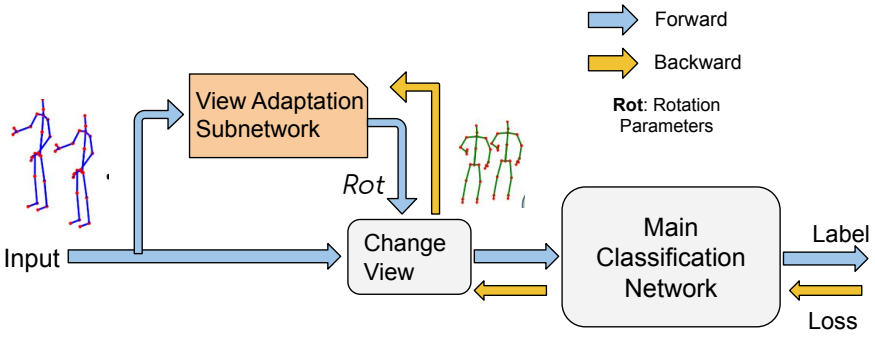

Figure 2: Pipeline

In further, we propose a modification on VA-CNN to get a better action recognition performance.

3.1 Image Representation

As in [17], a skeleton sequence is transformed into an RGB image. Each column represents different frames while different joints are represented in rows. Since skeletons are in 3D, 3D coordinates are encoded in every three channels of an image.

In the NTU dataset, each frame can have two skeletons. Each skeleton has 25 different 3D joints. Therefore, each frame is represented in a 150-dimensional vector. Samples from action sequences can have a different number of frames. As a detail from the implementation, the maximum number of frames is 300. Sequences with more than 300 are cropped and less than 300 are padded with zeroes. In the pre-processed files, the actions have fixed representations: 300x150.

$$\mathbf{u_{t,j}} = \text{floor}\left(255 \times \frac{\mathbf{v_{t,j}} - \mathbf{c_{min}}}{c_{\max} - c_{\min}}\right)$$

Before transforming fixed-sized action sequences into RGB images, unfilled and padded parts, such as an unused second skeleton or padded frames, are cropped. So, the initial size (300x150) of action sequences can decrease. After getting rid of zero parts, values $\mathbf{v_{t,j}}$ are normalized, then scaled to 0-255. Our image representation of action sequences is 224x224x3.

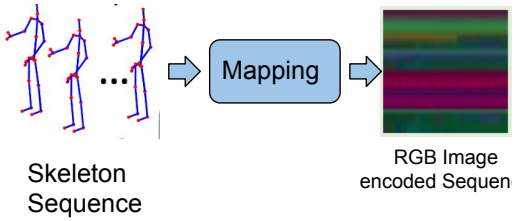

Figure 3: Mapping

3.2 View Transformation

The skeleton representation of the j^{th} joint in the t^{th} frame is denoted as $\mathbf{v_{t,j}}$. All the joints in the t^{th} frame is denoted as $\mathrm{V}_t = \{\mathbf{v}_{t,1}, \cdots, \mathbf{v}_{t,J}\}$. Then, we move the observation view point to a suitable global view point with translation $\mathbf{d}_t$ and rotation parameters $\alpha_t, \beta_t, \gamma_t$. So, under observation view point, the representation of j^{th} joint in the t^{th} frame is:

$$\mathbf{v}'_{t,j} = \left[x'_{t,j}, y'_{t,j}, z'_{t,j}\right]^{\mathrm{T}} = \mathbf{R}_t \left(\mathbf{v}_{t,j} - \mathbf{d}_t\right)$$

$\mathbf{R}_t$ is denoted as $\mathbf{R}_t = \mathbf{R}^x_{t,\alpha}\mathbf{R}^y_{t,\beta}\mathbf{R}^z_{t,\gamma}$. In the t^{th} frame, all joints in the skeleton share the view point transformation parameters since changing view point is a rigid motion. The view points for different frames can be different. Finally, view adaptive module tries to determine transformation parameters for the t^{th} frame: $\mathcal{T}_t = \{\alpha_t, \beta_t, \gamma_t, \mathbf{d}_t\}$, which corresponds to $6 \times T$.

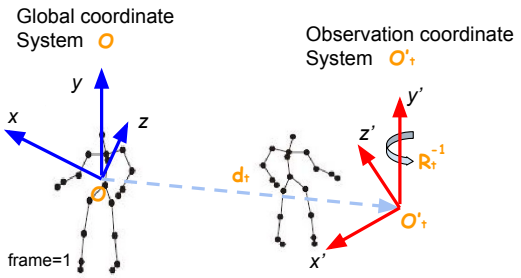

Figure 4: View transformation

3.3 View Adaptation Module with CNN (VA-CNN)

In [17], Zhang *et al.* propose the view adaptation module that finds the transformation parameters for a better viewpoint in terms of classification networks. By view transformation, the classification network does not have to focus on the view variances in the same action classes. So, the classification network can focus to learn the differences between action classes.

3.4 View Adaptation Module with Equivariant Steerable CNN (VA-ES-CNN)

In this study, we investigate the equivariance properties of ES-CNN over viewpoint transformation. Like the VA-CNN module, our adaptation module, VA-ES-CNN, takes the image encoded skeleton sequence as input and automatically determines the view transformation parameters for better action classification. Therefore, our VA-ES-CNN framework is the same as VA-CNN as depicted in Figure 5.

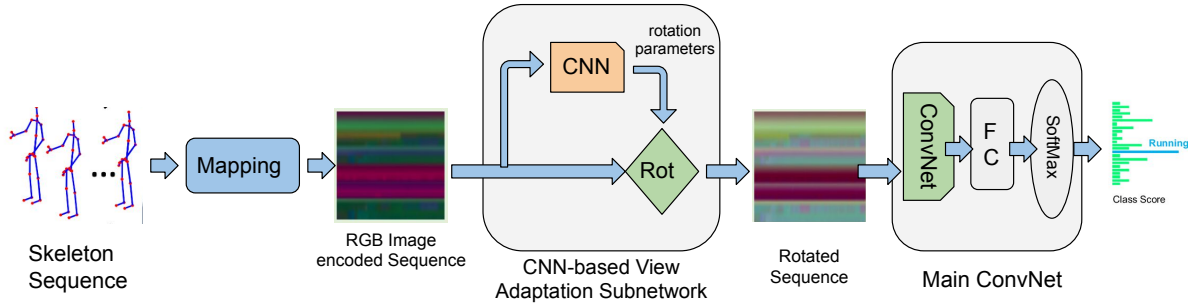

Figure 5: Pipeline with VA-CNN module.

To implement the ES-CNN based adaptation module, we utilize the *e2cnn* library published in [4]. The *e2cnn* library is compatible with PyTorch v1.1 and later with GPU acceleration.

3.5 The Fusion of VA-CNN and VA-ES-CNN

In the ML literature, to get further performance, it is popular to ensemble models trained on the same problem, but having different learning capabilities. Therefore, after training and evaluating different adaptation module architecture, we also investigate the classification performance of the ensemble model.

4 Experiments

4.1 NTU RGB+D Dataset

In this project, our experiments are only in the NTU RGB+D [12] dataset since action recognition datasets are big and model training can take a long time. NTU RGB+D dataset contains 56880 video samples. There are 60 different action classes. There are 40 different human subjects. Each human subject is represented by 25 3D joints. There are two standard benchmarks: Cross-Subject (CS), where the 40 subjects are divided into training and testing groups, and Cross-View (CV), where cameras 2 and 3 are used for training, and camera 1 for testing.

4.2 Experiment Setup and Training Details

The official implementation of the paper [17] is given in the url [5]. In the official code, PyTorch 1.0 and Python 2.7 are used. In our experiments, we make the codes and scripts compatible with PyTorch 1.7.1 and Python 3.7. We will publish the latest codes in our repository. We have an experiment environment with NVidia GTX1070ti (8GB) GPU and Ubuntu 20.04 OS.

For VA-CNN and VA-ES-CNN based models, ResNet-50 with pre-trained parameters on ImageNet [1] is utilized for classification. The batch size is 32 for all training scenarios. Adam optimizer [8] is chosen to train model parameters. Skeletons in different sequences can have varying tensor sizes and resize in 224x224x3. For better generalization, we also apply augmentation to skeletons, applying random small rotations around X, Y, and Z axes. The maximum training epoch count is 25 for all cases. We use early stopping to prevent overfitting.

4.3 Experiments

In our experiments, we use only NTU dataset benchmarks to evaluate the performance of different methods. As mentioned before, the NTU dataset has two different well-defined benchmarks: cross-subject (CS) and cross-view (CV). In action recognition research, the NTU dataset and its benchmarks are very common.

In Table 1, we present the performances of adding different adaptive modules. All the results are taken from our experiments. If we do not use any view-adaptation (No-VA CNN), the CNN-based action

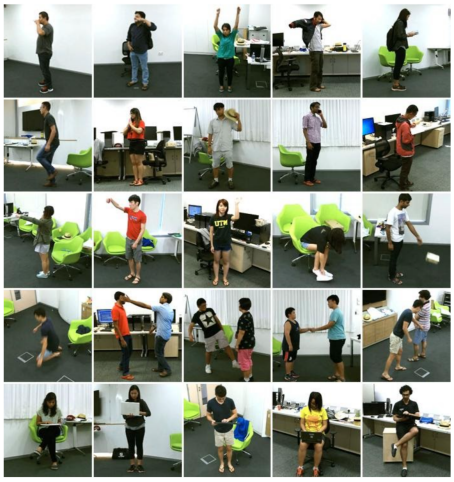

Figure 6: Sample frames from NTU RGB+D dataset [12].

classifier can present 87.50% accuracy for CS, 93.09% accuracy for CV benchmarks. This is the baseline performance of the idea which encodes sequences as RGB images and uses a pre-trained network as an action classifier. This simple and powerful method can perform better than some complex methods (e.g. LSTM or RNN based).

Adding VA-CNN or VA-ES-CNN modules increases the classification performance slightly. Applying augmentation to skeletons can also contribute to the classification performance especially for VA-CNN. In our experiments, we observe that the fusion of different VA modules also improves the accuracy.

Table 1: Experiments on the effectiveness of different view-adaptive modules. (accuracy(%))

MODEL	CS	CV
No-VA CNN	87.50	93.09
VA-CNN	88.02	93.67
VA-ES-CNN	87.91	93.09
Fusion	**89.16**	**94.68**
VA-CNN (+Aug)	88.74	93.85
VA-ES-CNN (+Aug)	87.72	93.47
Fusion(+Aug)	**89.32**	**94.66**
Fusion-X (VA-ES-CNN)&(VA-CNN +Aug)	**89.42**	**94.79**

In [17], it is shown that VA-CNN with augmentation performs 88.7% accuracy for CS, 94.3% accuracy for CV benchmark. In the CS benchmark, our trained VA-CNN based model shows the same performance as the original results. But, in the CV benchmark, there is a 0.45% difference between official results and our experiments.

The best performance in two different benchmarks is shown by the fusion model that combines the predictions of VA-CNN(+AUG) and VA-ES-CNN without augmentation. This specific model is named Fusion-X.

In Table 2, we compare the performance of different skeleton-based action recognition methods on the benchmarks of the NTU dataset. The performance of our VA-CNN based models is highly competitive in the literature among different architectures. The best configuration among our models, Fusion-X, is better than many methods. But, MS-AAGCN+TEM [11], published in 2020, is the current state-of-the-art

Table 2: Accuracy (%) on the NTU dataset.

MODEL	CS	CV
Skeleton Quads [3]	38.60	41.40
ST-LSTM+Trust Gate [10]	69.20	77.70
GCA-LSTM [13]	73.40	82.2
Clips+CNN+MTLN [7]	79.60	84.80
ST-GCN [16]	81.50	88.30
MS-AAGCN+TEM [11]	91.00	96.50
VA-CNN (+Aug) [17]	88.70	94.30
VA-ES-CNN	87.91	93.09
VA-CNN (+Aug) (ours)	88.74	93.85
VA-ES-CNN (+Aug)	87.72	93.47
Fusion-X	**89.42**	**94.79**

results among the methods as far as we can analyze.

5 Conclusion

The most powerful part of the model's success is mapping the skeleton sequences to images. We observe that CNNs can learn the dynamics of action well after sequences are encoded in images. View adaptation is a contributing schema even after the benchmark results are over 90% accuracy. As in many ML problems, the fusion of different models trained on the same problem still improves the performance. Even though Equivariant Steerable CNNs are different from traditional CNNs in terms of architecture, our proposed ES-CNN based adaptation module can perform as well as the traditional CNN-based VA module. The fusion of these two CNN-based VA modules can improve the classification performance further. As future work, well-designed and carefully trained ES-CNNs may perform better than traditional CNNs in view adaptation modules. ES-CNNs may also be utilized in the main classification network instead of the pre-trained ResNet network. Our code is available for research purposes at `https://github.com/hsyntemiz/VA-ES-CNNs-for-Skeleton-based-Human-Action-Recognition`.

References

[1] Jia Deng et al. "Imagenet: A large-scale hierarchical image database". In: *2009 IEEE conference on computer vision and pattern recognition*. Ieee. 2009, pp. 248–255.

[2] Yong Du, Wei Wang, and Liang Wang. "Hierarchical recurrent neural network for skeleton based action recognition". In: *Proceedings of the IEEE conference on computer vision and pattern recognition*. 2015, pp. 1110–1118.

[3] Georgios Evangelidis, Gurkirt Singh, and Radu Horaud. "Skeletal quads: Human action recognition using joint quadruples". In: *2014 22nd International Conference on Pattern Recognition*. IEEE. 2014, pp. 4513–4518.

[4] *Github: General E(2)-Equivariant Steerable CNNs*. `https://github.com/QUVA-Lab/e2cnn`. Accessed: 2021-02-14.

[5] *Github: View-Adaptive-Neural-Networks-for-Skeleton-based-Human-Action-Recognition*. `https://github.com/microsoft/View-Adaptive-Neural-Networks-for-Skeleton-based-Human-Action-Recognition`. Accessed: 2021-02-12.

[6] Min Jiang et al. "Informative joints based human action recognition using skeleton contexts". In: *Signal Processing: Image Communication* 33 (2015), pp. 29–40.

[7] Qiuhong Ke et al. "A new representation of skeleton sequences for 3d action recognition". In: *Proceedings of the IEEE conference on computer vision and pattern recognition*. 2017, pp. 3288–3297.

[8] Diederik P Kingma and Jimmy Ba. "Adam: A method for stochastic optimization". In: *arXiv preprint arXiv:1412.6980* (2014).

[9] Wenbo Li et al. "Adaptive RNN tree for large-scale human action recognition". In: *Proceedings of the IEEE International Conference on Computer Vision*. 2017, pp. 1444–1452.

[10] Jun Liu et al. "Spatio-temporal lstm with trust gates for 3d human action recognition". In: *European conference on computer vision*. Springer. 2016, pp. 816–833.

[11] Yuya Obinata and Takuma Yamamoto. "Temporal Extension Module for Skeleton-Based Action Recognition". In: *arXiv preprint arXiv:2003.08951* (2020).

[12] Amir Shahroudy et al. *NTU RGB+D: A Large Scale Dataset for 3D Human Activity Analysis*. 2016. arXiv: `1604.02808 [cs.CV]`.

[13] Sijie Song et al. "An end-to-end spatio-temporal attention model for human action recognition from skeleton data". In: *Proceedings of the AAAI conference on artificial intelligence*. Vol. 31. 1. 2017.

[14] Jiang Wang et al. "Learning actionlet ensemble for 3D human action recognition". In: *IEEE transactions on pattern analysis and machine intelligence* 36.5 (2013), pp. 914–927.

[15] Maurice Weiler and Gabriele Cesa. *General $E(2)$-Equivariant Steerable CNNs*. 2019. arXiv: `1911.08251 [cs.CV]`.

[16] Sijie Yan, Yuanjun Xiong, and Dahua Lin. "Spatial temporal graph convolutional networks for skeleton-based action recognition". In: *Proceedings of the AAAI conference on artificial intelligence*. Vol. 32. 1. 2018.

[17] Pengfei Zhang et al. "View adaptive neural networks for high performance skeleton-based human action recognition". In: *IEEE Transactions on Pattern Analysis and Machine Intelligence* (2019).