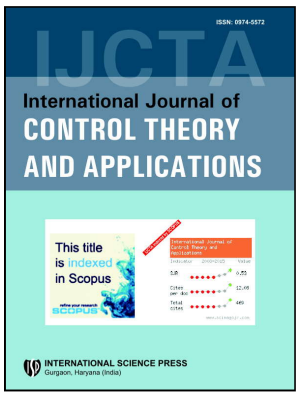


International Journal of Control Theory and Applications

ISSN : 0974-5572

Volume 10 • Number 19 • 2017

Analysis of computer vision based hand gesticulation using deep learning

Sagaya Mary J.[1] and Nachamai M.[1]

[1] *Department of Computer Science Christ University, Bangalore, Karnataka, India,*
Emails: sagaya.mary@res.christuniversity.in, nachamai.m@christuniversity.in

Abstract: In the contemporary situation, gesticulation recognition is an elementary yet exigent research topic. This non-verbal deed unites all level of people and deprived in particular. This technical survey sums up the pertinent work, much of it spotlight from the recent literature. The depths of the diverse techniques of deep learning are evaluated in different aspects, which prove to be useful for the researchers, whose research is based on the blend of gestures and deep learning. This paper encompasses the following divisions, the relationship of deep learning and computer vision, vision-based gesture in the facets of deep learning and depth analysis of deep learning in gesture and action recognition.

Keywords: Autoencoder, Convolution Neural Networks, Restricted Boltzmann Machine, Cubic Kernel

1. INTRODUCTION

In recent times, the research on hand gesticulation is employed in diverse fields and the machines are improvised to classify captured image and segregate it into desired categories. Researcher Karam (2006) has proved that hand is the mostly used part of the body to signal gestures (to express their thoughts, innate feelings, and notifications) since it is the intrinsic mode of communication connecting individual to individual and also individual to computers. The acquired gestures can be of two prime types - static and dynamic. Only spatial images are captured in static. In the dynamic, both temporal and spatial images are captured but become complex in its development than static.

1.1. Hand Gesture Recognition

In the aspect of human-computer interaction, the hand gestures can be recognized in two sorts namely contact based and vision based devices. Additional devices like data glove, accelerometers, and multi-touch screen are combined with physical interaction of user in contact based method. This work focuses on the vision-based method, which has the high-level understanding of image analysis. In order to avoid superfluous hardware gadgets, the above method is more opted as it directly deals with digital images or videos. Vision-based devices are further split into two major categories: 3D model based and appearance based methods [1].

1.2. Feature Learning in Computer Vision

The computer vision has three major processes – Feature Description, Dimensionality Reduction, and Classification. Though there are many techniques applicable in computer vision, limitations are higher to implement the process. Hence compared to handcrafted techniques, deep learning is much accessible because of its unique features greedy layer-wise unsupervised pre-training, automatic feature extraction and cojoint classification.

1.3. Deep Learning and Computer Vision

Deep learning is a subfield of machine learning that prompts to obtain top-notch understanding by utilizing stacked architectures [2]. As stated earlier, computer vision tasks involve the methods for capturing, implementing, exploring and interpreting digital images, and then extract the high-dimensional data from the real world conducive to construct numerical or symbolic information [3]. Deep networks are proved to be successful in computer vision as they have the capability of combining all the processes from the feature extraction to classification.

1.4. Computer Vision in Gesture Recognition

Real time hand gesture based computer vision has led to various researches in growing years with stirring applications such as Human-Robot Interaction(HRI), sign language interpretation, computer games control, virtual reality and assistive environments[4]. Technically, the mechanism of artificial systems is inbuilt in computer vision and this extracts even complex information from images in forms of continuous videos, calibrated views from multiple cameras, multi-dimensional data [5] and even miniscule information.

1.5. Deep Learning in Hand Gesture Recognition

Recent development in deep learning approaches has proved the performance of this state-of-the-art visual recognition system. This paper is an introspection of the various types of deep learning techniques which prove the expected enhancement in the application of gestures. Various deep learning neural network architectures for vision are Auto-encoders, Restricted Boltzmann Machines (RBM), and Convolution Neural Networks (CNN) [6].

2. ANALYSIS OF VARIOUS DEEP LEARNING TECHNIQUES

2.1. Autoencoder

The autoencoder is an outstanding nature of artificial neural network which challenges to learn proficient encoding. It uses an unsupervised learning algorithm that replicates back propagation, yielding the anticipated values to be identical to the inputs. (i.e), it uses $b^{(n)} = a^{(n)}$. The simple learning circuits of autoencoder transform inputs into outputs with minimal alterations. This minimal alteration optimizes efficient learning in autoencoder by minimizing the reconstruction error. The single layer architecture does not provide the high level understanding of features from original data. To overcome the difficulty and to secure the autoencoder, better techniques (denoising, sparse, variational and contractive) are required to learn the identity functions, to capture the important information and the affluent representations [7].

2.2. Denoising Autoencoder

The autoencoder is trained to restructure the input from a corrupted version rather forcing the hidden layer to grasp further interesting features and to secure it from merely learning the identity [8].

2.3. Stacked Autoencoder

As further transformations on the non-linear transformed data render complex information, here, autoencoders are stacked on top of the each other i.e. the next autoencoder is trained using the hidden layer activations of the

previous autoencoder and the depth is considered on the basis of the number of autoencoders stacked in the neural network.

2.4. Stacked Denoising Autoencoder

By stacking more hidden layers, more dispersed and stratified learning of feature is accomplished in [9]. The denoising network reconstructs tainted input data in the output layer [10]. By combining the above properties, the system can learn more interesting features in a linear manner.

2.5. Stacked Denoising Autoencoder in Static Hand Gesture Recognition

Stacked denoising autoencoders (SDAEs) of various measures (stacking many layers) are trained for the sake of observing few enhancements in performance. The three distinctive SDAEs are spelled as SDAE1, SDAE2 and SDAE3.

2.6. Sparse Autoencoder

The autoencoder has a trivial purpose of learning the identity function, (i.e) with similarities. To discover further appealing structures, the network demands less hidden layers. This limitation is overcome by implementing sparsity in autoencoder where the appealing structures are acquired even when the number layers are large [11].

2.7. Stacked Denoising Sparse Autoencoders [SDSA] in Static Hand Gesture Recognition

In this research the properties of denoising sparse and stacked are combined to observe the learning efficiency. This SDSA system is applied in American sign gesture recognition, where the unlabelled data of hand images are stacked on top of each other and it is also extended to solve the problem of rotation and Gaussian noise.

2.7.1. Result Analysis (Comparison of SDAE with SDSA)

The performance of different depths SDAE (with different neurons) and SDSA (with same neurons) is compared. When the neurons of the first hidden layers of different SDAEs are compared, the learned kernels of first hidden layer in SDAE1are found active, more active in SDAE2 and most active in SDAE3. In SDSA, accuracy rate is increased when the depth is increased with fixed neurons. It is observed that, as depth increases (hidden layers), more interesting representations are learnt in both.

2.8. Convolution Neural Networks (CNN)

CNNs, multi-layered Neural Networks (NN) recognize visual patterns directly from image pixels. Unlike traditional NN, several layers of convolutions with non-linear activation function are used over the input layer to compute the required output in CNN [12]. Hyperbolic tangent, logistic function, Rectified Linear Unit (ReLU), Leaky Rectified Linear Unit (LReLU), soft plus function are non-linear activation functions, of which sigmoidal hyperbolic tangent is proved to provide more robust performance with Deep Neural Networks [13]. The entire architecture of CNN comprises three layers: convolution, sub-sampling, and fully connected.

2.8.1. Convolution layer

The convolution layer, the heart of a CNN carries out most of the complex computational task [14] extracting the features from the raw input image. Figure 1 shows the process of feature extraction of a stacked convolution layer.

This layer performs two important functions namely filtering and rectification [15] to preserve the spatial correlation between pixels [16]. The working principles are explained in figure 2. In filtering, the 3x3 matrix

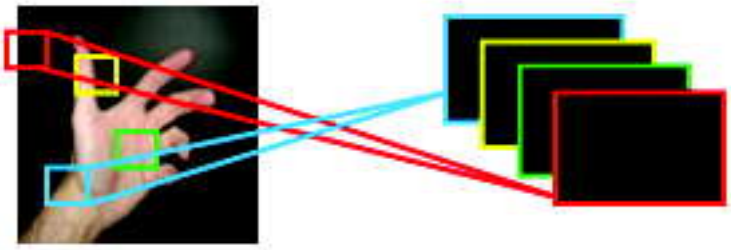
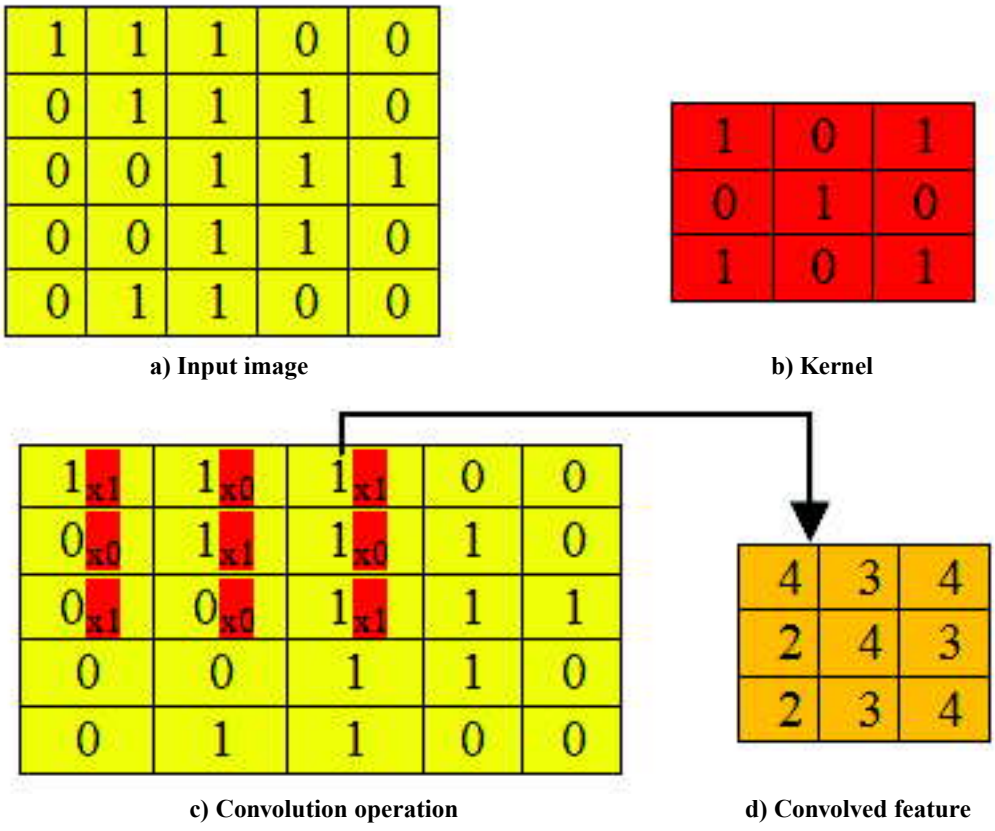


Figure 1: Feature extraction in convolution layer

called 'filter' or 'kernel' or 'feature detector' which is shown in figure 2b is scanned over the input image portrayed in figure 2a (0s and 1s). The product of input image and kernel produces the 'convolution feature' or 'activation feature map' or the 'feature map' shown in 2c. The sum value of the convolution feature (1x1+1x0+1x1+0x0+1x1+1x0+0x1+0x0+1x1=4….etc) form the convolved features plotted in 2d. In rectification, the non-linearity transforms negative to positive results obtaining similarities.



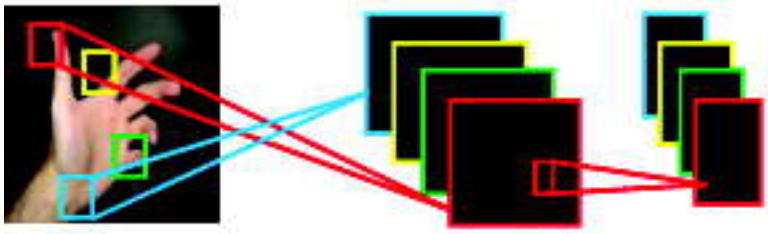
a) Input image **b) Kernel**

c) Convolution operation **d) Convolved feature**

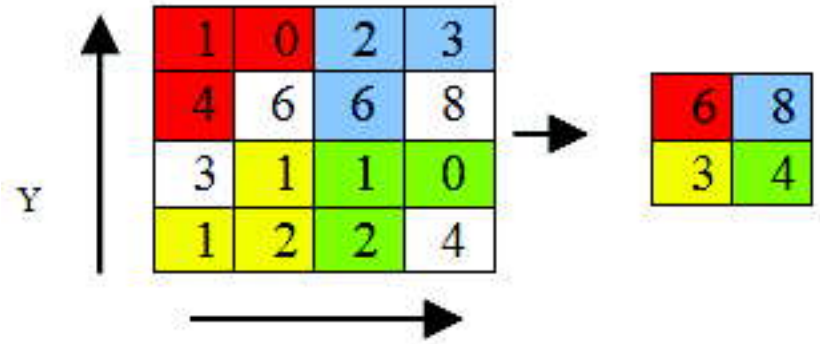
Figure 2:

2.8.2. Pooling / Sub-Sampling Layer and Fully Connected Layer

The information yielded from the convolution layer is the input to the next layer of CNN called pooling, a form of non-linear down-sampling. Pooling can be of distinct types: max, average, sum etc among which max pooling is the most common [17]. It segregates the input image into a set of non-overlying rectangles, picks the highest from every sub-region the white portion shown in figure 3b (6 from first sub-region, 8 from the next sub-region and 3 and 4 from the next successive sub-regions) and reduces the spatial size of the representation that leads to minimum amount of parameters, least computation and over-fitting. In spite of extraction, the pooling operation



a: Stacked convolution and sub-sampling layers



b: Extracting Maximum value from non-overlapping rectangles.

Figure 3:

sustains the property of translation invariance. Pooling layer can be inserted in-between successive convolution layers in CNN architecture. The convolution layer is extended with pooling layer shown in Figure 3a.

The Fully Connected layer is a conventional Multi Layer Perceptron that employs a softmax activation function in the output layer. The intention of the fully connected layer is to classify high-level features of the original image into multiple modules extracted from the convolutional and pooling layers.

2.9. Review of the Depth CNN

In further review, the depth of CNN is implemented in various forms like size, multi-channel, cubic kernel, max-pooling and learning multi-viewed features in the application of hand gesture recognition using 3D-CNN.

2.9.1. Depth CNN

CNN with different depth size is trained. $L \times M$ size of input image and a kernel size of $k \times k$ are utilized for the convolution operation. f convolution feature maps are extracted into size $c \times c$ each. The same kernel size is again used in the sub-sampling layer to attain the next convolution layer with e feature maps of size $g \times g$ each. Extracted features in every convolution feature map are structurally curtailed. Feature maps in all Convolution layers are reduced using the same mask size of $j \times j$.

2.9.2. Result Analysis

CNN1 with two hidden layers obtained the best recognition rate, but when the depth increases (CNN2, CNN3) recognition rate is decreased and the run time is increased which leads to saturation in neurons and vanishing gradients.

2.10. Multi-channel CNN

Multi-channel Convolutional Neural Network (MCNN), recognizes the hand postures by deploying novel architecture called cubic kernel concept and a multi-channel flow of information; it allows recognizing images

even if they have a small size. The model uses a cubic kernel to enhance the features for the classification and multi-channel architecture is implemented for tuning the filters based on the Sobel operators [18].

2.10.1 Result Analysis

The result is analyzed in the aspects of comparison of layers, kernels with different databases and different sizes (28x28 and 128x128). To evaluate the three channels, the author had compared with the utilization of one-channel, two-channel and three-channel architecture and 2D kernels are compared with cubic kernels. The overall result shows that the combination of multi-channel and cubic kernel has produced the effective result.

2.11. Max-Pooling Convolution Neural Network (MPCNN)

Big and deep NNs are focused in the aspect of merging convolution and max-pooling for supervised feature learning. Convolutional layer is parameterized by: the number of maps *(F)*, the size of the maps*(Fx,Fy)* and kernel sizes*(Kx,Ky)*.Max pooling generates position invariance, dimensionality reduction and improves the performance.

2.11.1 Result Analysis

The result of the MPCNN is compared with the Support Vector Machines (SVM) classifier approach which is suitable for vision based gesture and object recognition. This shows that the error rate (27.04%) of hand crafted technique is gradually reduced and a vast difference could be made when the automatic feature learning technique (3.23%) is implemented.

2.12. Multi-View 3D-CNN (MVCNN)

MVCNN enclose three essential divisions. The primary part involves the MV3D-CNN where autonomous 3D-CNNs are used to extract the features from videos acquired at different view cameras. The second part includes the view pooling layer which can cumulate all vision descriptors and train view relevant features. The final recognition and classification process is performed by the third part called Fully Connected Neural Networks (FNN) with softmax. Kernels of 3D-CNN can learn complicated spatiotemporal features [19].

2.12.1 3D Convolution Layer and Pooling Layer

The major role of convolution layer is to extract spatiotemporal features since it has the ability to convert temporal data from video. To enhance the recognition accuracy view pooling layer is used. Three kinds of pooling namely Multi-View Pooling (MVP), Average View Pooling (AVP) and Weighted Average View Pooling (WAVP) are implemented among which WAVP produced good upshot.

2.12.2. Result Analysis

The comparison of CNN with 3D-CNN is analyzed in two view related information such as side and front view. It is obvious that the recognition accuracy and acquisition of useful information from the front view (83.85 in CNN, 87.45 in 3D-CNN) is superior performance than that of the side view (81.57 in CNN, 86.92 in 3D-CNN). Evaluating with three view-pooling methods, the best performance is attained from WAVP layer (MVP-92.93, AVP-89.44, WAVP-93.52) in the MV3D-CNN model.

2.13. Restricted Boltzmann Machine and Multiple Restricted Boltzmann Machines (MRBMs)

A Restricted Boltzmann Machine (RBM) is a generative stochastic neural network, and was anticipated by Hinton et al. in 1986. An RBM is alternate of the Boltzmann Machine, with the limitation that the visible units and hidden

units form a bipartite graph where the inputs from all visible nodes are sent to all hidden nodes [20]. Spatiotemporal features are learned by using a novel approach called multiple RBMs in action recognition. Initially, preprocessing scans the action paths from the action videos. Later, the scanned tracks are split into block sequence and shape features are extracted instead of pixels. These extracted shape features are trained using MRBMs [21].

2.13.1. The Architecture of MRBMs

It has two layers: the first layer is a multiple RBM, which describes the allocation of the features of the action class in block shape. The spatial location of block shape features are the input to the RBM. Weight is signified in the form of symmetric interaction matrix among input and output. The intention of the second layer is to diminish the dimensionality of the output of the first layer. The same mode of parameters represented in the first layer is followed in the second layer. It is observed that the action videos of the same class have given the same output, whereas the output is different for other class of action.

2.13.2 Result Analysis

The result of the Keck gesture dataset is reviewed, though some features are similar, it is quite a complex task to distinguish them. MRBMs attain higher accuracy than 3DSIFT feature and acquire a significant improvement in mean accuracy when compared with (Bag of Features) BOF algorithm.

2.14. Local Interaction RBMs (LRBMs)

The researcher in this task paid attention on the RBMs by introducing model learning (capturing spatial and temporal patterns from continuous video) and multi-class classification (pair wise classifier). Normally RBM architecture is of two layers: visible and hidden layer. Data is represented in visible layer and feature extraction in hidden layer. The interaction between visible and hidden layer is established through the weight parameter. In contrast, Gaussian-Binary RBM is deployed, where the binary hidden units and the normal distribution are followed [22].

2.14.1 Model Learning and Multi-Class Classification

To overcome the limitations of direct connection between visible and hidden layers and the similarity information in spatial and temporal features, Local interaction RBM (LRBM) is introduced. The parameters of the LRBMs are bias for visible and hidden layer (*xi* and *yj)*, weight between two layers (*w)* and local interaction matrix (*V)*. Hidden units perform parallel sampling as it is independent, whereas the visible layers could not because of its dependent nature. To reconstruct the visible layer, sampling is done based on mean field algorithm. Normally RBM uses generative model for training different classes with multiple models. As the partition of this classification cannot be done with the largest number of hidden and visible units, annealed importance sampling is employed which requires more sampling for good estimation. This is extended with multi-class classification but the partitions function is computed with label ranking instead of computing directly, which leads to high score.

2.14.2. Result Analysis

In the analysis of G3W action dataset with gaming actions, the dimensionality must be reduced by using the 3D location information of four dominant joints since it is skeletal data. Normalization is performed to acquire the similarity in subject and body shapes. The overall accuracy achieved was 90.5%.

3. DISCUSSION

This study found that the deep learning and vision based techniques are suitable for analyzing hand gesture, since deep learning has the capability of automatic feature extraction and stacked architecture. Deep networks are proved to be successful in computer vision as they have the ability of combining all the processes from the

feature extraction to classification. The current available three architectures of deep learning namely autoencoder, CNN and RBM work well when the depth is increased. It is proved that the maximum recognition accuracy and less run time are achieved when the depth is increased.

4. CONCLUSION

Over the earlier periods the gesticulation employed for interrelating with technical devices has sustained to be a booming field of research. This paper has conferred a task-based framework for gesticulation recognition in the context of deep learning. The detailed mechanism of deep learning in diverse aspects provides evidence for implementing in non verbal deeds. Moreover, the drastic improvements in running time and in recognition accuracy encourage the upcoming researchers to broaden their attention.

REFERENCES

[1] D. Nithin and P. Sivakumar, "Generic Feature Learning in Computer Vision", Procedia Computer Science, vol. 58, pp. 202-209, 2015.

[2] Y. Guo, Y. Liu, A. Oerlemans, S. Lao, S. Wu and M. Lew, "Deep learning for visual understanding: A review", Neurocomputing, vol. 187, pp. 27-48, 2016.

[3] V. Kumar, G. Nandi and R. Kala, "Static hand gesture recognition using stacked Denoising Sparse Autoencoders", 2014 Seventh International Conference on Contemporary Computing (IC3), 2014.

[4] Jawad Nagi. "Max-pooling convolutional neural networks for vision-based hand gesture recognition", 2011 IEEE International Conference on Signal and Image Processing Applications (ICSIPA), 11/2011.

[5] "Computer vision", En.wikipedia.org, 2016. [Online]. Available: https://en.wikipedia.org/wiki/Computer_vision. [Accessed: 30-Nov-2016].

[6] S.Rautaray and A.Agarwal ,"Vision based hand gesture recognition for human computer interaction: a survey", Artificial Intelligence Review,vol.43, no.1, pp1-54,2012.

[7] "Autoencoder", En.wikipedia.org, 2016. [Online]. Available: https://en.wikipedia.org/wiki/Autoencoder. [Accessed: 26- Nov- 2016].

[8] "DenoisingAutoencoders(dA)—DeepLearning 0.1 documentation", Deeplearning.net, 2016. [Online]. Available:http://deeplearning.net/tutorial/dA.html. [Accessed: 26- Nov- 2016].

[9] P. Vincent et.al, "Stacked Denoising Autoencoders: Learning Useful Representations in a Deep Network with a Local Denoising Criterion", Journal of Machine Learning Research, vol. 11, pp. 3371-3408, 2010.

[10] O. Oyedotun and A. Khashman, "Deep learning in vision-based static hand gesture recognition", Neural Comput & Applic, 2016.

[11] Ng, Andrew. "Sparse autoencoder." CS294A Lecture notes 72 (2011): 1-19.

[12] "Understanding Convolutional Neural Networks for NLP", WildML, 2016. [Online]. Available: http://www.wildml.com/2015/11/understanding-convolutional-neural-networks-for-nlp/. [Accessed: 26- Nov-

2016].

[13] C. Fannjiang, and M. Fang,"Nonlinear activations for convolutional neural network acoustic models", (2016).

[14] "CS231n Convolutional Neural Networks for Visual Recognition", Cs231n.github.io, 2016. [Online]. Available: http://cs231n.github.io/convolutional-networks/. [Accessed: 26- Nov- 2016].

[15] K.Jarrett, K. Kavukcuoglu, and Y. Lecun, (2009, September). "What is the best multi-stage architecture for object recognition?", In 2000E 12th International Conference on Computer Vision , pp. 2146-2153. IEEE.

[16] "An Intuitive Explanation of Convolutional Neural Networks", the data science blog, 2016. [Online]. Available: https://ujjwalkarn.me/2016/08/11/intuitive-explanation-convnets/. [Accessed: 30- Nov- 2016].

[17] "Convolutional neural network", En.wikipedia.org, 2016. [Online]. Available: https://en.wikipedia.org/wiki/Convolutional_neural_network.[Accessed: 26- Nov- 2016].

[18] P. Barros, S. Magg, C. Webe and S. Wermter, "A multichannel convolutional neural network for hand posture recognition", International Conference on Artificial Neural Networks, pp. 403-410,2014.

[19] T. He, H. Mao and Z. Yi, "Moving object recognition using multi-view three-dimensional convolutional neural networks", Neural Comput & Applic, 2016.

[20] A. Chris Nicholson, "A Beginner's Tutorial for Restricted Boltzmann Machines - Deeplearning4j: Open-source, Distributed Deep Learning for the JVM", Deeplearning4j.org, 2016. [Online]. Available https://deeplearning4j.org/restrictedboltzmannmachine.[Accessed: 30- Nov- 2016].

[21] L. Pei, M.Ye, X. Zhao, T. Xiang, and T. Li, "Learning spatio-temporal features for action recognition from the side of the video" Signal, Image and Video Processing, vol.10,no.1, pp.199-206,2016.

[22] S.Nie, S, Z.Wang, and Q.Ji,"A generative restricted Boltzmann machine based method for high-dimensional motion data modeling.Computer Vision and Image Understanding", vol.136, pp.14-22, 2015.