International Journal of Engineering

Journal Homepage: www.ije.ir

Estimation of Hand Skeletal Postures by Using Deep Convolutional Neural Networks

A. Gheitasi, H. Farsi*, S. Mohamadzadeh

Department of Electrical and Computer Engineering, University of Birjand, Birjand, Iran

P A P E R I N F O

Paper history:
Received 07 May 2019
Received in revised form 04 February 2020
Accepted 06 March 2020

Keywords:
Deep Convolutional Neural Network
Deep Learning
Hand Posture Recognition
Skeletal Estimation

A B S T R A C T

Hand posture estimation attracts researchers because of its many applications. Hand posture recognition systems simulate the hand postures by using mathematical algorithms. Convolutional neural networks have provided the best results in the hand posture recognition so far. In this paper, we propose a new method to estimate the hand skeletal posture by using deep convolutional neural networks. To simplify the proposed method and to be more functional, the depth factor is ignored. So only the simple color images of hands are used as inputs of the system. The proposed method is evaluated by using two datasets with high-diversity named Mixamo and RWTH, which include 43,986 and 1160 color images, respectively, where 74% of these images are selected as a training set and, 26% of the rest images are selected as the evaluation set. The experiments show that the proposed method provides better results in both hand posture recognition and detection of sign languages compared to state-of-the-art methods.

***doi**: 10.5829/ije.2020.33.04a.06*

1. INTRODUCTION

In recent years there has been tremendous growth in devices and techniques for human-computer interaction [1, 2]. Human action recognition has attracted the attention of researchers due to its potential and applications [3]. In particular, interface incorporating hand postures have obtained popularity in many fields [2]. Hand postures provide dominant communication modes for human-computer interaction. Traditional input devices which are available for interaction with the computer are unable to provide a natural interface [4]. In particular, a hand posture-based human and machine interface is more intuitive, natural, and intelligent way than the traditional interface methodologies. Estimating hand pose is difficult because of many freedoms in movements and complex calculations. The process of hand posture recognition is performed by machine-vision and image processing. There are two approaches for hand posture recognition; sensor-based and vision-based methods [5]. In the first approach, particular sensors are used, and users may need to wear the glove or may use the web camera for capturing the hand image. Sensor devices are used in data-glove based methods for digitizing hand and finger motions into multi-parametric data. The other sensors will collect hand configuration and movements.

In contrast, the vision-based methods only require the camera, without using any other devices [4]. The challenging problems of vision-based systems to achieve high performance include: needing to have background invariant, lightening insensitive, person and camera independence and etc. The hand posture recognition process reduces the chance of error, increases system robustness, and stability, and improves saving time. Often a vision-based system is developed as a combined RGB and depth descriptor. In this case, the RGBD dataset has been used [2, 3]. However, in this paper for simplification, we ignore the depth factor. If the depth factor is used in the hand posture detection process, a depth camera should also be used to store depth information of images. Since the depth cameras are not as commonly available as regular color cameras, and they only work reliably in indoor environments [6], we decided to remove the depth factor, although with removing the depth factor, we faced lacking

*Corresponding Author Email: *hfarsi@birjand.ac.ir* (H. Farsi)

Please cite this article as: A. Gheitasi, H. Farsi, S. Mohamadzadeh, Estimation of Hand Skeletal Postures by Using Deep Convolutional Neural Networks, International Journal of Engineering (IJE), IJE TRANSACTIONS A: Basics Vol. 33, No. 4, (April 2020) 552-559

and ambiguity. In order to recover these lacks, deep convolutional neural networks are used.

The purpose of this paper is to determine the posture of hand by using its skeletal estimation with Deep Convolutional Neural Network (DCNN) algorithms. All studied postures in this paper are static, and the data includes RGB color images. In the proposed method, the problem of hand posture recognition is divided into three sub-problems, which are solved by using DCNNs. After segmenting the hand by using sampling windows, we encounter the first sub-problem, which is extracting the hand mask. In this part, the hand mask that contains 33 parts is extracted; after that, 21 key points of the hand in 2D space are extracted, and finally, in the third sub-problem, 2D information is transferred into 3D space, and the hand posture is estimated.

This paper is organized as follows: Section 2 briefly reviews the related works on posture recognition based on deep learning. Details of the proposed method are described in Section 3. Datasets, evaluation criteria, experimental results, and comparisons are presented in Section 4. Finally, Section 5 concludes the paper.

2. RELATED WORKS

Hand detection is commonly performed using skin analysis. There are various methods for skin color detection, which are based on either statistical or algebraic analysis. Algebraic-based methods mainly use known signal properties. In this case, the construction of the signal model is simple, and the model parameters values are only estimated, like histogram-based methods [2].

On the other hand, Gaussian models, mixed Gaussian models, Markov Chain, and Hidden Markov Model are statistical-based methods that are used to model the signal properties as a randomized parametric process. The Hidden Markov Model is powerful in the mathematical structure; therefore, it forms the theoretical foundations of many applications. So, the Hidden Markov Model is always known as one of the most exciting methods for researchers [3, 4].

Color and depth of image are two crucial factors in hand gesture and posture recognition. There are some more compelling works in adapting color descriptors when they are applied to depth data. In DDNN-based (Deep Dynamic Neural Network) methods, the features of skeletal, depth, and color are used in the hand recognition process [3]. In many kinds of researches, cameras with a Kinect depth sensor are used. Depth information is stored with the image by using their thresholds for hand detection [7]. This method is applied to complex backgrounds. Another way is to use optical sensors, because they provide superior tracking and detecting capabilities due to their high resolution [8]. In [9], both features of color and depth with a deep neural network are used. There are many other algorithms for hand segmentation in which some of them use the skin color feature, such as region growing algorithm [9–11]. Also, many different methods and algorithms perform hand segmentation by removing the background [12]. Dondi et al. [11] used a TOF camera (Time of Flight) to collect data. On the other hand, color skin-based methods are particularly important and attractive for the researchers, because these methods are simple and robust against scale changes, movements, and rotations [13].

Many factors prevent accurate segmentation by using these methods, such as complex backgrounds, lighting changes, and poor image quality. To solve these problems, researchers have turned to neural networks [14]. Neural network-based methods have been known to be robust in identifying data patterns, which are superior in speed, flexible against environmental changes, and provide higher performance compared to classic statistical models. On the other hand, neural network-based methods are used to construct educational models; they can model complex situations [15]. Among all the neural networks, convolutional networks provide superior performance. In [16], the sampling window is used to segment the hand from the image; also, a convolutional neural network is used for classification. These networks are known as one of the most important methods of deep learning [17, 18]. Therefore, in this paper, the proposed method is based on convolutional neural networks.

3. PROPOSED METHOD

The aim of this paper is estimation of hand posture in a color image. The overall system consists of four major parts. At first, hand segmentation is performed by using a sampling window, and then the first network results in hand masks. Based on the hand mask, the second network localizes hand key points. Finally, a 3D hand posture from the 2D key points is derived by the third network. Figure 1 shows the general structure of hand posture recognition by the proposed method.

The hand posture estimating framework is described as follows:

3. 1. Hand Segmentation The first step of hand posture recognition is hand tracking and segmentation [3, 19]. The segmentation is the process of finding a connected region in the image with a specific property such as color or intensity, or a relationship between the pixels, that is, a

pattern and the algorithms should be adaptable [4]. To segment the hand, we split the image into the windows. These windows are used to create accurate pixel-wise hand regions [16].

For windowing, the spatial and appearance distributions from the ground truth annotations in the training set are estimated. The sampling window combines spatial biases and appearance models in a unified probabilistic framework. The probability of the existence of a hand is not identical in all parts of the image, so by considering the spatial and apparent models of the hand, a simple method for sampling window is suggested. In this method, the probability of the existence of a hand in a specific area of the image is calculated. This can be formulated as $P(h|a,i)$, where h, a and i represent hand, area, and image, respectively. The steps of the probability method are presented as follows:

Step1: Calculating the occurrence probability of a hand. This probability is estimated directly from the training data and presented as $P(h)$.

Step2: Calculating the probability that is the prior distribution over the position of regions containing a hand, $P(a|h)$. This probability is obtained by using the 4D Gaussian kernel density estimator.

Step3: The third probability is the probability of the existence of a region that contains hand in the desired image, $P(i|a,h)$. This probability is calculated by a simple model. This model estimates the likelihood in which the central pixel of the target area is skin, based on the skin color in RGB color space.

This method allows sampling to be efficiently performed, by calculating the probability of the existence of a hand; and then obtaining the probability of the existence of a hand in a specific area of the image with considering the hand and then sampling it by the kernel density estimator as rectangular sampling windows. Finally, by calculating the probability of the existence of a region that contains hand in the desired image, the kernel weights are adjusted. According to the principle, the probability of the existence of a hand in a particular region of the image is obtained from the multiplication of three probabilities as:

$$P(h|a,i) = P(h).P(a|h).P(i|a,h). \quad (1)$$

$P(h|a,i)$ is calculated for all windows. The maximum likelihood window is considered as a hand area. Figure 2 (b) shows an output of the hand segmentation step.

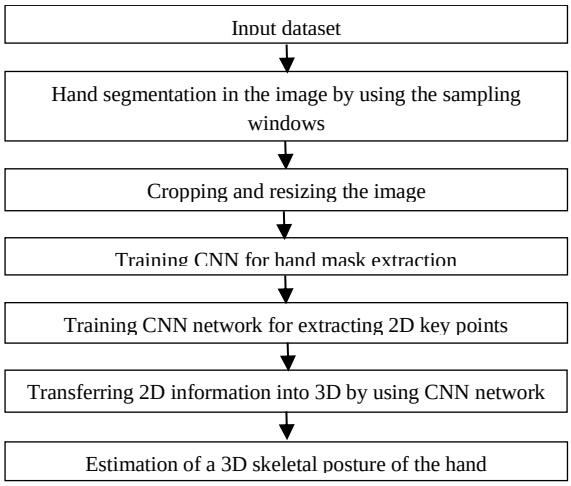

Figure 1. Block diagram of the hand detection process

3. 2. Cropping and Resizing Original images are in size of 320×320 pixels. After segmenting the hand, the separated window crop and resize the original image to 256×256 pixels. It means all of the cropped images are resized to 256×256. The resizing is performed for homogenizing and normalizing the images. These images are ready to apply on the convolutional network to extract the hand mask. Figure 2 (c) shows an example of the cropped image after the segmentation step.

3. 3. Extracting Hand Mask This section includes segmentation and classification. The input image is cropped and served as input to the CNN. The most likely position of the hand, as mentioned in Section 3.1, is determined as the center for the cropped image.

In the proposed method, 33 segmentation masks are defined for a hand including one for part of human hand, one for the palm, one for the background, and three masks for each finger (30 masks for all fingers). The mask of the hand has been presented in Figure 2 (d).

Figure 3 contains the architecture used for the hand mask. The network is initialized using weights of Wei et al. [20] for layers 1 to 16 and then trained for 40000 iterations using a standard Softmax cross-entropy loss. The learning rate is 1×10^{-5} for the first 20000 iterations, 1×10^{-6} for the following 10000 iterations, and 1×10^{-7} until the end.

(a) (b) (c) (d)

Figure 2. (a) An example of an input image. (b) An output of hand segmentation step (c) Cropped and resized hand image (d) A mask extracted for a hand

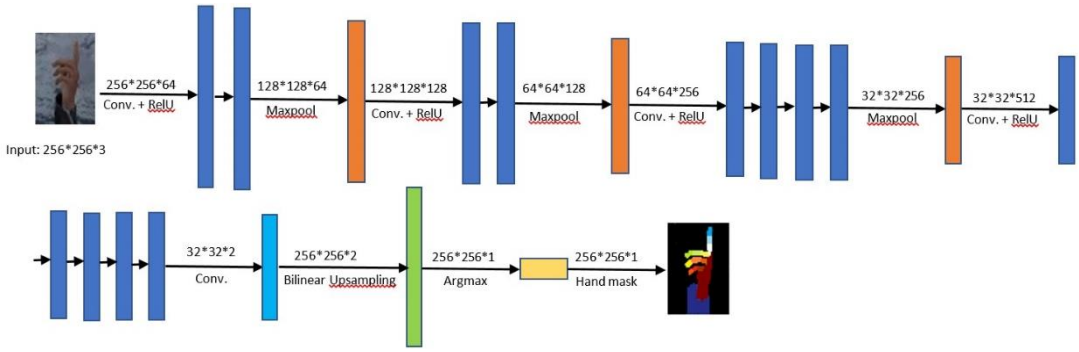

Figure 3. The architecture of mask extracting network

3. 4. Extracting Key Points Consider a color image $I \in R^{N\times M\times 3}$ as an input. After segmenting the hand by using sampling windows, the hand posture in 3D space can be defined as $P_i = (x_i, y_i, z_i)$. This presents the position of key points (J) in 3D space. $i \in [1, J]$ and $J = 21$. This means that 21 key points are considered for each hand to determine the general form of the hand with all its connections so that we can determine all the hand postures as four key points for each finger and one, close to the wrist. The wrist point determines the orientation of the hand. A convolutional neural network is used to extract key points [21]. A 3D structure of the hand is defined by learning the network for estimating normalized coordinates, P_i , based on [6]:

$$P_i^{norm} = \frac{1}{s}.P_i \quad (2)$$

where $s = ||P_{k+1} - P_k||_2$. It is a sample-dependent constant that normalizes the distance between each pair of the key points. *k* is chosen such that s=1 for the first bone of the index finger [6]. Finally, the 3D normalized coordinate is expressed by:

$$P_i^{rel} = P_i^{norm} - P_r^{norm} \quad (3)$$

where *r* is a root index that is equal to zero and P_i^{rel} shows the distance of each point to the wrist.

A primary map of the key points (μ) is predicted by using the features of the image. The points are continuously corrected. Initialization is started by using the weights from Wei et al. [20]. This network is a multi-layer convolutional network with the ReLU activation function, which is used in the continuation of the previous network. This network obtains the map of 2D key points. The values of the key points are mapped to one and zero, where the number one shows the existence of the key point, and zero indicates the absence of it.

Figure 4 contains the architecture used for the key points. The initial 16 layers of the network are initialized using weights of Wei et al. [20]; all others are randomly initialized. The network is trained for 30000 iterations. The learning rate is 1×10^{-4} for the first 10000 iterations, 1×10^{-5} for the following 10000 iterations, and 1×10^{-6} until the end.

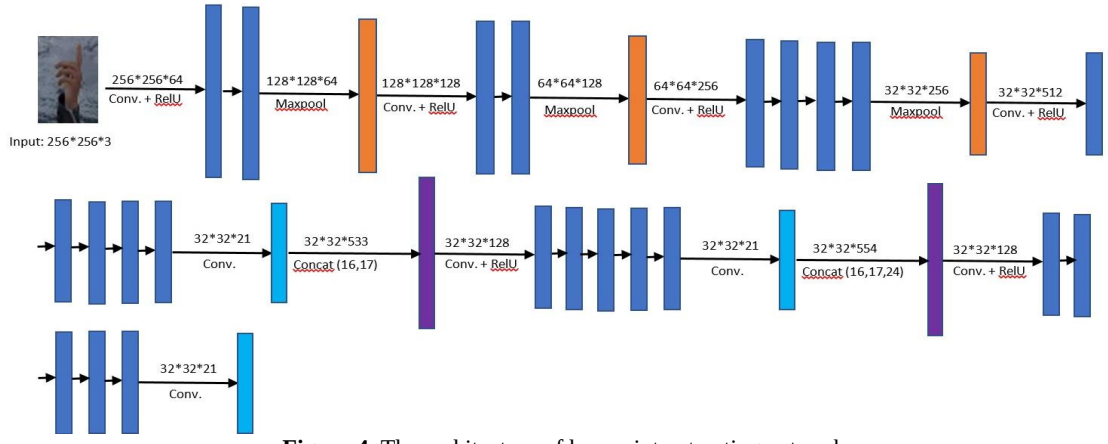

Figure 4. The architecture of key point extracting network

3. 5. Transferring 2D Information to 3D & Estimating 3D Hand Posture We need the 3D information for estimating the form of hand posture in a 3D coordinate. The third network is trained based on anticipated points obtained from the second network. This network displays a more likely 3D output for hand posture, according to the key points map (μ). In this section, the network is trained to convert 2D coordinates into 3D coordinates, instead of predicting the 3D coordinates directly. This network is a two-stream network, which means that its architecture consists of two parallel processing streams.

First, the key points are processed by a 6-layer convolutional neural network. The streams end with a fully connected layer, which is an estimation of central coordinates (p^{μ}) and viewpoint of the system (R matrix), which generally leads to an estimation of the overall hand posture. Consider, p^{μ} is the 3D coordinates of the central frame, which is obtained by using the key point map (μ), and R is a rotation matrix that indicates the predicted viewpoint of a given sample concerning the central frame by estimating its variations. Each of these two factors is obtained by one of the network's streams [6]. Therefore p^{rel} is obtained by:

$$p^{rel} = p^{\mu}.R^{T} \quad (4)$$

where p^{rel} is the normalized 3D coordinates or the 3D form of hand posture.

The error function used in the training process of mask and key points networks is obtained by:

$$L_M = \left\| p_{gt}^{M} - p_{pred}^{M} \right\|_2^2 \quad (5)$$

where p_{gt}^{M} depends on the ground truth factor, and p_{pred}^{M} is the prediction factor of the network [5]. The error function for the third network, which transfers 2D information to 3D is calculated by:

$$L_r = \left\| R_{pred} - R_{gt} \right\|_2^2 \quad (6)$$

This is obtained according to the central transferring matrix, where R_{pred} and R_{gt} are prediction, and ground truth matrix, respectively. The total error function is obtained by the summation of L_M and L_r [6].

Figure 5 contains the architecture used for each stream of the transferring network. All layers use the ReLU activation function. The network ceases with a fully-connected layer that estimates W parameters that is W=3for viewpoint estimation (R matrix), and W=63 for the coordinate estimation stream.

4. EXPERIMENTAL RESULTS

4. 1. Evaluation Criteria To evaluate the performance of the proposed method for estimating the 3D skeletal posture of the hand, we use three evaluations:

- evaluation of hand recognition and segmentation
- evaluation of 2D key-points diagnosis
- evaluation of the performance of hand posture estimator

In these evaluations, four metrics are used as follows:

Accuracy and Medium Accuracy: The accuracy and medium accuracy have been reported by [20] and [22], respectively. Accuracy is a description of systematic errors, a measure of statistical bias; as this causes a difference between a result and a "true" value, the ISO calls this as trueness [23]. The accuracy or trueness of the system performance in percentage is given by:

$$q = \left(100 - \frac{A-B}{B}\right) \times 100 \quad (7)$$

where 'A' is the measured value, and 'B' is the real value.

Medium accuracy is a combination of recall and accuracy, which is used to reduce the information. For each set of information, the accuracy of the recall level is computed from zero to one. It is possible to evaluate accuracy at specified call points. It is formulated by the following equation

$$\bar{q} = \frac{1}{n} \sum_{i=1}^{n} q_i \quad (8)$$

where 'n' is the total number of samples.

Area Under Curve (AUC): The AUC metric has been reported in [6], [16], and [19]. The AUC is an area under the curve of the accuracy of the key points in the range of zero to thirty pixels. It is computed by:

$$S = \int_0^{30} f(x)dx, \quad (9)$$

where f(x) is an aggregation distribution function. The general form of this function is given by:

$$f(x;k,\theta) = x^{k-1} \frac{exp\left(\frac{-x}{\theta}\right)}{\Gamma(k)\theta^{k}} \qquad for\ x > 0\ , k, \theta > 0 \quad (10)$$

where Γ(k) is the Gamma function given by:

$$\Gamma(k) = \int_0^{\infty} y^{k-1}.exp(-y)dy \quad k > 0 \quad (11)$$

Error Rate: Error rate is a powerful metric that is widely used in many articles. For example, according to [24], we can say that it is the same with the mean, standard error that represents the average system error rate. This is obtained by

$$SE = \frac{\delta}{\sqrt{n}} \quad (12)$$

where 'δ' is the standard deviation of a sample, and 'n' is the total number of samples.

4. 2. Datasets To evaluate the proposed method, two datasets have been used. The first one is a very diverse dataset with 43,986 images of 20 animated characters that perform 39 different actions called Mixamo [25, 26]. This dataset has been used to train networks because of its high diversity. Also, to avoid the problem of poor labeling performance by human annotators, utilizing this dataset has been considered. The dataset is split into a validation set and a training set, where a character or action can exclusively be in one set. As a result, the training set contains 32550 images of 13 characters performing 30 actions, and the validation set contains 11436 images of 7 characters with nine actions with a resolution of 320×320 pixels.

The second dataset is RWTH, which is related to the German sign language [27]. It includes 30 hand postures performing by 20 persons. It contains 1160 constant images. This dataset has been used to test the performance of the proposed system in sign language recognition.

4. 3. Evaluations In this section, we present the obtained results by the proposed method and compare it with several existing methods. For this purpose, we evaluate the system in three levels on the Mixamo dataset, and also, the result of an experiment on the RWTH database is expressed.

4. 3. 1. Evaluation of Hand Recognition and Segmentation In Table 1, the accuracy of the sampling window method for hand segmentation is shown. For evaluation, the coverage of the window has been measured. The comparison is performed in two modes, first by considering only the spatial model of hand and second by considering both the spatial and apparent models. The difference between the spatial and apparent models is in the extraction of features. In the spatial model, the features of the spatial position of the hand are estimated, where in the physical model, the features of the hand's appearance in 2D space are taken. As observed, the obtained results are relatively uniform and close to one, even when only the spatial model is considered. This shows the reliability of using sampling windows in segmentation.

4. 3. 2. Evaluation of 2D Key-Points Diagnosis Table 2 shows the performance of the network in the estimation of 2D key points. In this table, two types of evaluations are shown with the percentage of accuracy metric. The second column shows a full performance of estimating 2D key points, i.e., considering the sampling window method for segmentation. On the other hand, the third column shows only the performance of the key points extractor network when the sampling windows are unused. The results of Table 2 are derived from the implementation of the proposed method on the Mixamo dataset at the range of 0 to 30 pixels. As observed, the obtained results show the superior performance of using a sampling window method for segmentation for levels of pixel upper than 10.

4. 3. 3. Evaluation of the Performance of Hand Posture Estimator In Table 3, the performance of the proposed method in comparison with several methods is described based on AUC, error rate, and average accuracy.

TABLE 1. Checking the accuracy of the sampling window

Number of windows	Spatial model	Spatial & apparent model
500	0.82	0.88
1000	0.90	0.93
1500	0.92	0.94
2000	0.94	0.95
2500	0.94	0.95

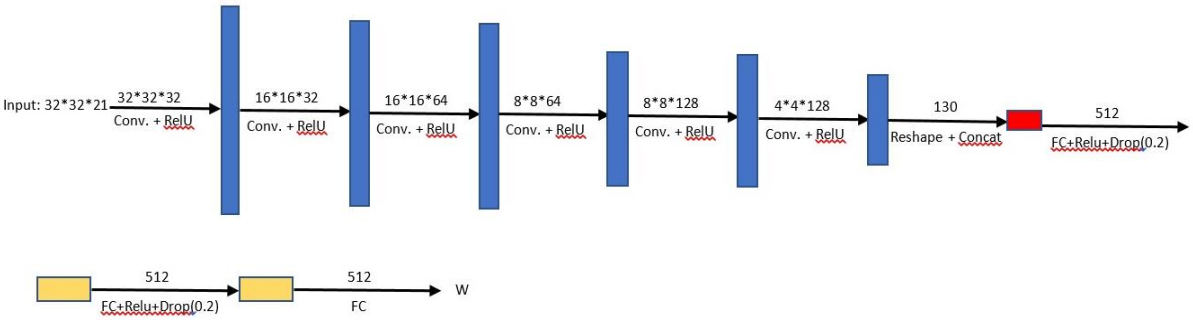

Figure 5. The network architecture of a single stream for the third CNN

According to the obtained results, the proposed method, in all cases, is better than the DDNN method. The proposed method is even better than the method reported in [11] because of using CNNs. The reported method in [16] used only one CNN. The results show that the proposed method provides higher performance by using three DCNNs. In [9], the diagnostic function has been tested separately for both deep-belief and DCNNs. The obtained results show the superiority of the proposed system in both of these modes. The results of the probabilistic method [24] in Table 3 represent a poor performance. The statistical-based models and methods provide high performance, but it is important how to use these methods. In the proposed method, we used a probabilistic method for hand segmentation followed by deep neural networks, which results in an appropriate combination for hand posture recognition. The 3DCNN method (Deep 3D Convolutional Neural Network), provides lower performance than the proposed method, which used three deep convolutional neural networks separately.

TABLE 2. Evaluation of the maximum accuracy of key points in both modes of using the hand segmentation system and not using it

Level of pixels	Performance of estimating 2D key points	Performance of key points extractor network
0-5	0.19	0.18
5-10	0.33	0.32
10-15	0.48	0.55
15-20	0.57	0.65
20-25	0.61	0.72
25-30	0.62	0.73

TABLE 3. Comparison of the proposed method with other methods in hand posture recognition

Method	AUC	Error Rate	Medium Accuracy
DDNN-Fusion [3]	0.865	0.149	0.879
TOF [12]	0.667	0.318	0.685
[17]	0.842	0.159	0.807
CNN [10]	0.882	0.207	0.891
DBN [10]	0.902	0.143	0.929
Probability [24]	0.660	0.327	0.684
3D-CNN [26]	0.723	0.168	0.758
The Proposed Method	**0.948**	**0.147**	**0.950**

For an experiment, the proposed method was used for sign language recognition. For this aim, the German finger spelling dataset called RWTH was used. Additionally, a fully connected three-layer classifier with the ReLU activation function was used. Finally, the word error rate in percent on the RWTH German finger spelling dataset resulted in 15%.

5. CONCLUSIONS

In this paper, we tried to estimate the skeletal hand posture by using DCNNs. To improve the hand segmentation and recognition process, we proposed a hand-method based on an apparent and spatial model of the hand in the sampling window. A 16-layer DCNN with activation function ReLU was used to extract hand masks. To extract the key points, the information of hand masks was used, and a map of key points in 2D space was extracted by using another DCNN. This information was finally transferred to 3D space by using the third CNN, to estimate 3D skeletal hand posture. The obtained results showed that the proposed method has a superior performance in hand posture recognition by using three DCNNs. The efficiency of this system is not merely limited to the detection of sign languages, it can also be used in different applications.

6. REFERENCES

1. Hosseini, S.M., Nasrabadi, A., Nouri, P. and Farsi, H., "A novel human gait recognition system", ***International Journal of Computer and Electrical Engineering***, Vol. 2, No. 6, (2010), 1043–1049.
2. Ohn-Bar, E. and Trivedi, M. M., "Hand gesture recognition in real time for automotive interfaces: A multimodal vision-based approach and evaluations", ***IEEE transactions on intelligent transportation Systems***, Vol. 15, No. 6, (2014), 2368–2377.
3. Wu, D., Pigou, L., Kindermans, P.J., Le, N.D.H., Shao, L., Dambre, J. and Odobez, J. M., "Deep dynamic neural networks for multimodal gesture segmentation and recognition", ***IEEE Transactions on Pattern Analysis and Machine Intelligence***, Vol. 38, No. 8, (2016), 1583–1597.
4. Pradipa, R. and Kavitha, S., "Hand gesture recognition–analysis of various techniques, methods and their algorithms", ***International Journal of Innovative Research in Science, Engineering and Technology***, Vol. 3, No. 3, (2014), 2003–2010.
5. Dinh, D.L., Kim, J.T. and Kim, T. S., "Hand gesture recognition and interface via a depth imaging sensor for smart home appliances", ***Energy Procedia***, Vol. 62, No. 62, (2014), 576–582.
6. Zimmermann, C. and Brox, T., "Learning to estimate 3d hand pose from single rgb images", In Proceedings of the IEEE International Conference on Computer Vision, IEEE, (2017), 4903–4911.

7. Cardoso, T., Delgado, J. and Barata, J., "Hand gesture recognition towards enhancing accessibility", ***Procedia Computer Science***, Vol. 67, (2015), 419–429.
8. Sharma, R.P. and Verma, G. K., "Human computer interaction using hand gesture", ***Procedia Computer Science***, Vol. 54, (2015), 721–727.
9. Tang, A., Lu, K., Wang, Y., Huang, J. and Li, H., "A real-time hand posture recognition system using deep neural networks", ***ACM Transactions on Intelligent Systems and Technology***, Vol. 6, No. 2, (2015), 1–23.
10. Farsi, H. and Mohamadzadeh, S., "Combining Hadamard matrix, discrete wavelet transform and DCT features based on PCA and KNN for image retrieval", ***Majlesi Journal of Electrical Engineering***, Vol. 7, No. 1, (2013), 9–15.
11. Dondi, P., Lombardi, L. and Porta, M., "Development of gesture-based human–computer interaction applications by fusion of depth and colour video streams", ***IET Computer Vision***, Vol. 8, No. 6, (2014), 568–578.
12. Husain, F., Gandhi, S., Nijhawan, T., Agarwal, V., Khatun, S. and Parveen, S., "Gesture Recognition System Using Matlab: Aliterature Review", ***International Journal of Scientific Research and Management Studies***, Vol. 2, No. 11, (2016), 425–432.
13. Khan, R.Z. and Ibraheem, N. A., "Hand gesture recognition: a literature review", ***International Journal of Artificial Intelligence & Applications***, Vol. 3, No. 4, (2012), 161–174.
14. Hosseini, S.M., Farsi, H. and Yazdi, H. S., "Best clustering around the color images", ***International Journal of Computer and Electrical Engineering***, Vol. 1, No. 1, (2009), 20–25.
15. Afifi, M., "11K Hands: gender recognition and biometric identification using a large dataset of hand images", ***Multimedia Tools and Applications***, Vol. 78, No. 15, (2019), 20835–20854.
16. Bambach, S., Lee, S., Crandall, D.J. and Yu, C., "Lending a hand: Detecting hands and recognizing activities in complex egocentric interactions", In Proceedings of the IEEE International Conference on Computer Vision, IEEE, (2015), 1949–1957.
17. Sezavar, A., Farsi, H. and Mohamadzadeh, S., "Content-based image retrieval by combining convolutional neural networks and sparse representation", ***Multimedia Tools and Applications***, Vol. 78, No. 15, (2019), 20895–20912.
18. John, V., Umetsu, M., Boyali, A., Mita, S., Imanishi, M., Sanma, N. and Shibata, S., "Real-time hand posture and gesture-based touchless automotive user interface using deep learning", In 2017 IEEE Intelligent Vehicles Symposium (IV), IEEE, (2017), 869–874.
19. Farsi, H., "Improvement of minimum tracking in minimum statistics noise estimation method", ***Signal Processing: An International Journal***, Vol. 4, No. 1, (2010), 17–23.
20. Wei, S.E., Ramakrishna, V., Kanade, T. and Sheikh, Y., "Convolutional pose machines", In Proceedings of the IEEE conference on Computer Vision and Pattern Recognition, IEEE, (2016), 4724–4732.
21. Zhao, R., Wang, Y. and Martinez, A. M., "A simple, fast and highly-accurate algorithm to recover 3d shape from 2d landmarks on a single image", ***IEEE Transactions on Pattern Analysis and Machine Intelligence***, Vol. 40, No. 12, (2017), 3059–3066.
22. Lee, S., Bambach, S., Crandall, D.J., Franchak, J.M. and Yu, C., "This hand is my hand: A probabilistic approach to hand disambiguation in egocentric video", In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops, (2014), 543–550.
23. Mohammadzadeh, S. and Farsi, H., "Image retrieval using color-texture features extracted from Gabor-Walsh wavelet pyramid", ***Journal of Information Systems and Telecommunication***, Vol. 2, No. 15, (2014), 31–40.
24. Farsi, H., Mozaffarian, M.A. and Rahmani, H., "Improving voice activity detection used in ITU-T G. 729. B", In Proceedings of the 3rd WSEAS International Conference on Circuits, Systems, Signal and Telecommunications, Ningbo, China, (2009), 11–15.
25. Molchanov, P., Gupta, S., Kim, K. and Kautz, J., "Hand gesture recognition with 3D convolutional neural networks", In Proceedings of the IEEE conference on computer vision and pattern recognition workshops, (2015), 1–7.
26. Mixamo database, Available at: http://www.mixamo.com.
27. RWTH German finger spelling database, Available at: http://wwwi6.informatik.rwthaachen.de/~dreuw/fingerspelling.php.

Persian Abstract

چکیده

تخمین و تشخیص حالت دست به خاطر کاربردهای فراوانش مورد توجه محققان زیادی قرار گرفته است. سیستم‌های تشخیص حالت دست سیستم‌هایی هستند که با به کارگیری الگوریتم‌های ریاضی حالت دست را شبیه‌سازی می‌کنند. تا به امروز شبکه‌های کانولوشن در این زمینه نتایج خوبی از خود نشان داده‌اند. در این مقاله با استفاده از شبکه عصبی کانولوشن عمیق فرم اسکلتی دست را تخمین می‌زنیم. با هدف کاربردی‌تر کردن روش و حفظ سادگی آن از فاکتور عمق صرف نظر کردیم و تنها ورودی استفاده شده برای سیستم تصاویر ساده رنگی دست هستند. مجموعه داده استفاده شده یک مجموعه داده با تنوع بالا، شامل ٤٣٩٨٦ تصویر رنگی است که حدود ٧٤٪ آن برای مجموعه آموزش و ٢٦٪ آن به عنوان مجموعه ارزیابی انتخاب شدند.