Dual Path Networks for Multi-Person Human Pose Estimation

Guanghan Ning, Zhihai He
University of Missouri
Columbia, MO

gnxr9@mail.missouri.edu, hezhi@missouri.edu

Abstract

The task of multi-person human pose estimation in natural scenes is quite challenging. Existing methods include both top-down and bottom-up approaches. The main advantage of bottom-up methods is its excellent tradeoff between estimation accuracy and computational cost. We follow this path and aim to design smaller, faster, and more accurate neural networks for the regression of keypoints and limb association vectors. These two regression tasks are naturally dependent on each other. In this work, we propose a dual-path network[9] specially designed for multi-person human pose estimation, and compare our performance with the openpose[2, 8] network in aspects of model size, forward speed, and estimation accuracy.

1. Introduction

The task of human pose estimation is to determine the precise pixel locations of body keypoints from a single input image [15, 21]. Human pose estimation is very important for many high-level computer vision tasks, including action and activity recognition, human-computer interaction, motion capture, and animation. Estimating human poses from natural images is quite challenging. An effective pose estimation system must be able to handle large pose variations, changes in clothing and lighting conditions, severe body deformations, heavy body occlusions [32, 31, 21]. It is naturally a regression task. With Convolutional Neural Networks (ConvNets) and many assistive methods such as batch normalization [16], resnet [13], and inception design [28, 29], single-person human pose estimation has recently achieved significant progress.

Recent research emphasis has been put on multi-person human pose estimation, where multiple individuals may exist in a natural scene. Compared to single person human pose estimation, where human candidates are cropped and centered in the image patch, the task of multi-person human pose estimation is more challenging. The best performance on MS COCO 2016 Keypoints challenge [1] is only around 60% in mean average precision (mAP).

Existing methods can be classified into two kinds of approaches, the top-down approach and the bottom-up approach. The top-down approach [11, 22] relies on a detection module to obtain human candidates and then apply a single-person human pose estimator to detect human keypoints. The bottom-up approach [8, 12, 34, 20], on the other hand, detects human keypoints from all potential human candidates and then assemble these keypoints into human limbs for each individual based on various data association techniques. The main advantage of the bottom-up approaches is its excellent tradeoff between estimation accuracy and computational cost. It takes the winner [8] of MS COCO 2016 keypoint challenge less than 200 ms to run the pose estimator for one frame on a Pascal TITAN X GPU. More importantly, contrary to top-down approaches, its computational cost is invariant to the number of human candidates in the image. We follow the bottom-up approach of the works of Zhe. et. al [8] and aim to design smaller, faster, and more accurate neural networks for multi-person keypoints regression. According to [8], their proposed Part Affinity Fields (PAF) and its corresponding data-association techniques are robust and reliable. More accurate keypoint and PAF regression would potentially increase the overall performance up to 10%.

In this work, we focus on the network regression part and leave the data association part to future works. We propose a dual-path network specially designed for multi-person human pose estimation, and compare our performance with the openpose [2] network in aspects of model size, forward speed, and estimation accuracy. Our contributions include: (1) We analyze the tasks of keypoint regression and PAF, where PAF estimation depends heavily on keypoints estimation but not vice versa. (2) We then design a dual-path network, the denseNet path responsible for PAF regression while the resNeXt path regressing human keypoints. Our performance is superior than the openpose [2] network even though the proposed network is of lower computational complexity and smaller model size.

The rest of the paper is organized as follows. In section

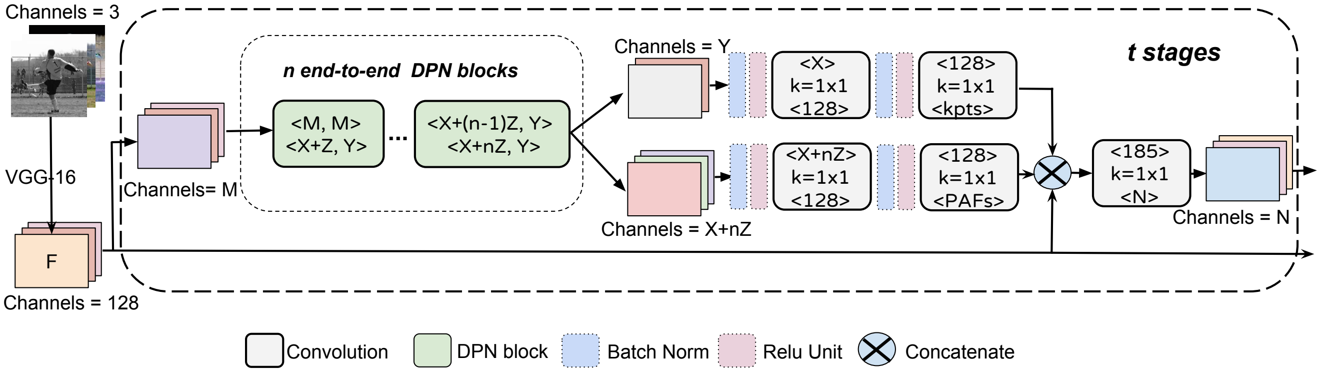

Figure 1. **Our proposed network**. The image feeds into the first 16 layers of VGG [26] network and outputs 128 channels of low-level visual features. These features are subsequently fed into each stage of the repetive sub-network illustrated in this image. Each DPN block takes as input two branches of feature maps and also output feature maps of two branches, one with a consistent number of channels, the other one with accumulated channels over DPN blocks.

2, we provide a brief review of recent works on multi-person human pose estimation. Section 3 introduces the proposed network. Section 4 presents our experimental results. Section 5 concludes our paper.

2. Related Work

2.1. Single-Person Human Pose Estimation

This task is simpler than multi-person pose estimation because it aims to estimate the pose of a single person, where the image is cropped assuming the person dominates the image content. Traditional methods for single-person human pose estimation are mostly based on pictorial structure models [25, 23, 27, 30, 10, 19]. Since the work of *DeepPose* by Toshev *et al.* [32], research on human pose estimation has shifted from traditional approaches to deep neural networks (DNN) due to their superior performance. Recent methods [21, 33, 15, 7] have achieved quite accurate performance on popular datasets [6, 18]. However, the assumption that the person can always be correctly located is not necessarily satisfied.

2.2. Multi-Person Human Pose Estimation

Multi-person human pose estimation is a more realistic problem. It attempts to estimate the poses of multiple persons in natural scenes. It is quite challenging due to the variance of sizes and scales of the persons. Existing methods can be classified into two kinds of approaches, the top-down approach and the bottom-up approach.

The top-down approach [11, 22] relies on a detection module to obtain human candidates and then apply a single-person human pose estimator to detect human keypoints. Insafutdinov *et al* [15] propose a pipeline which uses the Faster R-CNN [24] as detection module and a unary DeeperCut as their single-person pose estimator. Their method achieves 51.0 in mAP on MPII dataset [6]. Because the single-person pose estimator is usually sensitive to the detection results, this approach requires the detection module to be very robust. More accurate performance has been achieved by Hao *et al* [11]. Their framework facilitates pose estimation in the presence of inaccurate human bounding boxes by introducing more components into the pipeline that refine the detection and pose estimation results.

The bottom-up approach [8, 12, 34, 20], on the other hand, detects human keypoints from all potential human candidates and then assemble these keypoints into human limbs for each individual based on various data association techniques. Many of these techniques are graph-based [8, 34]. The main advantage of the bottom-up approaches is its excellent tradeoff between estimation accuracy and computational cost. The winner of COCO2016 [1] proposes to estimate human keypoints as well as *Part-Affinity Fields* (PAF) simultaneously. PAFs are limb association vectors that can be used to assemble the keypoints into multi-person poses with certain graph-based association techniques. According to [8], their proposed PAF and corresponding data-association techniques are robust and reliable. More accurate keypoint and PAF regression would potentially increase the overall performance up to 10%. We follow their works and focus on the network regression part, aiming to design smaller, faster, and more accurate neural networks for multi-person human pose estimation.

2.3. Dual Path Networks

According to [8], their proposed data-association technique is robust and reliable; more accurate keypoint and PAF regression would potentially increase the overall performance up to 10%. Motivated by this, we look into network engineering and explore more robust and efficient learning of features and spatial inter-dependencies.

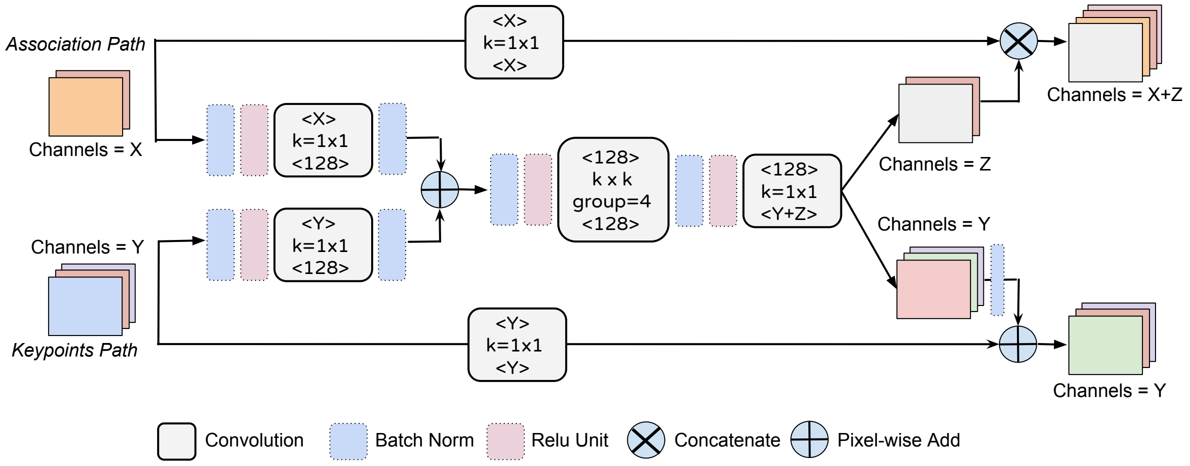

Figure 2. **Proposed DPN block**. It consists of two paths: the keypoints path and the data association path. The regression of human keypoints and association vectors are dependent on each other and share information from the previous block. The association vectors however, need more features to further explore spatial interdependency and they are regressed with accumulated channels of feature maps from all previous DPN blocks.

Dual Path Networks (DPN) is first proposed in [9] as a hybrid network design that incorporates the core idea of DenseNet [14] with that of ResNeXt [35]. ResNeXt is a variant of the widely-used ResNet [13], introducing a homogeneous, multi-branch architecture that has a new dimension called *cardinality* (the size of the set of transformations), as an essential factor in addition to the dimensions of depth and width. They show that increasing cardinality is able to improve classification accuracy, and is more effective than going deeper or wider when we increase the capacity of the network. The core of denseNet is that it connects each layer to every other layer in a feed-forward fashion. They alleviate the vanishing-gradient problem, strengthen feature propagation, encourage feature reuse, and substantially reduce the number of parameters.

According to the research of DPN, ResNet and its variants enable feature re-usage while DenseNet enables new features exploration which are both important for learning good representations. By carefully incorporating these two network designs into dual-path topologies, DPN shares common features while maintaining the flexibility to explore new features through dual path architectures.

Inspired by the DPN network that is originally designed for the task of image classfication, we aim to design a variant of DPN that is specially tailored for multi-person human pose estimation because the regression tasks for keypoints and association vectors are naturally two paths. The two tasks are dependent on each other but unique in their own ways. In the next section, we introduce our proposed DPN network, describe how the regression of keypoints and association vectors are assigned to each path and, explain the intuition behind it. For detailed description of part association techniques, please refer to the original paper of PAF [8].

3. Proposed Method

The proposed network is highly modulized. We intentionally follow the general network structure of openpose. As shown in Figure 1, there are multiple stages of repetitive subnetworks, where each subnetwork outputs estimated heatmaps of keypoints and PAFs and is enforced with loss functions as intermediate supervision. The network is first fed with an image into the first 16 layers of the VGG network and outputs 128 channels of low-level visual features, denoted by $\mathbf{F}$. These features are then fed into each stage of the following subnetworks. The modules in the figure only indicates the input and output channels and leaves out the resolution because the convolutional layers are all padded such that the resolution of the feature maps do not change.

Our proposed network differs from the openpose network in the structure of the subnetwork patterns, specifically, the DPN blocks. As shown in Figure 2, our proposed DPN block consists of two paths. The regression of human keypoints and association vectors are dependent on each other and share information from the previous block. With the operator of element-wise addition, the *Keypoints Path* (KP) leverages features before and after the feature fusion/transition within a DPN block. The association vectors in the *Association Path* (AP), however, accumulate features over blocks to further exploit spatial interdependencies. They are regressed with accumulated channels of feature maps from all previous DPN blocks. With such representation, features from the AP path is less constrained and more flexible than the KP path. It enforces the AP path to learn features at a higher level compared to the KP path, even though they are dependent and share common features.

It is declared in [8] that most of their false positives come from imprecise localization, other than background confusion and that there is more improvement space in capturing spatial dependencies than in recognizing body parts appearances. Therefore, we set the learning rate for the VGG layers to be zero, thus maintaining the same low-level visual features as that used in the openpose model. In this way, we can purely compare the capability of the networks in capturing spatial dependencies.

The network from the first stage produces a set of keypoint heatmaps $\mathbf{S}^1 = \rho^1(\mathbf{F})$ and a set of PAFs $\mathbf{L}^1 = \phi^1(\mathbf{F})$, where ρ^1 and ϕ^1 represent high-dimensional functions of the KP path and AP path networks. To guide the network to iteratively predict keypoint heatmaps and PAFs at each stage, we apply two loss functions in each sub-network. We use an L_2 loss between the estimated predictions and the groundtruth maps and fields. Specifically, the loss functions for the dual paths at stage t are:

$$f_{\mathbf{S}}^t = \sum_{j=1}^{J} \sum_{\mathbf{p}} \|\mathbf{S}_j^t(\mathbf{p}) - \mathbf{S}_j^*(\mathbf{p})\|_2^2, \qquad (1)$$

$$f_{\mathbf{L}}^t = \sum_{c=1}^{C} \sum_{\mathbf{p}} \|\mathbf{L}_c^t(\mathbf{p}) - \mathbf{L}_c^*(\mathbf{p})\|_2^2, \qquad (2)$$

where $\mathbf{S}_j^*$ is the groundtruth keypoint heatmap, $\mathbf{L}_c^*$ is the groundtruth PAF vector field, at an image location $\mathbf{p}$.

4. Experimental Results

Dataset The PoseTrack [3] dataset consists of over $68,000$ frames. The workshop is organized around a challenge with three competition tracks focusing on single frame multi-person pose estimation, multi-person pose estimation in videos, and multi-person articulated tracking. In our work, we focus on the single frame multi-person pose estimation.

Experimental Settings In order to make a fair comparison with the openpose network, which is trained on the MS COCO dataset [1], we use the same training data before testing on the PoseTrack dataset. In our experiments, all the experiment settings including the testing scales and parameters in the data association techniques are uniform for the two networks. Therefore, no special tuning on the training and testing for the PoseTrack dataset is made.

Quantitative Results We report our Average Precision (AP) scores on the PoseTrack test set[1]. The result on test set is performed at 3 stages and 2 scales (1, 0.75).

Method	Head	Sho.	Elb.	Wri.	Hip	Knee	Ank.	Total
Ours	48.2	75.4	68.8	59.5	63.6	60.1	53.9	62.4

Table 1. Average Precision (AP) scores on the PoseTrack test set.

4.1. Algorithm Performance Analysis

Average Precision We compare the AP scores of the proposed network with openpose on the validation set. All experiments are performed on the local machine with the same resolution and single scale.

Method	Head	Sho.	Elb.	Wri.	Hip	Knee	Ank.	Total
openpose@6stages	46.8	76.4	68.7	54.7	63.6	59.6	52.8	59.5
openpose@3stages	45.5	72.1	63.1	48.1	58.4	51.9	45.8	54.4
Ours@3stages	47.5	76.3	67.6	53.3	62.9	57.9	49.7	58.5

Table 2. Comparisons of Average Precision (AP) scores on the PoseTrack validation set. Experiments are performed at the same original resolution and single scale.

Speed and Model Size Comparison We test and compare the networks by averaged forward time for a single frame. The unit is miliseconds (ms). Both experiments are performed at the same original resolution and single scale.

Method	forward time (ms)
openpose@6stages	155.8
Ours@3stages	186.6

Table 3. Comparisons of feedforward time in miliseconds (ms) of different networks for a single frame. Evaluations are performed with a single Pascal TITAN X GPU.

Method	3 stages	4 stages	5 stages	6 stages
openpose	103.8	139.0	174.1	209.3
Ours	43.7	50.1	56.4	62.7

Table 4. Comparisons of model size of different networks in Mega Bytes (MB). Both models have the same input image resolution and share the same VGG-16 layers.

Even though our model is much smaller than the openpose model, the intermediate storage of network strucures including accumulated feature maps from AP path consume GPU memory greatly in current Caffe [17] version. In the future, by porting memory-efficient denseNet implementation from other deep learning frameworks [5, 4] into Caffe, which enables more stages of DPN to fit into the GPU memory, we believe the performance will potentially be better.

5. Conclusion

In this work, we propose a dual-path network specially designed for multi-person human pose estimation, and compare our performance with the openpose[8] network in aspects of model size, forward speed, and estimation accuracy. Extentive experiments on PoseTrack challenge dataset show that our method is both accurate and efficient. Even though the method described in this work regresses PAFs[8] as the association vector, the dual-path network is generic and not limited to specific vector representation and association techniques.

References

[1] Coco keypoints challenge. http://image-net.org/challenges/ilsvrc+coco2016, 2016. 1, 2, 4

[2] Openpose library. May, 2017. 1

[3] Posetrack: Iccv workshop. https://posetrack.net/workshops/iccv2017/, October, 2017. 4

[4] M. Abadi, A. Agarwal, P. Barham, E. Brevdo, Z. Chen, C. Citro, G. S. Corrado, A. Davis, J. Dean, M. Devin, et al. Tensorflow: Large-scale machine learning on heterogeneous distributed systems. *arXiv preprint arXiv:1603.04467*, 2016. 4

[5] B. Amos and J. Z. Kolter. A pytorch implementation of densenet, 2017. 4

[6] M. Andriluka, L. Pishchulin, P. Gehler, and B. Schiele. 2d human pose estimation: New benchmark and state of the art analysis. In *CVPR*, 2014. 2

[7] V. Belagiannis and A. Zisserman. Recurrent human pose estimation. *arXiv preprint arXiv:1605.02914*, 2016. 2

[8] Z. Cao, T. Simon, S.-E. Wei, and Y. Sheikh. Realtime multi-person 2d pose estimation using part affinity fields. *arXiv preprint arXiv:1611.08050*, 2016. 1, 2, 3, 4

[9] Y. Chen, J. Li, H. Xiao, X. Jin, S. Yan, and J. Feng. Dual path networks. *arXiv preprint arXiv:1707.01629*, 2017. 1, 3

[10] M. Dantone, J. Gall, C. Leistner, and L. Van Gool. Human pose estimation using body parts dependent joint regressors. In *CVPR*, 2013. 2

[11] Y.-W. T. Hao-Shu Fang, Shuqin Xie and C. Lu. RMPE: Regional multi-person pose estimation. In *ICCV*, 2017. 1, 2

[12] K. He, G. Gkioxari, P. Dollár, and R. Girshick. Mask r-cnn. *arXiv preprint arXiv:1703.06870*, 2017. 1, 2

[13] K. He, X. Zhang, S. Ren, and J. Sun. Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 770–778, 2016. 1, 3

[14] G. Huang, Z. Liu, K. Q. Weinberger, and L. van der Maaten. Densely connected convolutional networks. *arXiv preprint arXiv:1608.06993*, 2016. 3

[15] E. Insafutdinov, L. Pishchulin, B. Andres, M. Andriluka, and B. Schiele. Deepercut: A deeper, stronger, and faster multi-person pose estimation model. In *ECCV*, 2016. 1, 2

[16] S. Ioffe and C. Szegedy. Batch normalization: Accelerating deep network training by reducing internal covariate shift. *arXiv preprint arXiv:1502.03167*, 2015. 1

[17] Y. Jia, E. Shelhamer, J. Donahue, S. Karayev, J. Long, R. Girshick, S. Guadarrama, and T. Darrell. Caffe: Convolutional architecture for fast feature embedding. In *Proceedings of the 22nd ACM international conference on Multimedia*, pages 675–678. ACM, 2014. 4

[18] S. Johnson and M. Everingham. Clustered pose and nonlinear appearance models for human pose estimation. In *BMVC*, 2010. 2

[19] L. Karlinsky and S. Ullman. Using linking features in learning non-parametric part models. In *ECCV*, 2012. 2

[20] A. Newell and J. Deng. Associative embedding: End-to-end learning for joint detection and grouping. *arXiv preprint arXiv:1611.05424*, 2016. 1, 2

[21] A. Newell, K. Yang, and J. Deng. Stacked hourglass networks for human pose estimation. In *ECCV*, 2016. 1, 2

[22] G. Papandreou, T. Zhu, N. Kanazawa, A. Toshev, J. Tompson, C. Bregler, and K. Murphy. Towards accurate multi-person pose estimation in the wild. *arXiv preprint arXiv:1701.01779*, 2017. 1, 2

[23] L. Pishchulin, M. Andriluka, P. Gehler, and B. Schiele. Poselet conditioned pictorial structures. In *CVPR*, 2013. 2

[24] S. Ren, K. He, R. Girshick, and J. Sun. Faster r-cnn: Towards real-time object detection with region proposal networks. In *Advances in neural information processing systems*, pages 91–99, 2015. 2

[25] B. Sapp and B. Taskar. Modec: Multimodal decomposable models for human pose estimation. In *CVPR*, 2013. 2

[26] K. Simonyan and A. Zisserman. Very deep convolutional networks for large-scale image recognition. *arXiv preprint arXiv:1409.1556*, 2014. 2

[27] M. Sun and S. Savarese. Articulated part-based model for joint object detection and pose estimation. In *ICCV*, 2011. 2

[28] C. Szegedy, W. Liu, Y. Jia, P. Sermanet, S. Reed, D. Anguelov, D. Erhan, V. Vanhoucke, and A. Rabinovich. Going deeper with convolutions. In *CVPR*, 2015. 1

[29] C. Szegedy, V. Vanhoucke, S. Ioffe, J. Shlens, and Z. Wojna. Rethinking the inception architecture for computer vision. In *CVPR*, 2016. 1

[30] Y. Tian, C. L. Zitnick, and S. G. Narasimhan. Exploring the spatial hierarchy of mixture models for human pose estimation. In *ECCV*, 2012. 2

[31] J. J. Tompson, A. Jain, Y. LeCun, and C. Bregler. Joint training of a convolutional network and a graphical model for human pose estimation. In *NIPS*, 2014. 1

[32] A. Toshev and C. Szegedy. Deeppose: Human pose estimation via deep neural networks. In *CVPR*, 2014. 1, 2

[33] S.-E. Wei, V. Ramakrishna, T. Kanade, and Y. Sheikh. Convolutional pose machines. In *CVPR*, 2016. 2

[34] F. Xia, P. Wang, X. Chen, and A. Yuille. Joint multi-person pose estimation and semantic part segmentation. *arXiv preprint arXiv:1708.03383*, 2017. 1, 2

[35] S. Xie, R. Girshick, P. Dollár, Z. Tu, and K. He. Aggregated residual transformations for deep neural networks. *arXiv preprint arXiv:1611.05431*, 2016. 3

Notes

[1] Performance on test set is evalutaed by the PoseTrack server. Challenge results available at: https://posetrack.net/workshops/iccv2017/posetrack-challenge-results.html