Multihead Self Attention Hand Pose Estimation

Zhiqin Zhang[1], Bo Zhang[1], Fen Li[2,*], Dehua Kong[1]

[1]School of Computer Science, Wuhan Donghu University, Wuhan, China
[2]School of Computer Science, Wuhan Qingchuan University, Wuhan, China

Abstract. In This paper, we propose a hand pose estimation neural networks architecture named MSAHP which can improve PCK (percentage correct keypoints) greatly by fusing self-attention module in CNN (Convolutional Neural Networks). The proposed network is based on a ResNet (Residual Neural Network) backbone and concatenate discriminative features through multiple different scale feature maps, then multiple head self-attention module was used to focus on the salient feature map area. In recent years, self-attention mechanism was applicated widely in NLP and speech recognition, which can improve greatly key metrics. But in compute vision especially for hand pose estimation, we did not find the application. Experiments on hand pose estimation dataset demonstrate the improved PCK of our MSAHP than the existing state-of-the-art hand pose estimation methods. Specifically, the proposed method can achieve 93.68% PCK score on our mixed test dataset.

1 Introduction

The hand is the most important part for human and we can use our hands operate on many actions as a tool. We can use our hands as an input device for human-computer interaction, and hand estimation or hand key-points detection can be used in many fields, such as, object handover in robotics, learning from demonstration, sign language and gesture recognition. But, we must get the accurate hand estimation information: its key-points location, palm orientation and complex articulation in space etc. Most current applications rely on the mixed data including depth image from a depth camera and color image from a CCD camera. Unfortunately, depth cameras are commonly unavailable in contrast to regular color CCD cameras, and they can only work reliably in restricted conditions, for instance indoor environments. It is a confused problem for hand pose estimation from single images because of complex articulation, few context discrimination and serious finger self-occlusion, it is even more difficult to estimate accurately hand pose than for the overall human body. Therefore, in many applications, some specific sensing equipment are used, such as data gloves and markers, which restrict the wide application in different fields.

In recent years, attention module show its advantages on traditional CNN or RNN(recurrent neural network) module, self-attention is a kind of attention [1] that generates a sequence and every element in the sequence is a weighted average of the rest of the sequence. RNN like structure will be used for NLP in early results on transfer learning [2], but a kind of special multi-head self-attention architecture has been used more common recently which was named the "Transformer" architecture [3]. A lot of huge advantages has been shown by "Transformer" architecture which achieved better results than SOTA RNN or CNN like methods on a wide variety of NLP settings. Although "transformer" is better than RNN, it is rarely to see relative architecture used in computer vision especially for hand estimation. In this paper, we proposed the "transformer" like architecture for model hand estimation which use multi-head self -attention module and ResNet backbone estimate accurately hand finger 21 key-points moving. Second, we use two stack sub net to progressively refine key-point detection accuracy. Our whole network structure is visualized in Figure 1, we achieved 93.68% PCK score on our mixed test dataset which is better than most of the state-of-the-art traditional CNN model.

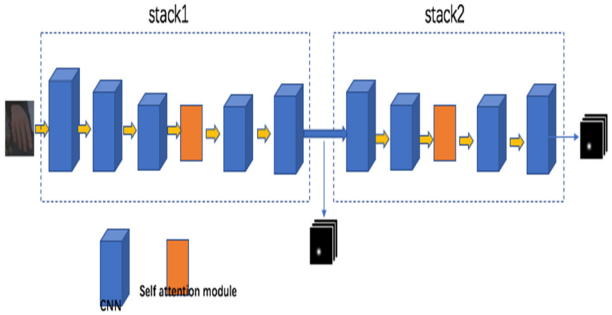

Fig1. The proposed self attention stacked network structure

2 Related work

A lot of large progress were achieved in pose estimation field because of some human pose benchmark and the advent of CNN and RNN in the last years. There are two main kinds of methods for pose estimation, the first one is the CNN architecture of Toshev [5] which directly regresses 2D cartesian coordinates using multiple ResNet

*Corresponding author's e-mail: 25879030@qq.com

blocks from one or more color image inputs, another one is the Thompson et al. [6] works which regress score maps. In the Thompson et al. works, a multi-resolution image pyramid structure was used for detection of hand key-points in 2D. In the Oberweger et al. [7] works, the relationship between different key-points was explored by coordinating in a compressing bottleneck.

In 3D hand pose estimation field, many works use a two part pipeline. First 2D key-points are detected by utilizing the discriminative feature extraction power of CNN layers and then this 2D detections can be feed to 3d net branch to regress 3D key-points. While these works are all on 3D body pose estimation, comparing with 2d pose estimation, 2d hand pose estimation is more difficult because of less dimension input image data, complex articulation and self-occlusion, as well as less training data being available.

There are not any "transformer" like approaches that tackle the problem of 2D hand pose estimation from a single color image. Previous approaches mainly depend on CNN structure which is not good at modeling prominent and discriminative feature maps.

3 our method

3.1 Hand pose representation and HandDecNet

Given I $\in$ $R_{N\times M\times 3}$ and I represents a single hand image which consists of 3 channel RGB data, to infer its 2D pose, we define the hand pose by a set of coordinates wi = (xi, yi), which describe the locations of k key-points in 2D space, i.e., i $\in$ [1,k] with k = 21 in our case.

To solve scale ambiguity, we train a scale-invariant 2D structure network to estimate normalized coordinates

$$\mathbf{w}_i^{\text{norm}} = \frac{1}{s} \cdot \mathbf{w}_i, \tag{1}$$

where s =||wk+1 – wk||$_2$ is a sample dependent constant and s is the normalized unit length distance. In our experiments, k is chose to be the bone of index finger.

For hand detection we deploy a SSD network architecture which provide the hand bounding box, by using the hand detection results we can crop and normalize the inputs in size, which simplifies the learning task for the PoseEstimationNet. We will not detail the hand detection method because of space cause.

3.2 Key-point score maps with PoseEstimationNet

In our paper, localization of 2D key-points were formalized as estimation of 2D score maps c={c1(u,v),...,cJ(u,v)}. We train a segmentation like network to predict k score maps .

ci $\in$ $R_{N\times M}$, where each map contains information about the likelihood that a certain key-point is belong to a hand articulation point, and the key-point spatial distribution is a gaussian distribution. The proposed network uses an encoder-decoder architecture, a multi-head self-attention module as the intermediate bottleneck was inserted to improve feature extraction power, self-attention can combine detail and spatial information in different level feature maps of the network, then two encoder-decoder architecture was stacked to successively refine the estimation accuracy. A complete overview over the network architecture is located in figure 1, the details self-attention module is showed in figure 2.

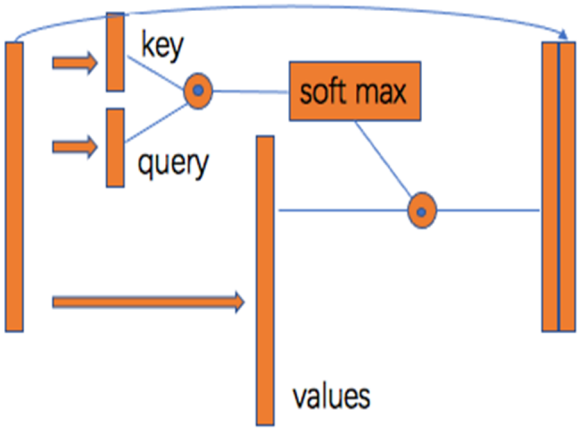

Fig2. details of self-attention module

3.3 Loss and training details

We apply standard softmax cross-entropy loss and L2 loss for the proposed PoseEstimationNet. We use α to balance two loss, and in our experiments α was set to be 0.7, The total loss function L is the weighted sum of CE loss and L2 loss.

$$L_1 = criterion(P_{GT} - P_{pre}) \tag{2}$$

$$L_2 = \left\|P_{GT} - P_{pre}\right\|_2^2 \tag{3}$$

$$L = \alpha \times L_1 + (1-\alpha) \times L_2 \tag{4}$$

Where P_{GT},P_{pre} is the ground truth hand pose 21 points and predicted pose points respectively. Criterion represents for cross-entropy loss.

The architecture of our PoseEstimationNet was showed in Table 1. In all our experiments, we set batch size to be 24 and using ADAM solver. The network is trained for 30000 iterations using our weighted loss. In the beginning 2000 steps, we used warm-up to stable the training procedure, then the learning rate is set to 3×10^{-4} for the first 10000 iterations, 3×10^{-5} for following 8000 iterations and 3×10^{-6} until the end. For the input 320*240 image, we use normal distributions to generate the ground truth score maps with a variance of 30 pixels and the mean being equal to the ground truth key-point location. After we normalized score maps position distribution, each map pixel value was normalized to be 0 to 1 such that the visible key-points were set to be 1, the invisible key-points were set to be zero everywhere.

We train the propose PoseEstimationNet on resized image which has the resolution of 256×256. The bounding box is chosen such that all key-points of a single hand are contained within the crop. We augment the cropping procedure by modifying the calculated bounding box. To be specific, we add noise to the calculated center of the bounding box, which is sampled from a zero mean normal distribution with variance of 8 pixels. Then, the size of the bounding box is changed accordingly to contain all hand keypoints.

Table1. The architecture of our PoseEstimationNet

id	Name	Kernel	Dimensionlity
	Input image	-	256*256*3
1	Conv+relu+bn	3*3	256*256*64
2	maxpool	2*2	128*128*64
3	Conv+relu+bn	3*3	128*128*96
4	Conv+relu+bn	3*3	128*128*96
5	maxpool	2*2	64*64*96
6	Conv+relu+bn	3*3	64*64*128
7	Conv+relu+bn	3*3	64*64*128
8	maxpool	2*2	32*32*128
9	Conv+relu+bn	3*3	32*32*256
10	Conv+relu+bn	3*3	32*32*256
11	conv	1*1	32*32*21
12	Concat(10,11)	-	32*32*277
13	Conv+relu+bn	3*3	32*32*128
14	Conv+relu+bn	3*3	32*32*128
15	Conv+relu+bn	5*5	32*32*128
16	Conv+relu+bn	5*5	32*32*128
17	ElementAdd(16,15)	-	32*32*128
18	Conv+relu	3*3	32*32*128
19	conv	1*3	32*32*21
20	conv	3*1	32*32*21
21	conv	1*1	32*32*21

4 EXperiments

In this paper, we used the so-called Stereo Hand Pose Tracking Benchmark [8] apply to our estimation problem, though the dataset provided both RGB images and 3D pose annotation, we only use the 2D colour images. The dataset includes 2D annotations of 21 key-points for 18000 pairs, and all the images have a resolution of 640 × 480. All the images in the dataset were taken under varying lighting conditions and there are 6 different backgrounds.

In our experiments, the dataset was divided into two subsets, an evaluation set of 1500 images and a training set with 16500 images. For training of PoseEstimationNet we apply standard softmax cross-entropy loss and L2 loss in our experiments. Figure 2 shows some qualitative results of this configuration.

We report the percentage of correct key-points (PCK) comparing with the approach from[8] for different error thresholds in table 2; The results show that the method works on our test dataset well, from the details of intermediate feature maps, the conclusion can be made that self -attention mechanism can capture salient hand key-point feature which is essential for improving the hand key-points detection performance.

Table2. PCK for different error thresholds

hPCK(mm)	0.15	0.2	0.25	0.3
CPM	84.7%	88.5%	90.9%	92.6%
Proposed	84.68%	99.05%	91.2%	93.68%

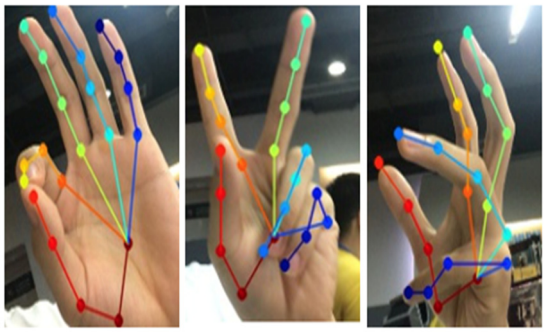

Fig3. qualitative results using the images shot from real room conditions

4.1 Additional experiments for sign language recognition

An important practical application for hand pose estimation are sign language recognition, but most previous hand pose estimation approaches depend on depth data, and they cannot be applied to sign language recognition datasets consisting of only color images. In our last experiment, we used our hand pose estimation system and trained a simple GestureNet for gesture recognition on top of it. The GestureNet is consisted of two CNN layers and two fully connected layers with ReLU activation functions, the details of the architecture of the network was presented in table 3.

In all our gesture recognition experiments, the so-called RWTH German Fingerspelling Database[9] was used. It contains 35 gestures representing the letters of the alphabet, German umlauts, and the numbers from one to five. The dataset comprises 20 different persons, and every one did two recordings each for every gesture. In our experiments, we only used the subset restricted to 30 static gestures for easy comparison.

Table3. The architecture of our GestureNet

id	Name	Kernel	Dimensionlity
	Input PointsPre	-	2*21
1	Conv+relu	1*7	2*21*16
2	Conv+relu	1*7	2*21*16
3	meanpool	2*1	1*21*16
4	FC+ReLU+ Dropout(0.5)	-	128
5	FC+ReLU+ Dropout(0.2)	-	35

All of the data in the database are short video sequences of 320×240 resolution, it was recorded by two different cameras, but we used only one camera data. We used the middle frame and randomly select the begin and last frames from each video sequence, the proportion is 2:1, then we selected the gesture class as labels. This dataset has 1160 images, and we divided it into a validation set with 200 images, and the remained images was training set. For the sake of consistency, all of the images were resized to 320 × 320 pixels and trained on randomly sampled 256 × 256 crops. Because the images were taken from a compressed video stream, the data distribution may be different, it is necessary to augment data and finetune our PoseEstimationNet. Thus, we labeled 50 images from the training set with hand keypoints, which we augmented to 500 images, then the generated images were used to finetune our PoseEstimationNet. After that the pose estimation part is kept fixed and we solely train the GestureNet. Table 4 shows that we can archive even small WER comparing with Dreuw et al. [9] on the subset of gestures.

Table4. Word error rates in percent on the RWTH German Fingerspelling Database

Method	WER(word error rate)
Dreuw[9]	35.7%
Dreuw on subset[9]	36.6%
The propose methond	34.6%

5 conclusion

In this paper, we propose multi-head self-attention module and two stacked encoder-decoder network architecture strategy to tackle 2d hand pose estimation from a single color image. Quantitative experimental results on SHPTB benchmark[8] show the effectiveness of our strategy, in our future works, we will focus on the quantitative analysis how “transformer” like module help to improve the pose key-points detection PCK.

Acknowledgment

This paper was co-supported by the Scientific Research Project of Education Department of Hubei Province under grant B2019272. Thanks for the support of Education Department of Hubei Province and Wuhan Donghu University.

References

1. Dzmitry Bahdanau, Kyunghyun Cho, and Yoshua Bengio. Neural machine translation by jointly learning to align and translate. In Third International Conference on Learning Representations, 2015.
2. Matthew E. Peters, Mark Neumann, Mohit Iyyer, Matt Gardner, Christopher Clark, Kenton Lee, and Luke Zettlemoyer. Deep contextualized word representations. arXiv preprint arXiv:1802.05365, 2018.
3. Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, Łukasz Kaiser, and Illia Polosukhin. Attention is all you need. In Advances in neural information processing systems, 2017.
4. A. Toshev and C. Szegedy. Deeppose: Human pose estimation via deep neural networks. In Proc. of the IEEE Conf. on Computer Vision and Pattern Recognition (CVPR), pages 1653–1660, 2014
5. J. Tompson, M. Stein, Y. Lecun, and K. Perlin. Real-Time Continuous Pose Recovery of Human Hands Using Convo lutional Networks. ACM Transactions on Graphics, 33(5):1– 10, Sept. 2014
6. M. Oberweger, P. Wohlhart, and V. Lepetit. Hands deep in deep learning for hand pose estimation. arXiv preprint arXiv:1502.06807, 2015.
7. S.-E. Wei, V. Ramakrishna, T. Kanade, and Y. Sheikh. Convolutional pose machines. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pages 4724–4732, 2016.
8. J. Zhang, J. Jiao, M. Chen, L. Qu, X. Xu, and Q. Yang. 3d Hand Pose Tracking and Estimation Using Stereo Matching. arXiv preprint arXiv:1610.07214, 2016.
9. P. Dreuw, T. Deselaers, D. Keysers, and H. Ney. Modeling image variability in appearance-based gesture recognition. In ECCV Workshop on Statistical Methods in Multi-Image and Video Processing, pages 7–18, Graz, Austria, May 2006.