SVIM: A Skeleton-Based View-Invariant Method for Online Gesture Recognition

Yang Zhao[1], Lanfang Dong[1](✉), Guoxin Li[1], Yingchao Tang[1], Yuhang Zhang[1],
Meng Mao[2], Guoming Li[2], and Linxiang Tan[2]

[1] University of Science and Technology of China, Hefei 230026, China
{yvngzhvo,tangyc314,yhzhang}@mail.ustc.edu.cn, lfdong@ustc.edu.cn,
guoxinli@mail.ustc.edu.cn

[2] AI Lab, China Merchants Bank, Shenzhen 518040, China
{melvinmaonn,tanlinxiang252}@cmbchina.com, lkm@cmbchina.com

Abstract. Online gesture recognition is a challenging task in practical application scenarios since the gesture is not always directly in front of the camera. In order to solve the challenges caused by multiple viewpoints of skeleton data, in this paper, we proposed a novel view-invariant method for online skeleton gesture recognition. The whole skeleton sequence data as a point set in our method and a PCA-based view-invariant data preprocessing algorithm is proposed and applied in this paper. We can transform similar skeleton data to relatively stable viewpoints by applying the PCA algorithm according to the similarity of distribution features of the point set, which can ensures the viewpoint stability of our gesture recognition model. We conduct extensive experiments on the NTU RGB+D and Northwestern-UCLA benchmark datasets which contain multiple viewpoints and the results have demonstrated the effectiveness of the method proposed in this paper.

Keywords: Online gesture recognition · View-invariant · PCA

1 Introduction

The view-invariant gesture recognition algorithm has a wide range of applications. When applying the gesture recognition model to real scenarios, the person doing the gesture action often does not happen to be standing directly in front of the camera. For example, in a robot scenario, the robot may need to respond to a user's waving gesture, and the user doing the waving gesture may not necessarily be directly in front of the robot, although he or she is within the robot's view. Another example is that when a self-driving car needs to detect the traffic police action, the location of the traffic police may not be right in front of the car's camera either.

Due to the change of viewpoint, the estimated skeleton coordinates from different viewpoints sometimes differ greatly, which seriously affects the recognition performance of the action recognition model. In addition, the movements

X. Yang et al. (Eds.): ADMA 2023, LNAI 14179, pp. 241–255, 2023.
https://doi.org/10.1007/978-3-031-46674-8_17

from different viewpoints are affected by self-occlusion, which causes the estimated skeleton to be disturbed by different degrees of noise. Therefore, gesture recognition with a constant viewpoint is a challenging problem.

The problem of view-invariance can be hardly solved by data enhancement due to the diversity of viewpoints. Some researchers try to weakens the effect of viewpoint variation on action representation by designing view-invariant features [8,15,25], but this approach can only handle small magnitude viewpoint changes. Other researcher split the skeleton into multiple parts and deal with viewpoint changes by modeling the geometric relationships and cannot really address the viewpoint change problem [23,26]. There is also literature on building new coordinate systems from the first frame or the skeleton of the previous frames [14,24], however this approach is highly sensitive to the onset motion of the gesture. Deep learning has achieved great success in many fields in recent years, and more and more researchers seek deep learning solutions. One solution is to use feature migration to seek a common feature space from data with different perspective [7], and other solution is to learn perspective-invariant representation from data [18]. However, the biggest problem with the learning-based approach is that the dataset used to train the model contains only a limited number of perspectives.

Unlike these approaches mentioned above, the basic idea of our method is that, the same action, although it may have various intra-class differences, is still composed of many similar motion states from a global perspective. If the whole motion sequence is treated as a point set, then these point sets tend to have similar shape characteristics. The difference in veiwpoint is reflected in the point set as a different in rotation direction. Therefore, if a way can be found to rotate this point set to a stable orientation, then this orientation can be considered as the standard viewpoint of this skeleton sequence. In this way the gesture recognition model can obtain an input source with a stable viewpoint. Moreover, this method only needs to be added to the preprocessing process of the data and can be applied directly to almost any existing gesture recognition method.

Based on the above ideas, we propose a view-invariant algorithm based on PCA. Principal Components Analysis(PCA) can compute a set of basis vectors from the point set that reflect the characteristics of the data distribution. In this paper, this set of basis vectors is used to transform the point set to a new basis coordinate space. Since this set of basis vectors is determined by the distribution characteristics of the data, data with similar distribution characteristics also have similar distribution characteristics in the new base coordinate space.

Our contribution is as follows:

1. For the multi-view problem of skeleton data, we propose a novel solution by applying the PCA algorithm to rotate similar skeleton sequences to relatively stable viewpoints based on the similarity of point set distribution, thus achieving view-invariant gesture recognition based on skeleton;
2. we demonstrate the effectiveness of the algorithm in several experiments on the multi-view datasets NTU RGB-D and Northwestern-UCLA, which contain multiple viewpoints for action recognition.

2 Related Work

Since the coordinates of the skeleton nodes obtained by the skeleton estimation algorithm vary greatly from viewpoint to viewpoint, and the differences in motion occlusion also cause the estimated skeleton to be disturbed by different degrees of noise. The viewpoint-invariant gesture recognition algorithm investigates the method that gestures taken from different angles from the training data can also be classified accurately.

Xia et al. [24] proposed method is to establish a spherical coordinate system in a specific direction on the skeleton. Specifically, they chose the hip center joint of the human skeleton as the midpoint, defined the horizontal reference vector as a vector projection from the left hip center joint point to the right hip center joint point onto the horizontal plane (parallel to the ground), and the zenith reference vector was defined as a vector perpendicular to the ground plane. Then they discretized the 3-dimensional space into n small intervals and discretized the joint point coordinates into these small intervals. Finally, they do probabilistic voting on these discretized coordinates to increase the stability of the features, use Linear Discriminant Analysis (LDA) to extract more discriminative features, k-Means clustering into dictionaries, and finally use Discrete Hidden Markov Model (DHMM) to do the classification. Zhang Yi et al [30] proposed to map the gesture trajectory features to be represented as global invariants based on the Centroid Distance Function (CDF). The center-of-mass distance function is the distance from each point on the trajectory point to the centroid, and the authors in the paper take the center of the hand as the centroid. Pei Xiaomin et al. [16] added the angle between the trajectory point and the centroid on top of this. Ghorbel et al. [6] proposed to independently fuse two multi-view invariant methods: the Ghorbel et al. [5] and Vemulapalli et al. [21]'s approach to perspective invariant classification. Ji et al [7] proposed using an attention mechanism to focus on the most critical joint points in the skeleton of multiple views and the relationship between them. Li et al. [14] create a new coordinate system from the first few frames of the camera view of the skeleton, and then convert the skeleton sequence to an orthographic view on this coordinate system, so that the skeleton has a stable view.

3 Method

It is known that the PCA algorithm can calculate from the data a set of basis vectors that reflect the characteristics of the data distribution, and the set of basis vectors is the eigenvectors of the covariance matrix of the data.

3.1 Calculate the Eigenmatrix

For a sequence of skeletons it can be considered as a point set $P = [p_1, p_2, \ldots, p_N] \in \mathbb{R}^{3\times N}$. Firstly, we center the point set P, i.e., for any point $p_i \in P$

$$\hat{p}_i = p_i - \frac{1}{n}\sum_{i=1}^{n} p_i \tag{1}$$

thus forming the new point set $\bar{P} = [\hat{p}_1, \hat{p}_2, \ldots, \hat{p}_n] \in \mathbb{R}^{3\times n}$. Then we eigndecompose the covariance matrix of the point set $\bar{P}$, i.e.

$$\bar{P}\bar{P}^T = R\Lambda R^T \tag{2}$$

Here, the matrix $R = [r_1, r_2, r_3] \in \mathbb{R}^{3\times 3}$ is the eigenmatrix and its three eigenvectors(also called the principal axes), and the diagonal matrix $\Lambda = \mathrm{diag}(\lambda_1, \lambda_2, \lambda_3)$ are three eigenvalues (also called the principal values) corresponding to the eigenvectors.

The point set P_{can} with rotational invariance can be obtained by aligning the principal axes with the world coordinate, that is, by computing $P_{\mathrm{can}} = R^T P$.

Theorem 1. *The point set P_{can} is rotation invariant.*

Proof. Assume that $Q \in \mathrm{SO}(3)$ is an arbitrary rotation matrix, then QP is the set of points after rotation of the point set P. The centerized point set $Q\bar{P}$ can be obtained from Eq. (1), then the covariance matrix of this point set can be convert to

$$\begin{aligned} Q\bar{P}(Q\bar{P})^T &= Q\bar{P}\bar{P}^T Q^T \\ &= Q(R\Lambda R^T)Q^T \\ &= (QR)\Lambda(QR)^T \end{aligned} \tag{3}$$

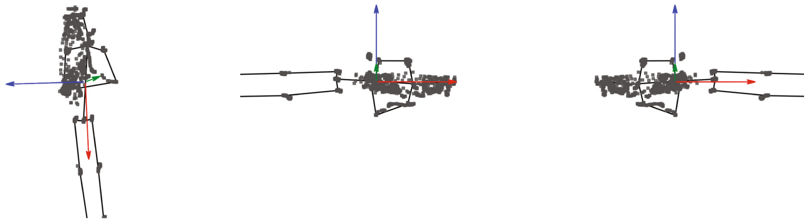
At this point, QR becomes the new principal axes. Therefore, after rotating the point set QP and aligning it with new main axis

$$\begin{aligned} (QP)_{\mathrm{can}} &= (QR)^T QP \\ &= R^T Q^T QP \\ &= R^T P = P_{\mathrm{can}} \end{aligned} \tag{4}$$

This means that rotating the matrix Q has no effect.

3.2 Ambiguity of the Feature Matrix

However, if we use the eigenmatrix as a transformation matrix directly, we will suffer from two kinds of ambiguities: sign ambiguity and order ambiguity. The sign ambiguity refers to the fact that, for a given eigenvector r_i, it can take either $+r_i$ or $-r_i$ under the condition that the eigendecompose is satisfied. By assigning the positive or negative sign to eigenvectors, then the transformed skeleton sequence will yield 8 possible perspectives. The order ambiguity refers to the problem of the order of the eigenvectors, which is not specified to be arranged in the order of the eigenvalues. In this case, 6 possible views will be generated when computing the standard view. That is, when changing the positive and negative signs of the eigenvectors or the order of the eigenvectors, it still produces



(a) Original

(b) Rotation

(c) Mirroring

Fig. 1. Rotation and mirroring comparison of the eigenmatrix

a point set with many different rotations. In fact, a total of 48 views are possible after the eigenmatrix transformation.

As Li et al [12] pointed out, among the 8 sign ambiguities, some cases are in fact not rational transformations. Specifically, if the combination of some eigenvector $R = [r_1, r_2, r_3]$ and its determinant is 1, then only four of the eight ambiguities with a determinant of 1 are true rotation, and the other determinant values of -1 are a combination of rotation and mirror transformation(see Figure 1). Table 1 list all of sign ambiguities.

Table 1. The sign ambiguities of eigenmatrix

Eigenmatrix	Determinant	Rotation
$[+r_1, +r_2, +r_3]$	$+1$	Yes
$[-r_1, +r_2, +r_3]$	-1	No
$[+r_1, -r_2, +r_3]$	-1	No
$[+r_1, +r_2, -r_3]$	-1	No
$[-r_1, -r_2, +r_3]$	$+1$	Yes
$[+r_1, -r_2, -r_3]$	$+1$	Yes
$[-r_1, +r_2, -r_3]$	$+1$	Yes
$[-r_1, -r_2, -r_3]$	-1	No

If mirroring is included, then for some gestures in opposite directions, such as waving to the left and waving to the right, these actions will not be distinguishable. After removing the sign ambiguity with the mirror transformation, the PCA-based perspective algorithm produces

$$4(\text{sign ambiguity}) \times 6(\text{order ambiguity}) = 24. \tag{5}$$

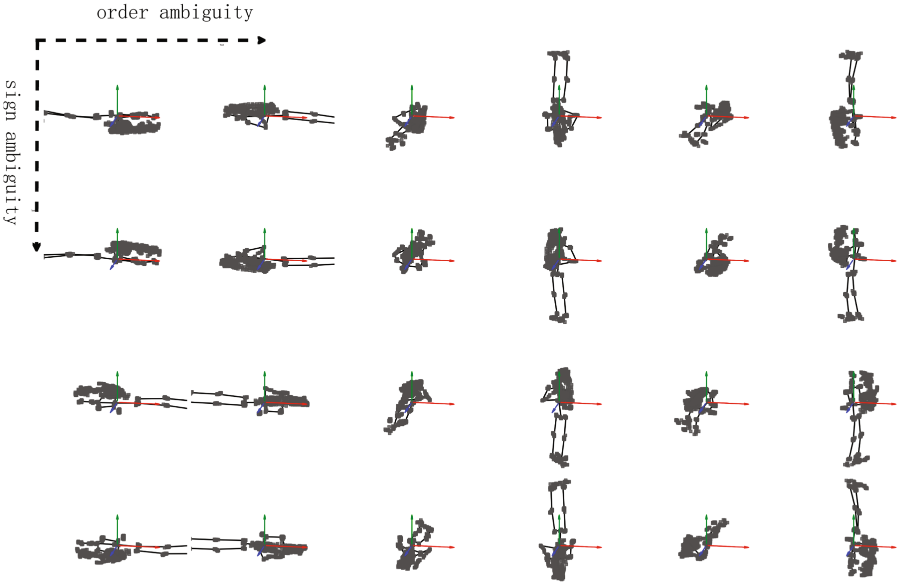


Fig. 2. The 24 ambiguities of the eigenmatrix

The ambiguities are listed in Fig. 2, which contains these 24 transformations.

3.3 Flow of Algorithm

To solve the ambiguity problem arising from PCA transformation, the solution proposed in this paper is to add as many ambiguous cases as possible to the training data during training, while only the order of the feature vectors is processed or not at all during testing. Although such an approach is similar to data augmentation of the viewpoints of the skeleton sequence, conventional multi-viewpoint data augmentation methods often enable the model to learn only a limited number of viewpoints, and in fact, it is impossible to learn all of them. In contrast, the preprocessing algorithm proposed in this paper enables the model to learn only a limited number of cases to be able to cover all viewpoint cases. The specific process is shown in Algorithms 1 and 2.

```
Data: Train model $M$, train set $D_{\text{train}} \in \mathbb{R}^{B\times 3\times T\times N}$
Result: Models with perspective invariance $M$
for $X \in D_{train}$ do
    $P \leftarrow$ Reshape$(X, (3, -1))$;                    /* $P \in \mathbb{R}^{3\times(T\times N)}$ */
    $\bar{P} \leftarrow P -$ Mean$(P)$;
    $\hat{P} \leftarrow \bar{P}\bar{P}^T$;
    $L, Q \leftarrow$ Eign$(\hat{P})$;    /* $L$ is eigenvalues, $Q$ is eigenmatrix */
    /* The function PcaAmbiguity returns the disambiguation of
       the feature matrix on demand                                 */
    $Rs \leftarrow$ PcaAmbiguity$(L, Q)$;
    for $R \in Rs$ do
        $P' \leftarrow R^T P$;
        $X' \leftarrow$ Reshape$(P', (3, T, N))$;
        Train$(M, X')$;
    end
end
```

Algorithm 1: Training process of PCA-based view-invariant algorithm

```
Data: Test model $M$, test set $D_{\text{test}} \in \mathbb{R}^{B\times 3\times T\times N}$
for $X \in D_{test}$ do
    $P \leftarrow$ Reshape$(X, (-1, 3))$;
    $\bar{P} \leftarrow P -$ Mean$(P)$;
    $\hat{P} \leftarrow \bar{P}\bar{P}^T$;
    $L, Q \leftarrow$ Eign$(\hat{P})$;    /* Optional: Sort the vectors in Q by the
    size of L */
    $P' \leftarrow Q^T P$;
    $X' \leftarrow$ Reshape$(P', (3, T, N))$;
    Test$(M, X')$;
end
```

Algorithm 2: Testing procedure of PCA-based view-invariant algorithm

3.4 Experiment and Analysis

In order to verify the effectiveness of the PCA-base view-invariant algorithm proposed in this paper, this section applies the algorithm to two datasets with multiple views, NTU RGB+D and Northwestern-UCLA for experiments. The dataset Northwestern-UCLA is a relatively small action recognition dataset, while the dataset NTU RGB+D is a large action recognition dataset. This section first introduces these two multiview datasets, then presents implementation details used in this section, and finally, the experimental results and analysis are presented.

3.5 Dataset

NTU RGB+D. NTU RBG+D [17] is a large dataset designed from human action recognition, containing a total of 56880 3D skeleton sequences. It contains 60 action categories, including "drinking", "snacking", "brushing teeth", "combing hair ", "picking things up", and so on. The sample actions are performed by a total of 40 volunteers, and a maximum of 2 people in a sample is guaranteed. Each action sample was simultaneously captured by 3 different views of the Microsoft Kinect v2 camera. The different people and perspectives presented a significant challenge in discriminating between intra- and inter-class differences. NTU RGB+D is quite a challenging dataset considering the size of the dataset, the effect of similar actions and the noise in the dataset. To experiment the viewpoint invariant algorithm proposed in this paper, we use the recommended cross-view (X-View) benchmark test for this dataset: training data from camera views #2 and #3, and test data from camera view #1.

Northwestern-UCLA. The Northwestern-UCLA multi-view 3D event dataset [23] is a multi-view multimodal dataset containing RGB, depth, and human skeleton data captured by three Kinects simultaneously. The dataset contains a total of 1494 video clips, covering 10 action categories, each performed by 10 different volunteers. The full list of actions and the corresponding sample sizes are shown in Table 2. In this paper, we use the recommended evaluation of this dataset: training data from the first two cameras and test data from the latter one.

Table 2. Gesture information for Northwestern-UCLA

No.	Gesture	Amount
1	pick up with one hand	150
2	pick up with two	152
3	drop trash	141
4	walk around	173
5	sit down	148
6	stand up	149
7	donning	142
8	doffing	142
9	throw	145
10	carry	142

3.6 Implementation Details

The experimental gesture recognition model in this paper is DD-Net [27], and the Stochastic Gradient Descent(SGD) algorithm is chosen as the optimizer during training, the base learning rate is set to 0.1, each experiment is trained for 100 rounds, and the warm up strategy is used in the first 5 rounds followed by the ReduceLROnPlateau is used as the learning rate adjustment strategy to reduce the learning rate from 10^{-1} to 10^{-5}, and cross-entropy is used as the loss function. To test the effectiveness of the perspective invariant algorithm, both the training and test data are randomly rotated and scaled in this paper.

Because the dataset NTU RGB+D contains 2 skeletons per frame on some action categories, i.e., it contains actions done by two people together, and DD-Net does not consider this situation, a simple strategy is proposed here to fix this problem. Suppose the input data size is $B \times C \times T \times N \times M$, where B represents the size of a training batch, C represents the dimension of the skeleton sequence (usually 3), T represents the duration of the skeleton sequence, N represents the number of joint points of the skeleton, and M represents the number of people contained in the frame. Then, before feeding into the model, this paper first adjusts the input data to the shape of $(B \times M) \times C \times T \times N$, which is equivalent to expanding the batch size by a factor of M. Then the M in $B \times M$ is eliminated using the mean function when the data enters the final fully connected layer stage.

3.7 Experimental Results and Analysis

Different Data Preprocessing Methods. To test which of the two ambiguities, signed ambiguity or order ambiguity, is more important for the accuracy of the result, and to find a balance between improving the performance and reducing the data expansion(for a dataset with large sample size). To find a balance between improving performance and reducing data augmentation(for a dataset with a large sample size, excessive data augmentation can seriously increase training time), we first conduct several comparative experiments on Northwestern-UCLA.

In this paper, we use the DD-Net (filters=32) model to conduct comparative experiments. The way of experimentation is to design three levels of elimination schemes for sign ambiguity and order ambiguity, which also affect the expansion of the number of samples in the training set. For sign ambiguity, three expansion options are designed: randomly assigning positive and negative signs to the three eigenvectors, using all combinations of positive and negative signs for the rotation cases, and using all combinations of positive and negative signs. For order ambiguity, three expansion options are designed, namely, random order of eigenvectors, following the order of eigenvalues from smallest to largest, and all possible orders. In the test set, if the order ambiguity uses the ranking of eigenvalues, then the eigenvectors of the test set are treated similarly; otherwise, the default computed eigenvectors are used (as the default computed eigenvectors are in random order).

Specific expansion schemes and resulting expansion scale to the training dataset are listed below:

1. Direct use of feature matrix (no expansion)

2. Eigenvectors are sorted by eigenvalue only(no expansion)
3. Eigenvectors sorted by eigenvalue, positive and negative signs for all rotation cases(expanded by a factor of 4)
4. Eigenvectors sorted by eigenvalue, positive and negative signs of all eigenvectors(expanded by a factor of 8)
5. All sorting of feature vectors(expanded by a factor of 6)
6. All ordering of feature vectors, positive and negative signs of all rotation cases(expanded by a factor of 24)
7. All ordering of feature vectors, positive and negative signs of all feature vectors(expanded by a factor of 48)
8. Positive and negative signs for all rotation cases(expanded by a factor of 4)
9. Positive and negative signs of all eigenvectors(expanded by a factor of 8)
10. Control group, without any special treatment(no expansion)

Table 3. Results of different experiments on Northwestern-UCLA

No.	Sign Ambiguity			Order Ambiguity			Result/%
	Random	Rotation	All	Random	Eigenvalue	All	
1	✓			✓			91.8
2	✓				✓		91.6
3		✓			✓		94.2
4			✓		✓		92.9
5	✓					✓	92.9
6		✓				✓	92.9
7			✓			✓	94.4
8		✓		✓			94.2
9			✓	✓			93.1
10							89.7

Finally, the experimental results obtained are shown in Table 3. From the experimental results, the following conclusions can be drawn:

1. Overall, the results processed by the PCA-based view-invariant algorithm(Exp 1-9) are generally better than the results without any processing(Exp 10). Even the eigenmatrix generated directly using PCA (i.e., the order and sign of the eigenvectors are variable, Exp 1) also improves the accuracy much more than the control experiment(Exp 10). This comparison demonstrates the effectiveness of our method.
2. In terms of the importance of elimination order ambiguity(Exp 2 and 5) v.s. sign ambiguity(Exp 8 and 9), eliminating Sign ambiguity is more effective in improving the model's performance. Intuitively, order ambiguity changes the overall orientation of the point set, which has a greater impact on the overall

distribution of the point set than sign ambiguity, which is only rotation and mirroring around the axes, thus affecting the performance improvement of the classifier.

3. The results sorted by eigenvalue are essentially equivalent to those without any treatment(Exp 1 and Exp 2). This is because even for the same actions, the overall distribution of the point set is different due to intra-class differences in the actions, and the eigenmatrix sorted based on the magnitude of the eigenvalues does not allow them to have similar orientations (e.g, so the heads of the skeletons are all oriented on the z-axis)
4. The comparison between Exp 3 and Exp 4, as well as the comparison between Exp 8 and Exp 9, verified the analysis done in sect. 3.2 of this paper, where the mirror transformation of the skeleton leads to some direction-dependent actions that are indistinguishable (eg, stand up actions and sit down actions), i.e., the non-rotating eigenmatrix has a certain degree of negative impact on performance.
5. However, when eliminating the sign ambiguity along with the order ambiguity(Exp6 and Exp 7), the non-rotating eigenmatrix is much higher than the rotating eigenmatrix. Considering that Exp 6 is already a larger expansion(24 times), Exp 7 is twice as large, Exp 7 likely has a larger amount of training data resulting in a more generalized model. In fact, Exp 7 has both the most expanded data and the best results of all the experiments.

Table 4. Results of different experiments on NTU RGB-D

No.	Sign Ambiguity			Order Ambiguity			Result/%
	Random	Rotation	All	Random	Eigenvalue	All	
1	✓			✓			85.7
2	✓				✓		86.0
3		✓			✓		89.6
4			✓		✓		90.0
5	✓					✓	88.5
6		✓				✓	88.5
7			✓			✓	91.0
8		✓		✓			89.3
9			✓	✓			90.0
10							86.4

In order to test the generalizability of the above findings, the above experiments were redone on the NTU RGB-D, and the results are shown in Table 4. The main difference between this dataset and Northwestern-UCLA is in the sample data size, which is much higher than the latter. The performance impact of data expansion cannot be ignored when expanding exponentially on a dataset with an already large sample size base. In general, the larger the sample size of the dataset, the less likely the trained model is to be overfitted, resulting in better model performance. In addition, the intra-class differences of each action will be highlighted by the increased sample size.

For NTU RGB-D, the result of Exp 10 is slightly better than the result of directly using the PCA sign matrix(Exp 1) and eigenvalue ranking(Exp 2). We analyze that this is due to the intra-class variation in this dataset. The same class of actions has been transformed by the eigenmatrix due to the different distributions aggravating the differences between them, which leads to performance degradation.

The conclusion that the elimination of sign ambiguity is more important than the elimination of order ambiguity still holds for NTU RGB-D. However, the eigenmatrix of rotation in the sign ambiguity is not higher than the eigenmatrix of all symbols. We analyze that for this dataset, the data expansion have a more important impact on the performance improvement, e.g, the accuracy of experiments without data expansion is around 86% (Exp1, Exp 2, and Exp 10) the accuracy of experiment with 4 times expansion is around 89%(Exp 3 and Exp 8) accuracy of experiments with 8 times expansion is 90%(Exp 4 and Exp 9), while the accuracy of the experiment with a 48-fold expansion was 91%(Exp 7). There are some exceptions to the results for the 6-fold and 24-fold expansions, both of which have an accuracy of 88.5%, which we estimate to be due to random factors when using the model.

Combining the experimental result of both datasets, the implementation of the sign ambiguity data expansion is more helpful to improve the model performance, while the maximum expansion(48 times) gives the best results. If it is necessary to find a balance between training time, we recommend using the 4-fold expansion of Exp 3, i.e., sorting the eigenvectors by eigenvalues, with all rotated eigenvectors. In addition, although the increase in the amount of training data has an impact on the experimental results, the above comparison experiments can still fully demonstrate the effectiveness of our PCA-based view-invariant algorithm.

Table 5. Results of comparison with other methods on the dataset NTU RGB-D

Methods	X-View/%
Ind-RNN [13]	88.0
HCN [11]	91.1
ST-GCN [26]	88.3
AGC-LSTM [19]	95.0
DDGCN [9]	**97.1**
CA-GCN [29]	91.4
SGN [28]	94.5
Shift-GCN [3]	96.5
CTR-GCN [1]	96.8
DD-Net(filters=64)	89.6
DD-Net(filters=64, Exp 7)	91.2

Table 6. Results of comparison with other methods on the dataset Northwestern-UCLA

Methods	X-View/%
Actionlet ensemble [22]	76.0
Lie Group [20]	74.2
HBRNN-L [4]	78.5
Ensemble TS-LSTM [10]	89.2
AGC-LSTM [19]	93.3
Shift-GCN [3]	94.6
DC-GCN+ADG [2]	95.3
CTR-GCN [1]	**96.5**
DD-Net(filters=64)	89.7
DD-Net(filters=64, Exp 7)	94.4

Comparison with Other Methods. Tables 5 and 6 show the results of this paper's gesture recognition method before and after using PCA-based view-invariant algorithm, compared with other methods. On the large skeleton action recognition dataset, DD-Net does not stand out compared to other methods, because it is designed to be lightweight and efficient without using complex feature learning methods and deep neural networks, so the difference with the best method on the relatively small dataset Northwestern-UCLA is not as large as that on another large dataset. However, after applying the PCA-based view-invariant algorithm, the gesture recognition algorithm still has significant improvement.

4 Conclusions

In this paper, a PCA-based view-invariant algorithm is proposed. The method treats the whole skeleton sequence as a point set, obtains the feature matrix of the point set using the PCA algorithm, and then uses the feature matrix as a transformation matrix to transform the skeleton sequence to a relatively stable viewpoint. However, there are two kinds of ambiguities in the eigenmatrix generated from PCA, namely sign ambiguity and order ambiguity. In order to eliminate the effects of these two ambiguities, we propose to add all possible ambiguities to the training set to enhance the generalization ability of the model. We design several sets of experiments on two multi-view action recognition datasets to verify the effectiveness of this approach and analyze which ambiguities have a more significant impact on the performance, so as to find a balance between improving the performance and expanding the data. Finally, the PCA-based view-invariant algorithm is applied to the proposed gesture recognition model and compared with other methods. Although the gesture recognition performance of this paper still differs from the best method after applying the algorithm, there is still a significant improvement compared with that before applying the algorithm.

Acknowledgements. This work was supported by the National Key Research and Development Program of China under Grant No. 2020YFB1313602, The authors thank the reviewers for their valuable suggestions.

References

1. Chen, Y., Zhang, Z., Yuan, C., Li, B., Deng, Y., Hu, W.: Channel-wise topology refinement graph convolution for skeleton-based action recognition. In: Proceedings of the IEEE/CVF International Conference on Computer Vision, pp. 13359–13368 (2021)
2. Cheng, K., Zhang, Y., Cao, C., Shi, L., Cheng, J., Lu, H.: Decoupling GCN with DropGraph module for skeleton-based action recognition. In: Vedaldi, A., Bischof, H., Brox, T., Frahm, J.-M. (eds.) ECCV 2020. LNCS, vol. 12369, pp. 536–553. Springer, Cham (2020). https://doi.org/10.1007/978-3-030-58586-0_32
3. Cheng, K., Zhang, Y., He, X., Chen, W., Cheng, J., Lu, H.: Skeleton-based action recognition with shift graph convolutional network. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp. 183–192 (2020)
4. Du, Y., Wang, W., Wang, L.: Hierarchical recurrent neural network for skeleton based action recognition. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 1110–1118 (2015)
5. Ghorbel, E., Boutteau, R., Boonaert, J., Savatier, X., Lecoeuche, S.: Kinematic spline curves: a temporal invariant descriptor for fast action recognition. Image Vis. Comput. **77**, 60–71 (2018)
6. Ghorbel, E., et al.: A view-invariant framework for fast skeleton-based action recognition using a single RGB camera. In: Proceedings of the 14th International Joint Conference on Computer Vision, Imaging and Computer Graphics Theory and Applications, pp. 573–582 (2019)
7. Ji, Y., Xu, F., Yang, Y., Xie, N., Shen, H.T., Harada, T.: Attention transfer (ant) network for view-invariant action recognition. In: Proceedings of the 27th ACM International Conference on Multimedia, pp. 574–582 (2019)
8. Junejo, I.N., Dexter, E., Laptev, I., Pérez, P.: Cross-view action recognition from temporal self-similarities. In: Forsyth, D., Torr, P., Zisserman, A. (eds.) ECCV 2008. LNCS, vol. 5303, pp. 293–306. Springer, Heidelberg (2008). https://doi.org/10.1007/978-3-540-88688-4_22
9. Korban, M., Li, X.: DDGCN: a dynamic directed graph convolutional network for action recognition. In: Vedaldi, A., Bischof, H., Brox, T., Frahm, J.-M. (eds.) ECCV 2020. LNCS, vol. 12365, pp. 761–776. Springer, Cham (2020). https://doi.org/10.1007/978-3-030-58565-5_45
10. Lee, I., Kim, D., Kang, S., Lee, S.: Ensemble deep learning for skeleton-based action recognition using temporal sliding LSTM networks. In: Proceedings of the IEEE International Conference on Computer Vision, pp. 1012–1020 (2017)
11. Li, C., Zhong, Q., Xie, D., Pu, S.: Co-occurrence feature learning from skeleton data for action recognition and detection with hierarchical aggregation (2018)
12. Li, F., Fujiwara, K., Okura, F., Matsushita, Y.: A closer look at rotation-invariant deep point cloud analysis. In: Proceedings of the IEEE/CVF International Conference on Computer Vision, pp. 16218–16227 (2021)
13. Li, S., Li, W., Cook, C., Zhu, C., Gao, Y.: Independently recurrent neural network (indrnn): building a longer and deeper RNN. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 5457–5466 (2018)

14. Li, Y., Xia, R., Liu, X.: Learning shape and motion representations for view invariant skeleton-based action recognition. Pattern Recogn. **103**, 107293 (2020)
15. Papadakis, A., Mathe, E., Spyrou, E., Mylonas, P.: A geometric approach for cross-view human action recognition using deep learning. In: 2019 11th International Symposium on Image and Signal Processing and Analysis, pp. 258–263 (2019)
16. Xiaomin, P., Fan Huijie, T.Y.: Action recognition method of spatio-temporal feature fusion deep learning network. Infrared Laser Eng. **47**(2), 55–60 (2018)
17. Shahroudy, A., Liu, J., Ng, T.T., Wang, G.: NTU RGB+D: a large scale dataset for 3D human activity analysis. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 1010–1019 (2016)
18. Shao, Z., Li, Y., Zhang, H.: Learning representations from skeletal self-similarities for cross-view action recognition. IEEE Trans. Circuits Syst. Video Technol. **31**(1), 160–174 (2020)
19. Si, C., Chen, W., Wang, W., Wang, L., Tan, T.: An attention enhanced graph convolutional LSTM network for skeleton-based action recognition. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp. 1227–1236 (2019)
20. Veeriah, V., Zhuang, N., Qi, G.J.: Differential recurrent neural networks for action recognition. In: Proceedings of the IEEE International Conference on Computer Vision, pp. 4041–4049 (2015)
21. Vemulapalli, R., Arrate, F., Chellappa, R.: Human action recognition by representing 3D skeletons as points in a lie group. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 588–595 (2014)
22. Wang, J., Liu, Z., Wu, Y., Yuan, J.: Learning actionlet ensemble for 3D human action recognition. IEEE Trans. Pattern Anal. Mach. Intell. **36**(5), 914–927 (2013)
23. Wang, J., Nie, X., Xia, Y., Wu, Y., Zhu, S.C.: Cross-view action modeling, learning and recognition. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 2649–2656 (2014)
24. Xia, L., Chen, C., Aggarwal, J.K.: View invariant human action recognition using histograms of 3D joints. In: 2012 IEEE Computer Society Conference on Computer Vision and Pattern Recognition Workshops, pp. 20–27 (2012)
25. Yan, P., Khan, S.M., Shah, M.: Learning 4D action feature models for arbitrary view action recognition. In: 2008 IEEE Conference on Computer Vision and Pattern Recognition, pp. 1–7 (2008)
26. Yan, S., Xiong, Y., Lin, D.: Spatial temporal graph convolutional networks for skeleton-based action recognition. In: Proceedings of the Thirty-Second AAAI Conference on Artificial Intelligence, pp. 7444–7452 (2018)
27. Yang, F., Wu, Y., Sakti, S., Nakamura, S.: Make skeleton-based action recognition model smaller, faster and better. In: Proceedings of the ACM Multimedia Asia, pp. 1–6 (2019)
28. Zhang, P., Lan, C., Zeng, W., Xing, J., Xue, J., Zheng, N.: Semantics-guided neural networks for efficient skeleton-based human action recognition. In: proceedings of the IEEE/CVF conference on computer vision and pattern recognition, pp. 1112–1121 (2020)
29. Zhang, X., Xu, C., Tao, D.: Context aware graph convolution for skeleton-based action recognition. In: 2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp. 14321–14330 (2020)
30. Yi, Z., Shuo, Z., Yuan, L.: View-invariant 3D hand trajectory-based recognition. J. Univ. Electr. Sci. Technol. China **43**(1), 60–65 (2014)