Pattern Recognition Letters 140 (2020) 143–149

Contents lists available at ScienceDirect

Pattern Recognition Letters

journal homepage: www.elsevier.com/locate/patrec

3D PostureNet: A unified framework for skeleton-based posture recognition

Jianbo Liu[a,b], Ying Wang[a,*], Yongcheng Liu[a,b], Shiming Xiang[a], Chunhong Pan[a]

[a] National Laboratory of Pattern Recognition, Institute of Automation, Chinese Academy of Sciences, Beijing 100190, China
[b] School of Artificial Intelligence, University of Chinese Academy of Sciences, Beijing 100049, China

ARTICLE INFO

Article history:
Received 1 March 2020
Revised 17 September 2020
Accepted 25 September 2020
Available online 28 September 2020

MSC:
41A05
41A10
65D05
65D17

Keywords:
Human posture recognition
Static hand gesture recognition
Skeleton-based
3D convolutional neural network

ABSTRACT

Image-based posture recognition is a very challenging problem as it is difficult to acquire rich 3D information from postures in 2D images. Existing methods founded on 3D skeleton cues could alleviate this issue, but they are not particularly efficient due to the application of handcrafted features and traditional classifiers. This paper presents a novel and unified framework for skeleton-based posture recognition, applying powerful 3D Convolutional Neural Network (CNN) to this issue. Technically, bounding-box-based normalization for the raw skeleton data is proposed to eliminate the coordinate differences caused by diverse recording environments and posture displacements. Moreover, Gaussian voxelization for the skeleton is employed to expressively represent the posture configuration. Thereby, an end-to-end framework based on 3D CNN, called 3D PostureNet, is developed for robust posture recognition. To verify its effectiveness, a large-scale writing posture dataset is created and released in this work, including 113,400 samples of 30 subjects with 15 postures. Extensive experiments on the public MSRA hand gesture dataset, body pose dataset and the proposed writing posture dataset demonstrate that 3D PostureNet achieves significantly superior performance on both skeleton-based human posture and hand posture recognition tasks.

1. Introduction

Posture recognition mainly includes two research fields, i.e., human posture recognition and hand posture recognition. Both of them have been attractive topics for years due to their wide range of applications, such as human-machine interaction, ambient assisted living, intelligent health care systems and sign language recognition. In the last decade, various data modalities have been explored to facilitate posture recognition, in which the main ones are RGB image, depth map and skeleton.

Traditional works use RGB images and depth maps to characterize posture configurations, whose common pipeline is shown in the first row of Fig. 1. They usually focus on extracting low-level features from RGB and depth images, and then employing traditional classifiers for posture recognition [2,11,23]. Recently, some works also apply convolutional neural networks to posture recognition and achieve superior performance [3,7]. However, the RGB-D based methods may be less effective due to the inherent limitations of 2D images to model actual 3D postures.

In addition to the RGB image and depth map, skeleton is another data modality that is popularly applied for posture recognition. Skeleton data is utilized to build high level features for characterizing the 3D configurations of postures. In the existing literature, few works have been developed for skeleton-based posture recognition with the tricks of CNNs which are now popularly used in the field of computer vision. Previous works usually take the strategy that extracts a feature vector from joint positions, joint angles and joint distances [6,7,18,20], and then employs SVM or MLP for classification. The pipeline of this strategy is shown in the second row of Fig. 1. Though achieving some success, these methods have certain drawbacks such as losing the natural expression of the relative position of joints, lacking scalability for invisible joints.

To solve the aforementioned problems and exert the power of CNN to this issue, this paper develops a unified end-to-end framework called 3D PostureNet, by introducing Gaussian voxel modeling and 3D CNN for skeleton-based posture recognition. The pipeline of our method is shown in the third row of Fig. 1.

Specifically, in order to eliminate the coordinate differences of raw skeleton data caused by diverse recording environments and posture displacements, a bounding-box-based normalization method for raw skeleton data is proposed. When there are large

* Corresponding author.
E-mail address: ywang@nlpr.ia.ac.cn (Y. Wang).

https://doi.org/10.1016/j.patrec.2020.09.029
0167-8655/

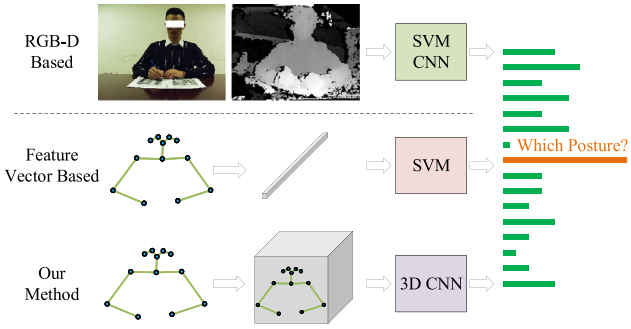

Fig. 1. The procedures of different methods for posture recognition. The first row is RGB-D based approach, the second row is feature vector based method and the third row is our method.

displacements among posture samples in original data, it can preserve the original scale among three dimensions while avoiding the normalized joints falling into a small area of [0, 1] cube space. Moreover, a Gaussian voxelization approach for skeleton modeling is devised. It can naturally reflect the mutual positional relationship of joints. As a consequence, the 3D PostureNet based on the Gaussian voxel representation can be highly effective for posture recognition. To validate our method, a large-scale writing posture dataset is proposed and released, including 113,400 samples with 30 subjects and 15 postures. Extensive experiments on the public MSRA hand gesture dataset, body pose dataset and the proposed writing posture dataset demonstrate that 3D PostureNet achieves superior performance.

In summary, the contributions of this paper are as follows:

- A bounding-box-based normalization method is proposed. It can eliminate the coordinate differences caused by diverse recording environments and posture displacements.
- A Gaussian voxelization approach is devised to represent posture configurations. It can naturally reflect the mutual positional relationship of joints.
- A unified end-to-end framework call 3D PostureNet is developed by introducing 3D CNN to learn on the Gaussian voxel representation of the skeleton.
- A large-scale writing posture dataset[1] including 113,400 samples of 30 subjects with 15 postures is proposed.

2. Related works

Human posture recognition. Early works for human posture recognition usually use images captured by conventional RGB cameras to model posture configurations [32]. These methods have inherent limitations for posture recognition since RGB images only provide 2D information rather than 3D information which is crucial to distinguish different postures. The emergence of RGB-D cameras (e.g. Microsoft Kinect) allows researchers to exploit depth information in computer vision tasks. In this context, recent approaches employ depth maps from RGB-D cameras to model human posture. Torres et al. [26] used multimodal data including RGB images, depth maps and pressure maps for sleep poses classification in an Intensive Care Unit (ICU) environment. Elforaici et al. [7] trained convolutional neural networks on RGB and depth images to recognize human postures. With the development of skeleton detection technology, skeleton data extracted from depth map is utilized to build high level features characterizing the 3D configurations of human body. Thus, skeleton is used for human posture recognition in many papers. Le et al. [18] recognized human postures using skeleton information provided by Kinect. They designed a multiclass SVM to classify the feature vector consisting of joint absolute coordinates. A new feature vector combining joint angles and the relative position of arm joints with respect to head was proposed by Mangera [20]. K-means classifier was used to identify each posture. Elforaici et al. [7] designed a handcraft feature vector which was composed of 3D pairwise distances between joints and the geometrical angles of adjacent limbs. Then SVM was used to perform posture recognition. Wang and Liu [28] collected bone information of the human body using the direction cosine method for feature extraction, feature vector was sent to the BP neural network for human body gesture recognition. Guerra et al. [10] used skeletal joint vertical coordinates and relative angles to represent each skeleton, a multi-layer perceptron with two hidden layers and a SoftMax output layer was built to classify the human posture. Ding et al. [5] proposed a new method based on multiple features (angle features and distance features) and rule learning. Esmaeili et al. [8] designed an ensemble model to combine 2D skeleton features and RGB features.

Hand posture recognition. Similar to human posture recognition, previous works for hand posture recognition extract low-level features from RGB or depth images. Traditional classifiers are then proposed to classify hand postures according to the features. Shukla and Dwivedi [24] computed hand features based on contour area and convexity defects. Pugeault and Bowden [23] proposed a large hand posture recognition dataset corresponding to letters of the American Sign Language (ASL) alphabet. Color and depth maps have been used to characterize hand shapes, with random forests chosen as classifiers. Wang and Yang [29] presented a 2D volumetric shape descriptor based on the polar representation of hand image and rotation invariance. Dong et al. [6] used a hierarchical mode-seeking method to localize hand joint positions and built a Random Forest (RF) classifier to recognize ASL signs using the joint angles. Feng et al. [9] generated depth projection maps to extract the bag of contour fragment descriptors, which were concatenated as a final shape representation of the original depth data. A SVM with a linear kernel was used as a shape classifier. Chevtchenko et al. [3] applied a convolutional neural network with feature fusion for real-time hand posture recognition. Dadashzadeh et al. [4] presented a fusion network for hand gesture segmentation and recognition. Kane and Khanna [15] recognized static hand gesture using depth matrix and 1-nearest neighbor strategy. Mirsu et al. [21] proposed a deep neural network by employing PointNet architecture for hand gesture recognition using depth data. Recently, skeleton-based methods get more attention. Kapuscinski and Organisciak [16] encoded differences of hand skeleton using finger directions and palm normal. Kapuściński and Warchoł [17] combined this feature with distance descriptor for static hand gesture recognition.

3. Proposed method

The proposed framework for skeleton-based posture recognition is shown in Fig. 2. The skeleton is normalized using bounding-box-based normalization to eliminate the coordinate differences caused by diverse recording environment and posture displacement. Afterwards, Gaussian voxel modeling for skeleton is executed to represent posture configurations. A simple but powerful 3D PostureNet based on 3D CNN is designed to classify the constructed 3D features.

3.1. Joint coordinate normalization

In 3D skeleton-based posture recognition, a posture is described as a collection of 3D positions of all joints in the skeleton. This

[1] https://drive.google.com/drive/folders/1x51kWIoa_eKm-UPvUt46N4VQiNxrfX0t?usp=sharing

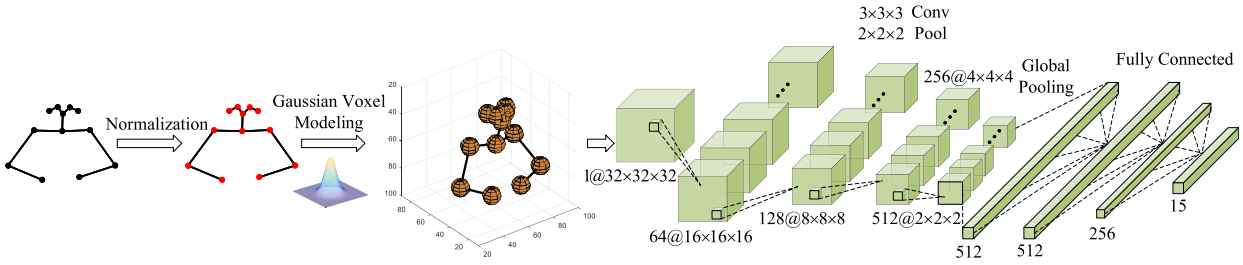

Fig. 2. The proposed framework for skeleton-based posture recognition. The skeleton is normalized using bounding-box-based normalization. Gaussian voxel modeling for skeleton is executed to represent posture configurations. Then, 3D PostureNet is used to classify the constructed 3D features. For network architecture of the 3D PostureNet, $3 \times 3 \times 3$ Conv represents 3D filter with kernel size of 3, $2 \times 2 \times 2$ Pool is a 3D max pooling layer, 64@16 $\times$ 16 $\times$ 16 means a set of 16 $\times$ 16 $\times$ 16 3D features with 64 channels.

representation depends on the choice of the reference coordinate system, which is different in every recording environment, and on biometric differences [22]. However, given a specific posture, it is independent of the absolute spatial position of joints but is dependent on the relative position among all joints. Generally, the absolute 3D coordinates of joints are based on the coordinate system of the data acquisition camera. In this context, a variety of normalization methods are proposed to eliminate the recording environment and biometric differences in skeleton data. Wei et al. normalized human poses by aligning the torsos and the shoulders [30]. Wu and Shao [31] changed the coordinate system from the world coordinate system to person centric coordinate system by placing the hip center at the origin. Besides, skeletons were normalized by the head length and aligned based on the head location [27]. Hussein et al. [13] normalized the 3D coordinates of joints to range from 0 to 1 in all dimensions to make it scale-invariant.

In this paper, we attempt to model skeleton in a cube volume as the input of 3D convolutional neural networks. Hence, the coordinates of joints are supposed to be normalized to range in [0,1] and then multiplied by a scale factor to be embedded into a cube space with a specific resolution. The normalization approach proposed by Wang et al. [27] which used limb length (e.g. head length) for skeleton normalization is robust to eliminate biometric differences. While this method has a strictly restricted condition for the selected limb, that is the limb should be visible in all skeleton samples. Thus, it is not applicable to our writing posture dataset, since a few joints are invisible for some postures and there is no constant visible limb for all skeleton samples. As a result, a new normalization approach without referring to limb information should be proposed. In this work, several normalization strategies are explored.

Global normalization. The simplest and most straightforward strategy is global normalization. In this strategy, joint coordinates in each dimension are normalized by the corresponding minimum and maximum values of the entire training data. Specifically, the x-coordinate is normalized as follows,

$$x_{norm} = \frac{x - G^x_{\min}}{G^x_{\max} - G^x_{\min}}, \tag{1}$$

where $G^x_{\min}$ and $G^x_{\max}$ are the minimum and maximum x-coordinate values of the joints in the whole training data, x is the original x-coordinate, and x_{norm} is the normalized x-coordinate. The coordinates of Y and Z are processed in the same way. This method has an obvious drawback when there are large displacements among posture samples of original data. In that case, the normalized joints will gather in a small area of [0, 1] cube space leaving most regions empty.

Local normalization. Another natural normalization strategy is local normalization which normalizes each skeleton using its own minimum and maximum values of coordinate. If the minimum and maximum x-coordinate values of this skeleton are defined as $x_{\min}$, $x_{\max}$ respectively, the x-coordinates of all joints in this skeleton are normalized by

$$x_{norm} = \frac{x - x_{\min}}{x_{\max} - x_{\min}}. \tag{2}$$

Joint coordinates in all three dimensions will be transformed to [0, 1] for every skeleton. Although joints will be dispersed as much as possible via this strategy, it has a fatal weakness as well. Joint coordinates in each dimension are standardized using different scale factors. It will destroy the scale ratio among the three dimensions, which is a crucial feature to characterize the posture skeleton.

Bounding-box-based normalization. In order to solve the issue caused by global and local normalization, an innovative normalization strategy based on the bounding box of skeleton is proposed. The main idea of this approach is to normalize skeletons using the maximum side length of the bounding box for skeletons in all training data. Concretely, for the given ith skeleton in training data, the minimum and maximum coordinate values of three dimensions for this skeleton are defined as $x^i_{\min}$, $x^i_{\max}$, $y^i_{\min}$, $y^i_{\max}$, $z^i_{\min}$, $z^i_{\max}$ respectively. The maximum side length of the bounding box for this skeleton is defined as

$$l_i = \max\left(x^i_{\max} - x^i_{\min}, y^i_{\max} - y^i_{\min}, z^i_{\max} - z^i_{\min}\right). \tag{3}$$

Hence, the scale factor used for normalization can be written as $L = \max(l_i)$, where $i = 1, \ldots, N$, N is the number of samples in total training data. Finally, the x-coordinate of joints for each skeleton are normalized as follows,

$$x_{norm} = \frac{x - \frac{x_{\min} + x_{\max}}{2}}{L} + 0.5, \tag{4}$$

where $x_{\min}$, $x_{\max}$ are the minimum and maximum x-coordinate values of this skeleton, x is the original x-coordinate, and x_{norm} is the normalized x-coordinate. The coordinates of Y and Z are processed in the same way with the coordinate of X. In this way, each skeleton is normalized by a constant L and the center of the skeleton is aligned to (0.5, 0.5, 0.5) which is the center of the [0, 1] cube space. The shift among postures is eliminated via this normalization strategy. Meanwhile, the structure characteristic of the skeleton remains consistent owing to the constant scale factor in all three dimensions.

3.2. Gaussian voxel modeling for skeleton

Previous works for skeleton-based posture recognition usually use a method where a feature vector is extracted from joint positions, joint angles and joint distances. The feature vector is treated as the input of a traditional classifier e.g. KNN, SVM and MLP. These methods have the following disadvantages:

- Using a vector as an input can't express the relative location of joints naturally.
- The structure of the classifier depends on the input form. Therefore, the classifier structure has to be in line with the vector size which is determined by the number of involved joints. That is, these methods lack scalability due to the fixed vector size.
- The input vector can't express invisible joints well. As is discussed above, the vector size isn't flexible, the coordinates of all joints should be valued even when there are invisible joints. That means the vector may contain some meaningless entries corresponding to invisible joints.

Voxel modeling. To overcome the drawbacks discussed above, the skeleton is proposed to be modeled in voxel space. It is inspired by the volumetric representation for the geometric object in computer graphics community. The process of generating volumetric representation from the geometric object is typically called voxelization. It converts a continuous geometric model to a set of voxels in the 3D discrete space with a specific resolution. Meanwhile, the appearance and shape feature of the original object is maintained as much as possible. In this context, the skeleton is transformed into a volume occupancy grid. That is, the skeleton is put into a voxel space, the value of the occupied voxels equal to 1 and the values of the rest voxels equal to 0.

More specifically, a skeleton data with J joints can be represented as a set $S = \{\mathbf{p}_n | \mathbf{p}_n = (x_n, y_n, z_n, v_n), n = 1, \ldots, J\}$, where $\mathbf{p}_n$ is a vector consists of 3D coordinate (x_n, y_n, z_n) and visible label v_n of the nth joint. $v_n = 1$ indicates that the joint is visible. On the contrary, when $v_n = 0$, the joint is invisible. It's worth noting that $x_n, y_n, z_n \in [0, 1]$ due to the process of coordinate normalization. For simplicity, the skeleton is embedded in a cube voxel space with $M \times M \times M$ resolution. Hence, it is necessary to discretize the 3D coordinate of joint to apply to the resolution. The discretization procedure for coordinates can be conducted as follows,

$$\begin{cases} x_n^d = \lfloor x_n \times (M-1) \rfloor, \\ y_n^d = \lfloor y_n \times (M-1) \rfloor, \\ z_n^d = \lfloor z_n \times (M-1) \rfloor, \end{cases} \tag{5}$$

where $n = 1, \ldots, J$ and x_n^d, y_n^d, z_n^d are discretized coordinates, $x_n^d, y_n^d, z_n^d \in \{0, 1, \ldots, M-1\}$. Simultaneously, the discretized skeleton data is turned into $S^d = \{\mathbf{p}_n^d | \mathbf{p}_n^d = (x_n^d, y_n^d, z_n^d, v_n), n = 1, \ldots, J\}$. After voxel modeling, the volumetric representation for the skeleton can be represented as a tensor T of (M, M, M) dimension. The entry with index (i, j, k) of tensor T can be written as Eq. (6), where $i, j, k = 0, \ldots, M-1$.

$$T(i, j, k) = \begin{cases} 1 & \text{if } (i, j, k, 1) \in S^d, \\ 0 & \text{otherwise.} \end{cases} \tag{6}$$

Gaussian voxel modeling. By voxel modeling for skeleton, we overcome the weaknesses of the prior approaches successfully which extracts features from skeleton to form a vector. The relative positions between joints are reflected naturally since every joint is embedded in the same 3D voxel space as a voxel cell. In addition, the size of feature tensor is fixed as $M \times M \times M$, which is independent of the number of involved joints. This improves the scalability of subsequent classifier significantly. As for invisible joints, only the voxel corresponding to the visible joint has a value of 1, and the values of other voxels are 0, so the invisible joints remain invisible in the voxel space as well. This technique distinguishes visible and invisible joints visually. However, extensive entries of the tensor T constructed above have a value of 0. As a result, when using a high resolution, the tensor T will be too sparse for the subsequent classifier to learn an effective feature for posture classification. This motivates us to propose Gaussian voxel modeling for skeleton.

The main idea of Gaussian voxel modeling for skeleton is to spread the value of visible voxel cells to their neighbor voxels using 3D Gaussian function. In detail, for the nth joint $\mathbf{p}_n^d = (x_n^d, y_n^d, z_n^d, v_n)$, a tensor T_n is created to characterize this joint according to the formula as follows,

$$T_n(i, j, k) = \begin{cases} \exp\left(-\frac{(i-x_n^d)^2}{2\sigma^2} - \frac{(j-y_n^d)^2}{2\sigma^2} - \frac{(k-z_n^d)^2}{2\sigma^2}\right) & \text{if } v_n = 1, \\ 0 & \text{if } v_n = 0, \end{cases} \tag{7}$$

where $i, j, k = 0, \ldots, M-1$, and σ is the variance coefficient in the direction of X, Y, Z coordinate axes. For skeleton $S^d = \{\mathbf{p}_n^d | \mathbf{p}_n^d = (x_n^d, y_n^d, z_n^d, v_n), n = 1, \ldots, J\}$, T_n is integrated together to characterize the configurations of the skeleton. Finally, the tensor T_G used to model the skeleton in 3D voxel space is obtained according to Eq. (8). $T_G(i, j, k)$ is the entry value with index (i, j, k) in tensor T_G.

$$T_G(i, j, k) = \max\left(T_n(i, j, k)\right), n = 1, \ldots, J. \tag{8}$$

3.3. 3D PostureNet

After generating the 3D Gaussian voxel feature, it is fed to our 3D PostureNet to classify postures. The 3D PostureNet is based on 3D convolutional neural network, which is used to aggregate local 3D features with a 3D convolutional kernel. 3D CNN is first proposed for action recognition [14], where features from the spatial and temporal dimensions are extracted by performing 3D convolutions. After that, studies based on 3D CNN emerge in large numbers. Recently, Li et al. [19] used separable 3D CNN to extract spatial and spectral information for hyperspectral image super-resolution. In this paper, 3D CNN is used to capture the structure information of the posture skeleton encoded in the 3D Gaussian voxel feature.

The network architecture of the proposed 3D PostureNet is shown in Fig. 2. For simplicity, a network with an input resolution of $32 \times 32 \times 32$ and an output size of 15 is illustrated. With regard to other input resolution and output size, the feature size of the hidden layers can be inferred effortlessly. As shown in Fig. 2, four 3D convolutional layers with a kernel size of $3 \times 3 \times 3$ are applied to the 3D input to extract high level features gradually for the skeleton. The sizes of the output channels of four 3D convolutional layers are 64, 128, 256 and 512. Each 3D convolutional layer is followed by a $2 \times 2 \times 2$ max pooling layer to reduce the size of the feature. The output of the last 3D convolutional layer is transformed into a feature vector using global max pooling. The network is designed to be scalable to the inputs with different resolution using global max pooling layer. After that, three fully connected layers with sizes of 512, 256 and 15 are used to classify the posture skeleton. ReLU is applied to the output of every layer except for the last layer. The last output layer uses Softmax as an activation function, and cross entropy loss is applied. In order to avoid overfitting, dropout is used between the second and the third 3D convolutional layers as well as the fully connected layers.

4. Experiments

In this section, three datasets are introduced, one of which is a large scale writing posture dataset including 113,400 samples of 30 subjects with 15 postures collected by us. The others are MSRA hand gesture dataset and body pose dataset which are publicly available. Then, comparison experiments are performed against other methods. Ablation experiments and analyses are provided as well.

4.1. Datasets

Writing posture dataset. Most of the existing works for skeleton-based posture recognition performed quantitative evalu-

ation only on small datasets [7,18,20,26], which contain less than 10 human postures. The datasets involve a small number of subjects and the skeletons are always visible, which leads to a lack of challenge. Therefore, in this work, a large and more challenging dataset is collected for writing posture recognition.

The writing posture dataset was collected using a calibrated binocular camera in a laboratory environment, RGB video was recorded for each subject. We performed human pose estimation on the RGB videos using OpenPose [1] to get the joint positions and visibility. By binocular reconstruction, 3D skeleton data was provided. In total, it consists of 113,400 images captured from 30 subjects with different genders and heights. Skeleton data and posture category for each frame are provided. During our data capture, each subject was asked to perform one of the 15 writing postures each time. For each writing posture, about 300 frames were recorded. The skeleton data contains 12 body joints, and each joint is recorded as a 3D coordinate (x, y, z) in the coordinate system centered on the binocular camera. The 15 writing postures with corresponding skeletons are shown in Fig. 3. It shows that joints suffer from severe occlusion in some postures such as head down, lying down, turning left, turning backward, turning right and standing up. This increases the difficulty to recognize the postures. During training, the data of the first 7 subjects were used for testing, and the rest data of 23 subjects were used for training.

MSRA hand gesture dataset. In order to verify the effectiveness and scalability of our method, we perform hand posture recognition on a challenging dataset which was proposed by Sun et al. [25]. This dataset consists of 76, 500 depth images captured from 9 subjects, with 17 hand gestures that are mostly from American Sign Language. The dataset was originally collected for hand pose estimation using depth images. It provides skeleton data with 3D joint coordinates for each posture. Containing all the information for hand posture recognition, this dataset can be used to further evaluate our method. As is shown in Fig. 4, the dataset has large viewpoint variations. It is a challenging task to distinguish

Fig. 3. Illustration of the 15 postures in writing posture dataset. The corresponding skeleton is drawn on each image.

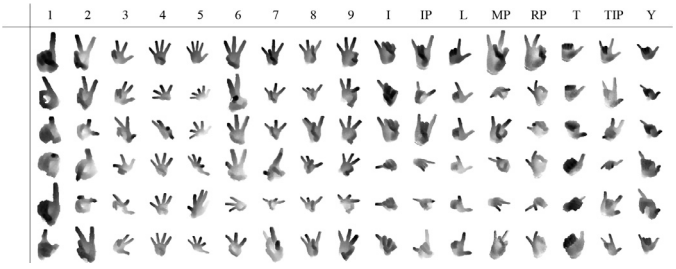

Fig. 4. Illustration of the 17 hand gestures in MSRA hand gesture dataset. Each column displays six depth images with various viewpoints corresponding to the specific posture.

different kinds of gestures in this dataset. During training, the data of the first 2 subjects were used for testing, and the rest data of 7 subjects were used for training. Due to the large viewpoint variations in the dataset, data argumentation was performed during the training stage. While training the network, the skeleton from the training set was randomly rotated an angle from -45 degrees to 45 degrees around X-axis, Y-axis and Z-axis respectively.

Body pose dataset. Body pose dataset [12] has 12 recorded subjects performing 10 different standstill body poses. The dataset provides image, depth map, skeleton and pose label for each pose sample. It provides 8400 pose samples in total. We use the data of the first 3 subjects for testing, and the rest data of 9 subjects for training.

4.2. Comparison with the state of the art

In order to evaluate and compare our method with previous methods proposed by other researchers, experiments were conducted on the writing posture dataset, MSRA hand gesture dataset and body pose dataset. The selected works for comparison are listed in Table 1.

Mangera [20] proposed a new feature vector combining joint angles and the relative positions of arm joints with respect to head. The feature vector then inputs to the k-means classifier to cluster each posture. In our experiments, the joint nose, wrist and head were chosen as the reference joints in writing posture dataset, MSRA hand gesture dataset and body pose dataset respectively. The relative position of all the other joints with respect to the reference joint was used in the three datasets. The skeleton-based approaches in Table 1 didn't take into consideration the invisible joints. To evaluate these methods on our writing posture dataset which contains plenty of postures with invisible joints, the values of joint absolute coordinates, joint angles and joint distances are fixed to 0 when encountering invisible joints. We implemented these methods and applied them to the above three datasets respectively.

The mean accuracy of the above methods as well as our approach is listed in Table 1. The data shows that our approach outperforms the state of the arts with a larger margin. Our method achieved the highest accuracy on the three datasets with 97.77% on the writing posture dataset, 98.56% on the MSRA hand gesture dataset and 98.16% on the body pose dataset. The performance of our method is better than the skeleton-based methods as well as the RGB and depth-based methods. It is clear that our approach has better effectiveness on skeleton modeling and posture recognition.

To further analyze the results on the proposed writing posture dataset, we provide the confusion matrix of the predictions in Fig. 5. It can be seen that our method can distinguish most of posture categories effectively. For posture "Sloping left shoulder" and "Sloping right shoulder", our method confuses these postures with postures "Head tilted left" and "Head tilted right", since these posture skeletons are similar in spatial configurations.

4.3. Ablation study

Normalization strategies. In this work, three normalization strategies are presented, i.e., global normalization, local normalization and bounding-box-based normalization. As analyze above, global normalization and local normalization have inherent disadvantages. Global normalization has an obvious drawback when there are large displacements among posture samples. The normalized joints will be gathered in a small area of [0, 1] cube space leaving most regions empty. This will increase the difficulty for the network to recognize the postures. With regard to local normalization, joint coordinates in all three dimensions will be transformed

Table 1
Recognition accuracy comparison with the state-of-the-art methods on the writing posture dataset (WPD), MSRA hand gesture dataset (MSRA-HGD) and body pose dataset (BPD).

Method	Modality	Features	Classifier	WPD(%)	MSRA-HGD(%)	BPD(%)
Elforaici et al.	RGB	–	CNN	91.82	–	92.43
Esmaeili et al.	RGB	–	CNN	94.55	–	94.96
Chevtchenko et al.	Depth	–	CNN	93.28	92.35	93.26
Le et al.	skeleton	absolute coordinates	SVM	85.65	83.72	86.41
Dong et al.	skeleton	joint angles	RF	84.37	85.64	85.63
Mangera	skeleton	relative positions and joint angles	K-means	86.15	87.78	88.82
Elforaici et al.	skeleton	joint distances and angles	SVM	87.03	87.17	89.33
Wang and Liu	skeleton	bone angle	BP NN	89.95	88.58	90.25
Kapuscinski and Organisciak	skeleton	finger directions and palm normal	SVM	–	90.67	–
Kapuściński and Warchoł	skeleton	finger directions and distances	SVM	–	93.27	–
3D PostureNet	skeleton	Gaussian voxel feature	3D CNN	**97.77**	**98.56**	**98.16**

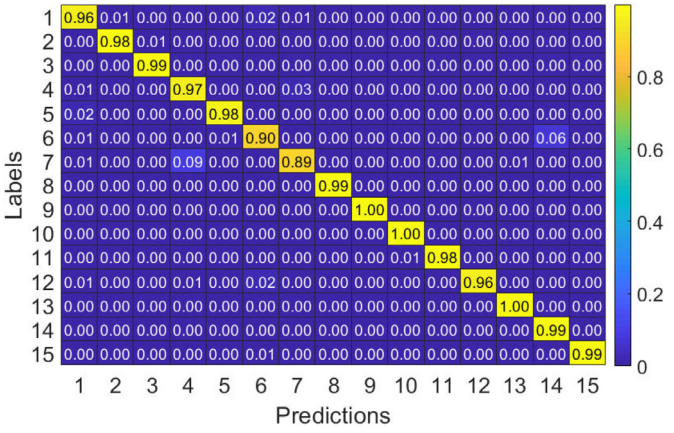

Fig. 5. Confusion Matrix of predication results of our method on the writing posture dataset. The posture labels corresponds to Fig. 3. Best seen on computer, in color and zoomed in.

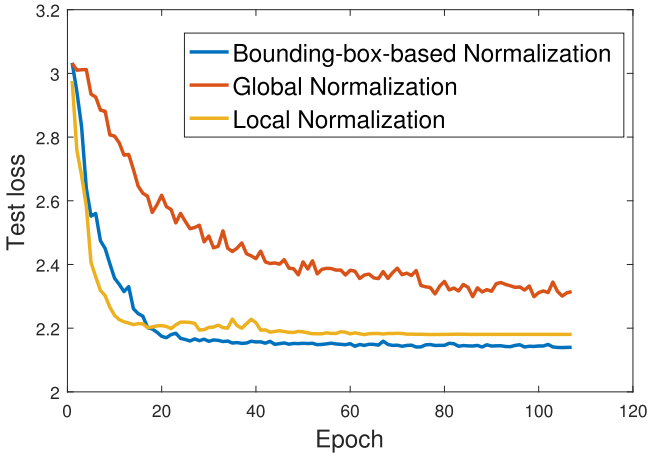

Fig. 6. The test loss curve with different normalization strategies during training on the MSRA hand gesture dataset.

Table 2
Recognition accuracy with different normalization strategies and rotation augmentation on both the writing posture dataset and MSRA hand gesture dataset.

Normalization Strategies	WPD	MSRA-HGD
Global normalization	96.41%	83.82%
Local normalization	7.65%	94.34%
Bounding-box-based normalization	**97.77**%	**98.56**%
without rotation augmentation	97.15%	95.32%

Table 3
Recognition accuracy with different variances on both the writing posture dataset and MSRA hand gesture dataset.

Variance	WPD	MSRA-HGD
$\sigma = 2.0$	96.58%	98.08%
$\sigma = 1.5$	97.10%	98.27%
$\sigma = 1.0$	97.17%	**98.56**%
$\sigma = 0.5$	**97.77**%	97.93%
sparse	91.35%	91.95%

to [0, 1] for every skeleton. Although joints will be dispersed as much as possible via this strategy, it has a fatal weakness as well. Joint coordinates in each dimension are standardized using a different scale factor. This will result in the destruction of the relative position between joints which is a crucial feature to characterize the posture skeleton. The bounding-box-based normalization is presented to address these problems. In order to evaluate the performance of the three normalization strategies, experiments were conducted using different normalization strategies on two datasets. The results are given in Table 2. Bounding-box-based normalization achieves the best performance on both datasets. The result is in line with the above analysis. Duo to the large displacements among gesture samples in MSRA hand gesture dataset, the performance deteriorates with the global normalization. Note that local normalization failed on the writing posture dataset, the network didn't converge. Compared to MSRA hand gesture dataset, the scale factors of the three dimensions differ considerably for the writing posture dataset. It leads to severe destruction of the relative position between joints. Besides, rotation augmentation shows about 3% improvement on MSRA hand gesture dataset but little improvement on the writing posture dataset, since the MSRA hand gesture dataset has large viewpoint variations. To further analyze the effectiveness of the bounding-box-based normalization, Fig. 6 shows the test loss curve with different normalization strategies during training on the MSRA hand gesture dataset. The test loss curve with bounding-box-based normalization is smoother than global normalization and local normalization. It shows that test data is encoded closer to train data using bounding-box-based normalization.

Variance and resolution. To explore the influence of the variance and input resolution in Gaussian voxel modeling on the performance, ablation experiments with different variances and resolution were conducted. When performing ablation experiments with various variances, the input resolution was fixed to $32 \times 32 \times 32$. Similarly, the variance was fixed to 1.0 when exploring the influence of input resolution. The result for different variances and resolution are listed in Tables 3 and 4 respectively. These results clearly show that the best variance and input reso-

Table 4
Recognition accuracy with different input resolution on both the writing posture dataset and MSRA hand gesture dataset.

Resolution	WPD	MSRA-HGD
16 × 16 × 16	95.69%	98.13%
24 × 24 × 24	96.37%	98.22%
32 × 32 × 32	97.17%	**98.56**%
40 × 40 × 40	**97.57**%	97.40%

Table 5
Average inference time of different methods.

Method	Inference time (ms)
Elforaici et al.	158
Esmaeili et al.	363
Chevtchenko et al.	174
Le et al.	26
Dong et al.	6
Mangera	12
Elforaici et al.	25
Wang and Liu	23
Kapuscinski and Organisciak	27
Kapuściński and Warchoł	63
3D PostureNet	87

lution for MSRA hand gesture dataset are 1.0 and 32 × 32 × 32. Compare with MSRA hand gesture dataset, smaller variance and higher resolution lead to better performance on the writing posture dataset. Some of the postures (e.g., turning left and turning right) in the writing posture dataset have joints too adjacent to each other, which requires a smaller variance and a higher resolution to distinguish each joint.

4.4. Computational efficiency

To investigate the computational efficiency of the proposed method, we list the average inference time of different methods in Table 5. Experiments are conducted on a PC with an Intel i7-7700HQ (2.8GHz) CPU and 16GB RAM. Due to the use of deep learning, our method is not superior in speed compared with other approaches based on traditional classifiers. However, our method achieves superior performance with an acceptable increase on computation complexity.

5. Conclusion

In this paper, we have proposed a novel framework for skeleton-based posture recognition using 3D CNN. To eliminate the coordinate differences caused by diverse recording environments and posture displacements, the skeleton is normalized using bounding-box-based normalization. Gaussian voxel modeling for skeleton is executed to represent posture configurations. Afterwards, a simple but powerful 3D PostureNet based on 3D CNN is designed to classify the constructed 3D features. To verify the effectiveness of our method, a large-scale writing posture dataset including 113,400 samples of 30 subjects with 15 postures was collected. Experiments on the writing posture dataset, MSRA hand gesture dataset and body pose dataset show that our method achieves superior performance on both skeleton-based human posture and hand posture recognition tasks.

Declaration of Competing Interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Acknowledgements

This research was supported by the Major Project for New Generation of AI under Grant No. 2018AAA0100400, the National Key Research and Development Program under Grant No. 2016YFB0501100, and the National Natural Science Foundation of China under Grants 91646207, and 61976208.

References

[1] Z. Cao, T. Simon, S.-E. Wei, Y. Sheikh, Realtime multi-person 2d pose estimation using part affinity fields, in: CVPR, 2017, pp. 1302–1310.
[2] K.W. Chen, X. Guo, J.G. Wu, Gesture recognition system based on wavelet moment, in: Applied Mechanics and Materials, vol. 401, Trans Tech Publ, 2013, pp. 1377–1380.
[3] S.F. Chevtchenko, R.F. Vale, V. Macario, F.R. Cordeiro, A convolutional neural network with feature fusion for real-time hand posture recognition, Appl. Soft Comput. 73 (2018) 748–766.
[4] A. Dadashzadeh, A.T. Targhi, M. Tahmasbi, M. Mirmehdi, HGR-Net: a fusion network for hand gesture segmentation and recognition, IET Comput. Vis. 13 (8) (2019) 700–707.
[5] W. Ding, B. Hu, H.Y. Liu, X. Wang, X. Huang, Human posture recognition based on multiple features and rule learning, Int. J. Mach. Learn. Cybern. (2020).
[6] C. Dong, M.C. Leu, Z. Yin, American sign language alphabet recognition using microsoft kinect, in: CVPRW, 2015, pp. 44–52.
[7] M.E.A. Elforaici, I. Chaaraoui, W. Bouachir, Y. Ouakrim, N. Mezghani, Posture recognition using an RGB-D camera: exploring 3d body modeling and deep learning approaches, in: LSC, IEEE, 2018, pp. 69–72.
[8] B. Esmaeili, A. AkhavanPour, A. Bosaghzadeh, An ensemble model for human posture recognition, in: MVIP, IEEE, 2020, pp. 1–7.
[9] B. Feng, F. He, X. Wang, Y. Wu, H. Wang, S. Yi, W. Liu, Depth-projection-map-based bag of contour fragments for robust hand gesture recognition, IEEE Trans. Hum. Mach. Syst. 47 (4) (2017) 511–523.
[10] B.M.V. Guerra, S. Ramat, G. Beltrami, M. Schmid, Automatic pose recognition for monitoring dangerous situations in ambient-assisted living, Front. Bioeng. Biotechnol. 8 (2020) 415.
[11] P. Gurjal, K. Kunnur, Real time hand gesture recognition using sift, Int. J. Electron. Electr.Eng. 2 (3) (2012) 19–33.
[12] J.R. Hidalgo, J.R. Casas, Body pose dataset, image processing group, Accessed 1 July 2020. http://imatge.upc.edu/web/resources/body-pose-dataset.
[13] M.E. Hussein, M. Torki, M.A. Gowayyed, M. El-Saban, Human action recognition using a temporal hierarchy of covariance descriptors on 3d joint locations, in: IJCAI, 2013, pp. 2466–2472.
[14] S. Ji, W. Xu, M. Yang, K. Yu, 3D convolutional neural networks for human action recognition, IEEE Trans. Pattern Anal. Mach. Intell. 35 (1) (2013) 221–231.
[15] L. Kane, P. Khanna, Depth matrix and adaptive bayes classifier based dynamic hand gesture recognition, Pattern Recognit. Lett. 120 (2019) 24–30.
[16] T. Kapuscinski, P. Organisciak, Handshape recognition using skeletal data, Sensors 18 (8) (2018) 2577.
[17] T. Kapuściński, D. Warchoł, Hand posture recognition using skeletal data and distance descriptor, Appl. Sci. 10 (6) (2020) 2132.
[18] T.-L. Le, M.-Q. Nguyen, T.-T.-M. Nguyen, Human posture recognition using human skeleton provided by kinect, in: ComManTel, 2013, pp. 340–345.
[19] Q. Li, Q. Wang, X. Li, Mixed 2d/3d convolutional network for hyperspectral image super-resolution, Remote Sens. 12 (10) (2020) 1660.
[20] R. Mangera, Static gesture recognition using features extracted from skeletal data, 2013.
[21] R. Mirsu, G. Simion, C.D. Caleanu, I.M. Pop-Calimanu, A pointnet-based solution for 3d hand gesture recognition, Sensors 20 (11) (2020) 3226.
[22] L.L. Presti, M. La Cascia, 3D skeleton-based human action classification: asurvey, Pattern Recognit. 53 (2016) 130–147.
[23] N. Pugeault, R. Bowden, Spelling it out: real-time ASL fingerspelling recognition, in: ICCVW, IEEE, 2011, pp. 1114–1119.
[24] J. Shukla, A. Dwivedi, A method for hand gesture recognition, in: CSNT, IEEE, 2014, pp. 919–923.
[25] X. Sun, Y. Wei, S. Liang, X. Tang, J. Sun, Cascaded hand pose regression, in: CVPR, 2015, pp. 824–832.
[26] C. Torres, V. Fragoso, S.D. Hammond, J.C. Fried, B. Manjunath, Eye-CU: sleep pose classification for healthcare using multimodal multiview data, in: WACV, IEEE, 2016, pp. 1–9.
[27] C. Wang, Y. Wang, A.L. Yuille, An approach to pose-based action recognition, in: CVPR, 2013, pp. 915–922.
[28] J. Wang, X.H. Liu, Human posture recognition method based on skeleton vector with depth sensor, IOP Conf. Ser. Mater. Sci.Eng. 806 (2020) 012035.
[29] Y. Wang, R. Yang, Real-time hand posture recognition based on hand dominant line using kinect, in: ICMEW, IEEE, 2013, pp. 1–4.
[30] P. Wei, N. Zheng, Y. Zhao, S.-C. Zhu, Concurrent action detection with structural prediction, in: ICCV, 2013, pp. 3136–3143.
[31] D. Wu, L. Shao, Leveraging hierarchical parametric networks for skeletal joints based action segmentation and recognition, in: CVPR, 2014, pp. 724–731.
[32] M. Yu, A. Rhuma, S.M. Naqvi, L. Wang, J. Chambers, A posture recognition-based fall detection system for monitoring an elderly person in a smart home environment, IEEE Trans. Inf. Technol. Biomed. 16 (6) (2012) 1274–1286.