Cycle Fusion Network for Multi-Person Pose Estimation

Yuanxiang Wang¹ and Teng Wang¹∗

¹School of Automation, Southeast University, Nanjing 210096, China
∗Corresponding authors e-mail: wangteng@seu.edu.cn

Abstract. Multi-person pose estimation has been largely improved with the development of convolutional neural network, which however remains a challenging task in computer vision. The challenge is particularly pronounced in high flexibility of occlusion, camera angles, imaging-caused body limbs. Existing methods are mostly based on networks with deep architectures, which are capable of inferring the semantic labels with high accuracy while suffering from positioning accuracy degeneration caused by the loss of the spatial information. To address these problems, we present a novel network called cycle fusion network to improve the pose estimation accuracy in multi-person environments. The cycle fusion network has a two-stage structure. In the first stage, a cycle-path feature fusion is used to increase the semantic information for low level layers and the spatial information for high level layers respectively. In the second stage, a cross-scale feature fusion is used for refining the positions of keypoints. Equipped together with RFB module and multi-level supervision, the proposed method has shown the state-of-the-art performance on the large-scale MS COCO benchmark with extensive experiments.

1. Introduction

Human pose estimation which aims at positioning key points such as joints and facial features of the human body in 2D images, has received increasing attention in recent years because of its various applications like application action recognition and virtual fitting. Pose estimation, especially multiple person pose estimation in the wild environment is an extremely challenging task in computer vision due to the high complexity caused by a wide range of factors including imaging-caused body limbs, self and outer occlusion, various camera angles, background noises, various gestures of individual persons.

Early work targeting pose estimation combined handcrafted features with a spatial model to capture both local and global information[1] [2], [3], [4]. Recent researches[5], [6], [7], [8][9] have turned to deep Convolutional Neural Networks (CNN) and made significant progresses in improving the estimation accuracy since deep networks are much more powerful in inferring the semantic labels of the regions in the image. To achieve more accurate semantic inference, recently, some deep model based researches [9], [10], [11] proposed to use more context information by deepening the CNN architectures or enlarging the receptive fields with more powerful convolution schemes.

Admittedly, these methods have well addressed the problem of inferring the semantic labels of the regions around the keypoints of the human body. However, they are not quite effective in positioning the keypoints since the spatial information required to accurately position the keypoints is partially lost during the feature extraction through multiple layers. Present state-of-art methods therefore proposed to combine low-level features with high-level features to obtain both rich semantic information and spatial information for keypoint recognition and positioning. For instance, stacked hourglass network[3] optim
Figure 1. Illustration of the architecture of CFN. Functionally, Block coarse deeply fuses the semantic information and spatial information; Block refine concatenates multi-level semantic and spatial information and generates a refined estimation.

izes both the semantic and spatial inference by repeated a hourglass module which captures and consolidates information across all scales of the image with a bottom-up-top-down architecture. CPN[12] injects the semantic information from deeper layers back to the shallower layers with more spatial information in its cascaded front end and concatenates the merged features of different scales in the back end for more accurate keypoint positioning.

In this paper, we propose a new architecture named Cycle Fusion Network (CFN) for multi-person pose estimation problem. The whole framework follows a classic top-down pipeline which first employs a human detector to generate human bounding boxes and then uses CFN to recognize and localize all the keypoints of human body within each single bounding box. Our major contribution of the framework is CFN, which has an architecture where spatial information and semantic information are fused much more deeply than previous work. The two-stage network structure of CFN is shown in Figure 2. The first stage, called cycleNet, is a sub network based on ResNet backbone. This sub network learns a coarse estimation for all keypoints. More importantly, cycleNet deeply fuses the semantic and spatial information through a cycle feature fusion path. By deep multi-level feature fusion, it provides sufficient context information for the second stage called wedgeNet. Functionally, wedgeNet infers a refined estimation for all keypoints. It has a wedge-like multi-scale structure with four channels built on the backbone of RFB. WedgeNet injects the lower-level multi-scale features from upper channels to lower channels for enhancing spatial information. Taking the deeply fused spatial and semantic features from cycleNet as the input, wedgeNet mainly targets a tuning of keypoint position through mining a larger field of context information.

We have evaluated the proposed method on MS COCO benchmark. It achieves the average precision of 74.2 which is a 3% relative improvement compared with CPN from the COCO 2018 keypoint challenge. The rest of this paper is divided as follows. In the next section, we present a review of the most relevant related work. The proposed method is presented in Section 3. In Section 4, we show the experimental evaluations, followed by our conclusions in Section 5.
2. Related work
In this section, we present some of the most relevant methods to our work, which are divided into CNN-based human pose estimation and hard example mining.

**CNN-based human pose estimation.** Early pose estimation methods are usually based on local modeling of joints. This approach has great limitations on the ability to represent because it uses local detectors and can only model partial relationships between body joints. In the last few years, with the development of deep convolutional neural networks, pose estimation has made a major breakthrough. Some methods treat pose estimation as a point regression problem\[13\], [14], [15], [16]. Toshev and Szegedy\[9\] first introduced deep neural networks into the field of pose estimation. They proposed a cascaded CNN joint point regression method, which estimates the attitude roughly on a whole image, and then optimizes the prediction results of neighborhood sub-images (with higher resolution) using multiple CNN-based regressions with the same structure. Carreira et al.\[16\] proposed an error iterative feedback framework that predicts the deviation of each position and then performs early error correction to improve accuracy. Diogo C. Luvizon et al.\[18\] proposed soft-argmax, which makes the process of transforming the feature map to coordinates different, so the entire regression network can be end-to-end training.

However, the direct regression of keypoints is difficult to converge and has a general effect, which is usually considered to be sub-optimal. Recent pose estimation methods have yielded significant improvements in accuracy by using heatmaps, demonstrated the importance of receptive fields in such problems\[19\], [20], [21], [22], [23], [24]. For example, Newell et al.\[3\] proposed stacked hourglass networks with symmetrical coding and decoding structures, which is a coarse-to-fine process that enlarges the receptive field while maintaining resolution. Following the similar strategy, Xiao et al.\[25\] proposed a stacked hourglass network with CRF based multi-resolution attention mechanism to generate attention maps with different resolution features. Wei et al.\[38\] integrate CNN into Pose Machines and use multi-stage network structure to gradually accurately locate joint points. Recently, Chen et al.\[12\] proposed Cascaded Pyramid Network, which integrates the characteristics of different scales and different receptive field sizes, and achieves state-of-the-art performance. As such, our model is based on detection to generate the heatmaps of keypoints.

**Hard example mining.** In a human pose estimation task, the training set is distinguished by a large imbalance between the number of easy examples and the number of hard examples. The proportion of difficult samples in training set is low. However these hard samples tremendously affect the performance of the model.

There is recent work \[26\], [27], [28] that selects hard examples for training deep networks. All these methods base their selection on the current loss for each datapoint. \[27\] investigates online selection of hard examples for mini-batch SGD methods. \[26\] focus on online hard example selection strategy for region-based object detectors. CPN adopts the online hard keypoints mining strategy to localize the hard keypoints. Our model adopts the similar strategy. Complementary to CPN, we focus on the hard keypoints of the whole batch rather than the hard keypoints of a single person.

3. Proposed method
Human pose estimation aims to locate pre-set K keypoints of the human body in an image. These keypoints are usually some areas with significant features in the human body(e.g., eye, elbow, knee, etc).

The proposed approach follow the top-down pipeline, which first locates the position of the human body in the image by the detector and then estimates the pose on each human bounding box. In what follows, we first present the global architecture of our method, and then detail its important parts.
3.1. Network architecture

An overview of the proposed method is presented in Fig.1. Our approach is based on a coarse-to-fine convolutional neural network essentially composed of two main parts: a Resnet-based coarse positioning network, and a refined multi-resolution aggregated head.

The network architecture has its entry flow based on ResNet[29] that is used to provide basic features extraction. For ResNet we use the features output by each stages last residual block, which we denote as \{C2, C3, C4, C5\}. The feature maps is reduced to half after each stage. Low-level feature maps usually contain a lot of spatial information that facilitates accurate positioning because of their larger resolution. On the contrary, small-resolution high-level feature maps contain more semantic information that facilitates classification. As for the human pose estimation, the integration of two types of information are vital.

Recently, the U-shaped structure has been widely used, which can simultaneously retain low-level spatial information and high-level semantic information. Similarly to what is found in FPN[30], an U-shape structure are used in Block A to combine information in different resolutions. At the same time, in order to make better use of the low-level spatial information, we added a bottom-up shortcut to shorten the propagation path of lower-level but higher-resolution information.

For Block-B, which is used to refine the output of front network, we take the four different scale output feature maps of the Block-A named \{H1, H2, H3, H4\} as the input. In order to make the extracted features have more different subfield information and richer context information, an intuitive method is to increase the width of the network, in other words, to use a multibranch convolution layer. Therefore, based on the parallel multi-scale of block B, we adopt a multi-branch structure similar to that in RFB [31] to further enrich the network’s receptive field. For each multi-branch module, the first branch contains a 1×1 convolution layer and a 3×3 convolution layer, and the second branch is followed by an additional dilated convolution layer of stride 2, and the stride of the dilated convolution layer becomes 3 in the third branch. Finally, concatenate the feature maps of the three branches, and shortcut design is also applied. And we have adopted different numbers of multi-branch modules at different scales.

In addition, in BLOCK B, the operation of the four scales is almost independent, which makes the fusion effect of features in different stages not good. To mitigate this issue, inspired from the feature fusion work [32] we have adopted a cross-scale feature fusion, which is given as follows,

\[
h_{s,r} = C[h_{s,r-1}, f(h_{s-1,r-1})]
\]

where \(h_{s,r}\) represents the output feature maps of \(r\) th RFB unit in the \(s\) th stage, and \(C\) is the concatenating operation, \(f\) is a convolution layer with stride 2.
At the end of the network, a $3 \times 3$ convolution layer is used to generate $K + 1$ ($K$ keypoints and a background) confidence maps.

3.2. Multi-level supervisions

The focus of the network should be distinct at different stages. At the initial stage of the network, all we need is a rough positioning. Moreover, different types of key points should also be distinguished in terms of supervision. Training with looser supervision could help detect the ambiguous or indistinct keypoints, but this comes at a cost to localization accuracy for those keypoints with distinctive appearances. Therefore, we applied the strategy of multi-level supervision, which is described below.

**Loss.** For the pose estimation task, we train the network with the mean square error between the groundtruth and the predicted heat map as defined in the equation bellow:

$$L = \frac{1}{MN} \sum_{m=1}^{M} \sum_{n=1}^{N} \left( \| \hat{h}^m_{ij} - h^m_{ij} \| \right)$$

where $M$ is the number of training images, $N$ is the number of keypoints for a single person, $\hat{h}$ is the groundtruth heatmap, $h$ is the heatmap predicted by the network.

**Multi-level Keypoint Supervision.** The performance improvement through the use of intermediate supervision has been proven in many previous work. In our approach, intermediate supervision is also used to ensure normal updates of the underlying parameters. Besides, we also use Keypoint Multi-level Supervision, which uses various strict levels of supervision for different types of key points and different stages of the network.

In order to gradually improve the positioning accuracy of network, we use different sizes of Gaussian kernels to generate ground truth heatmaps for different stages of the network, and deeper networks use smaller Gaussian kernels. In addition, as shown in Fig.2, although the deviation are the same, the final keypoint similarities differs depending on the type of key points, because of the different size of the human joints(e.g. eye and knee). Therefore, on the basis of multi-level supervision, we also use different sizes of Gaussian kernels to generate ground truth heat maps for keypoint types of different sizes so that the final loss actually reflects the error of keypoint similarity, which is given as follows,

$$H^j_{x,y} = 255 \times e^{-\frac{(x-x^j)^2+(y-y^j)^2}{2\sigma^2}}$$

where $H^j_{x,y}$ represents the pixel values of the $j$th joint ground truth heatmap at coordinates $(x, y)$, $(\hat{x}, \hat{y})$ is the ground truth coordinate, and $\sigma$ is a normalize parameter, which has different values for different types of keypoints.

**Global Hard Keypoint Mining.** In human pose estimation, the difficulty between the samples is quite different, and many unoccluded keypoints can be well positioned in the coarse positioning stage. Therefore, deeper networks should focus on those keypoints that are inaccurate or mistargeted. The online hard keypoints mining(OHKM) strategy proposed in CPN made some modifications to the keypoint positioning based on OHEM. They only punish the top $P$ ($P \leq N$) keypoint losses out of $N$ (the
Table 2. AP of basenet with different components. CPN is used as basenet. PA means path aggregation. RFBc denotes RFB with cross-scale feature fusion. MKS means Multi-level Keypoint Supervision. And GHKM means Global Hard Keypoint Mining.

| Components | Base | PA | RFBc | MKS | GHKM | AP  |
|------------|------|----|------|-----|------|-----|
|            | √    | √  | √    |     |      | 69.4|
|            | √    |    |      | √   |      | 69.9|
|            | √    |    |      |     | √    | 70.2|
|            | √    |    |      |     | √    | 70.4|
|            | √    |    |      |     |      | 70.5|

Table 3. Comparison of different hard keypoint numbers in global hard keypoint mining.

| Number of annotated keypoints | AP(OKS) |
|-------------------------------|---------|
| 4                             | 69.9    |
| 6                             | 70.2    |
| 8                             | 70.5    |
| 11                            | 70.4    |
| 13                            | 70.2    |
| 15                            | 70.3    |

number of annotated keypoints in one person, say 17 in COCO dataset) in the refining phase of the network. OHKM only trains top P keypoint losses of a single person. It is possible that these P keypoints are difficult in a single sample, but they are not difficult in all samples. Therefore, we search for difficult key points throughout the batch instead of a single sample.

4. Experiments

We evaluate the proposed method on two widely used benchmarks on pose estimation, i.e., the very challenging MS COCO Keypoint [40] and MPII Human Pose [41] datasets. All our networks are pre-trained on the Imagenet classification dataset [42].

4.1. Implementation Details

Dataset. The MPII dataset includes around 25K images containing over 40K people with annotated body joints, which 15K are training samples, 3K are validation samples and 7K are testing samples.

MS COCO train2017 split contains 64K images including 260K person instances which 150K of them have keypoint annotations. Keypoints of persons with small area are not annotated in COCO. We did ablation experiments on COCO val2017 split which contains 2693 images with person instances.

Data Augmentation. According to the detection results, we first extend the bounding box, then crop the images with the target human centered at the images. The cropped images resized to 384×288 pixels on MS COCO. After cropping from images, We perform random rotations (−40°, 40°) and random rescaling from 0.7 to 1.3 to make the network more robust to different scales and directions.

Training and Testing Details. For the MS COCO dataset, the network is trained on 2 Nvidia TITAN Xp GPUs with mini-batch size 12 per GPU. We optimize the network using back propagation and the adam optimizer. The learning rate begins at 1e-3 and decreases by a factor of 0.5 every 40 epochs (5k iterations a epoch). The weight decay is 1e-5. And we train the network for 300 epochs.

For the testing, we adopt the same strategy on MS COCO dataset. We average the predicted heatmaps of original image with heatmaps of flipped image. Then, use a Gaussian filter on the heatmap to predict the final keypoint coordinates.

4.2. Results on MSCOCO Keypoint Dataset

We evaluate our method on the MSCOCO Keypoint dataset. The COCO evaluation defines the object keypoint similarity (OKS) and uses the mean average precision (AP) over 10 OKS thresholds as main

Table 4. Comparison of different hard keypoints mining strategies applied in Block refine. “OHKM” means the online hard keypoints mining adopted in CPN. “GHKM” means our global hard keypoint mining.

| Strategy | OHKM | GHKM |
|----------|------|------|
|          |      |      |
Table 5. Comparison of different feature fusion strategies applied in Block refine. C-scale* means only applies crossscale feature fusion in the last RFB module of each layer in refine block.

| Strategies | AP(OKS) | C-scale* | C-scale |
|------------|---------|----------|---------|
|            | 69.4    | 69.5     | 69.7    |

4.3. Ablation Study

In this subsection, we performed varied experiments to show the contributions of each component of our methods. All experiments in this subsection are trained on MS COCO train2017 dataset and evaluated on MS COCO val2017 dataset. And Mask R-CNN[34] is adopted as our human detector. Unless otherwise stated, the input size of all models is 256×192, and the base model is ResNet50. The overall result is shown in Table 2. CPN based on Resnet50 is used as the baseline. Path aggregation strategy, which is a bottom-up shortcut, brings about a 0.5 gain from 69.4 to 69.9. In order to ensure that the FLOPs is roughly the same, we appropriately reduce the number of channels in the original RFB module. The use of RFB module and cross-scale feature fusion further brings about a 0.3 gain. And multi-level keypoint supervision and global hard keypoint mining also brings gain of 0.1 and 0.2. Finally, after adopting all the above strategies, our model obtains a 1.2 improvement.

Global Hard Keypoint Mining. Here we discuss the implications for the performance of Global Hard Keypoint Mining strategy used in our network. In the global hard keypoint mining, we only train the network with the loss of the top \( P \times B \) \( (P \leq N) \) keypoints of each batch. The value of \( P \), which is an artificially set hyperparameter, greatly affects the performance of the network. Table. 3 shows the performance of the network of different value of \( P \). As shown in the Table. 4, we compare the performances of models without hard keypoint mining strategy, with GHKM and with OHKM used in CPN. When online hard keypoints mining is applied, the performance of the network increases 0.2 AP and achieves 70.5 AP. And the performance of the network further increase when we use global hard keypoint mining instead of online hard keypoints mining.

Cross-scale Feature Fusion. Table 5 shows the influence of our cross-scale feature fusion used in block refine on performance. When cross-scale feature fusion strategy is not used, the performance of the network is 69.4 AP. When using A at the end of each layer in block refine, the performance of the network increases 0.1 AP and yields the results of 69.5 AP. And applying the same cross-scale feature fusion in each RFB module of each layer in block refine, which increases the result by 0.2 AP and finally achieves 69.7 AP.
5. Conclusion
In this paper, follow the top-down pipeline, we present a coarse-to fine network to better fusion features and locate keypoints. More specifically, our model consists a coarse block and a refine block. The coarse block is a pyramid structure with path optimization in feature fusion to locate the approximate location of key points. And the refine block is a multi-branch structure with a cross-scale feature fusion to refine the previous results. In addition, a multi-level keypoint supervision strategy with global hard keypoint mining is adopted to distinguish different types of key points and address difficult samples.
References

[1] Min Sun and Silvio Savarese. Articulated part-based model for joint object detection and pose estimation. In 2011 International Conference on Computer Vision, pages 723–730. IEEE, 2011.

[2] Wang Fang and Li Yi. Beyond physical connections: Tree models in human pose estimation. 2013.

[3] Alejandro Newell, Kaiyu Yang, and Jia Deng. Stacked hourglass networks for human pose estimation. In European Conference on Computer Vision, pages 483–499. Springer, 2016.

[4] Pedro F. Felzenszwalb and Daniel P. Huttenlocher. Pictorial structures for object recognition. 61(1):55–79, 2005.

[5] V. Ferrari, M. Marin-Jimenez, and A. Zisserman. Progressive search space reduction for human pose estimation. In IEEE Conference on Computer Vision & Pattern Recognition, 2008.

[6] Georgia Gkioxari, Alexander Toshev, and Navdeep Jaitly. Chained predictions using convolutional neural networks. In European Conference on Computer Vision, 2016.

[7] Arjun Jain, Jonathan Tompson, Yann Lecun, and Christoph Bregler. Modeep: A deep learning framework using motion features for human pose estimation. 2014.

[8] Tomas Pfister, Karen Simonyan, James Charles, and Andrew Zisserman. Deep convolutional neural networks for efficient pose estimation in gesture videos. In Asian Conference on Computer Vision, 2014.

[9] Alexander Toshev and Christian Szegedy. Deeppose: Human pose estimation via deep neural networks. In Proceedings of the IEEE conference on computer vision and pattern recognition, pages 1653–1660, 2014.

[10] Shih En Wei, Varun Ramakrishna, Takeo Kanade, and Yaser Sheikh. Convolutional pose machines. 2016.

[11] Shaoli Huang, Mingming Gong, and Dacheng Tao. A coarse-fine network for keypoint localization. In Proceedings of the IEEE International Conference on Computer Vision, pages 3028–3037, 2017.

[12] Yilun Chen, Zhicheng Wang, Yuxiang Peng, Zhiqiang Zhang, Gang Yu, and Jian Sun. Cascaded pyramid network for multi-person pose estimation. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pages 7103–7112, 2016.

[13] Adrian Bulat and Georgios Tzimiropoulos. Human pose estimation via convolutional part heatmap regression. 2016.

[14] Diogo C. Luvizon, David Picard, and Hedi Tabia. 2d/3d pose estimation and action recognition using multitask deep learning. 2018.

[15] Sun Xiao, Jiaxiang Shang, Liang Shuang, and Yichen Wei. Compositional human pose regression. 2017.

[16] Joao Carreira, Pulkit Agrawal, Katerina Fragkiadaki, and Jitendra Malik. Human pose estimation with iterative error feedback. In Proceedings of the IEEE conference on computer vision and pattern recognition, pages 4733–4742, 2016.

[17] Matthias Dantone, Juergen Gall, Christian Leistner, and Luc Van Gool. Human pose estimation using body parts dependent joint regressors. In IEEE Conference on Computer Vision & Pattern Recognition, 2013.

[18] Diogo C. Luvizon, Hedi Tabia, and David Picard. Human pose regression by combining indirect part detection and contextual information. 2017.

[19] Lipeng Ke, Ming Ching Chang, Honggang Qi, and Siwei Lyu. Multi-scale structure-aware network for human pose estimation. 2018.

[20] Xuecheng Nie, Jiashi Feng, Junliang Xing, and Shuicheng Yan. Pose partition networks for multi-person pose estimation. 2018.

[21] Xiao Chu, Wanli Ouyang, Hongsheng Li, and Xiaogang Wang. Structured feature learning for pose estimation. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pages 4715–4723, 2016.
[22] Hao-Shu Fang, Shuqin Xie, Yu-Wing Tai, and Cewu Lu. Rmpe: Regional multi-person pose estimation. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 2334–2343, 2017.

[23] Mihai Fieraru, Anna Khoreva, Leonid Pishchulin, and Bernt Schiele. Learning to refine human pose estimation. 2018.

[24] Tao Kong, Anbang Yao, Yurong Chen, and Fuchun Sun. Hypernet: Towards accurate region proposal generation and joint object detection. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 845–853, 2016.

[25] Chu Xiao, Yang Wei, Wanli Ouyang, Ma Cheng, Alan L. Yuille, and Xiaogang Wang. Multicontext attention for human pose estimation. In *Computer Vision & Pattern Recognition*, 2017.

[26] Ilya Loshchilov and Frank Hutter. Online batch selection for faster training of neural networks. *arXiv preprint arXiv:1511.06343*, 2015.

[27] Abhinav Shrivastava, Abhinav Gupta, and Ross Girshick. Training region-based object detectors with online hard example mining. In *IEEE Conference on Computer Vision & Pattern Recognition*, 2016.

[28] Xiaolong Wang and Abhinav Gupta. Unsupervised learning of visual representations using videos. In *IEEE International Conference on Computer Vision*, 2015.

[29] Zhenli Zhang, Xiangyu Zhang, Chao Peng, Xiangyang Xue, and Jian Sun. Exfuse: Enhancing feature fusion for semantic segmentation. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pages 269–284, 2018.

[30] Muhammed Kocabas, Salih Karagoz, and Emre Akbas. Multiposenet: Fast multi-person pose estimation using pose residual network. In *Computer Vision and Pattern Recognition*, pages 3686–3693, 2014.
[42] Jia Deng, Wei Dong, Richard Socher, Li Jia Li, Kai Li, and Fei Fei Li. Imagenet: A large scale hierarchical image database. In *IEEE Conference on Computer Vision & Pattern Recognition*, 2009.