A simple yet Effective Data Augmentation for Human Pose Estimation

Tien-Dat Tran, Xuan-Thuy Vo, Ge Cao and Kang-Hyun Jo
School of Electrical Engineering, University of Ulsan
Ulsan (44610), South Korea
Email: (tdat,xthuy)@islab.ulsan.ac.kr, caoge9706@gmail.com, acejo@ulsan.ac.kr

***Abstract*—Accurate occluded key point identification is a challenge and hot topic for human pose estimation. To make the occluded or invisible keypoint better, data augmentation play an important role which makes the network overcome complex case. In this paper, we want to apply the mosaic and mix-up technique which is a powerful method to tackle the problem. Furthermore, data augmentation demonstrates its superiority over other methods without enlarging the computational cost. Correspondingly, the proposed work focuses on powerful data augmentation for occluded keypoints. First, following a human detection in the detector network, feed the human proposal region into the data augmentation, which makes the network can learn more about the occluded cases. The data after data augmentation then apply to train for the pose estimator. The estimator collects more information in occluded keypoints, illustrating higher precision efficiency. The outputs of our experiments would also demonstrate a distinction between the use of mix-up and mosaic data augmentation and existing approaches. The predicted joint heatmaps are more accurate than the baseline technique despite using the same amount of parameters due to the transition to a high-resolution network (HRNet) for the pose estimator. Regarding AP, the proposed design outperforms the baseline network which is HRNet by 1.0 points, but in the occluded case, the pose estimator performs much better. Additionally, the COCO 2017 benchmarks, now accessible as an open and the most popular dataset for pose estimation, were used to train the proposed network.**

***Index Terms*—high-resolution network, efficient attention module, human pose estimation, machine learning.**

I. Introduction

Nowadays, 2D human pose estimation plays a crucial part but challenging function in computer vision, which can serve in diverse objectives such as human robotics [1], [2], activity recognition [3], [4], human re-identification [5], [6], or film industry [7], [8]. Human pose estimation has a primary mission which is to identify body parts for human body joints.

In human pose estimation, there are many challenges that attack the network performance. Among the challenges, the occluded keypoint shown in Fig.1 is one of the biggest challenges for the network training to get better performance. To solve this kind of problem many researchers used another network such as a graph neural network [**?**] or Generative adversarial network [**?**] to generate a new structure for the human pose to train. However, utilizing a new network for the occluded problem is costly. To solve the problem, data augmentation is a potential candidate that can remedy the challenges, which is not consumed much more resources than using another network. Data augmentation does not only enhance the value of information from the image but also not consume more parameters in the training process. In more detail, the data augmentation performs a global transformation for the images. The transformation gives the network many extra points of view about images, which show a lot of improvement [9]. Besides all of the advantages, data augmentation also brings extra unimportant data, which makes the data redundant. On the other hand, many kinds of data augmentation such as crop makes the data much more margin or rotate can make the data lost information. Hence, choosing the suitable for data augmentation is really important to make the network can get better performance.

In the proposed work, we make a deep investigation into data augmentation which compares the original method and a new one. The original data augmentation [10] apply flip, rotate, scale, and half body transform. This kind of method can enhance the accuracy of keypoint however for the occluded keypoint, it shows their disadvantages which can not significantly improve the accuracy of the occluded keypoint. Hence, the proposed research applies a new kind of data augmentation which call mix-up and mosaic. By using mix-up and mosaic method, the whole architecture can gain more accuracy, especially in the case of occluded keypoint which can check at the experiment result.

In particular, the proposed study was based on a simple framework [9], which applies the top-down method for human pose estimation. Without taking the data much more different from the original but more occlude cases appear, the proposed network can be easy to learn the invisible keypoint. For instance, with the extra training data, the network may learn to connect the keypoint for the visible part such as occlude wrist or ankle keypoint. In addition, the quantity of parameters was not changed, which made the speed not increase while the accuracy for the occluded keypoint improved much more.

To make clear the mix-up and mosaic augmentation, the transformation can apply for all of the pose estimators apply the data augmentation. Also, this method is easy to apply not only for estimator but also detector

In summary, the main contribution of the paper describes in two-fold:

• We design and apply a data augmentation called the mix-up and mosaic that makes the data more information about the occluded problem.

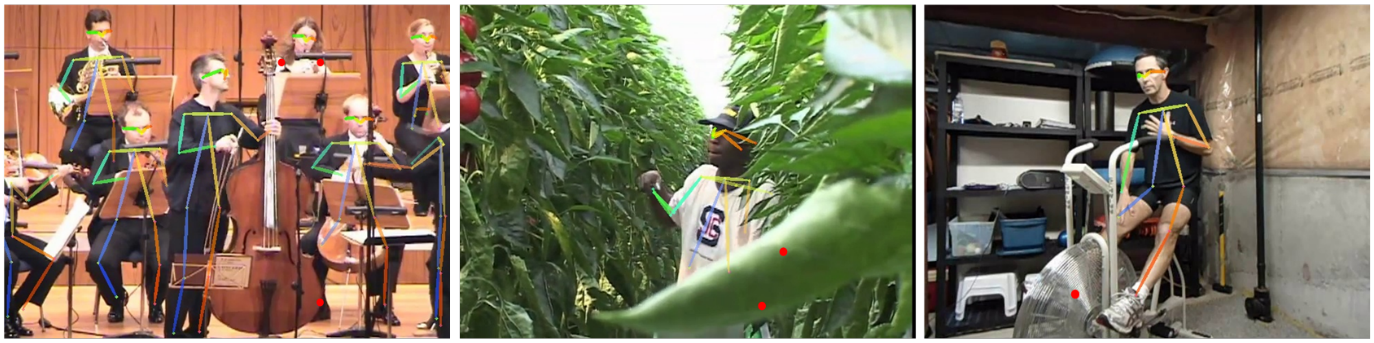

Fig. 1. Occlusion Keypoint in the testing on the MPII dataset. The red dot is the occluded keypoint which is one of the big challenges nowadays for human pose estimation

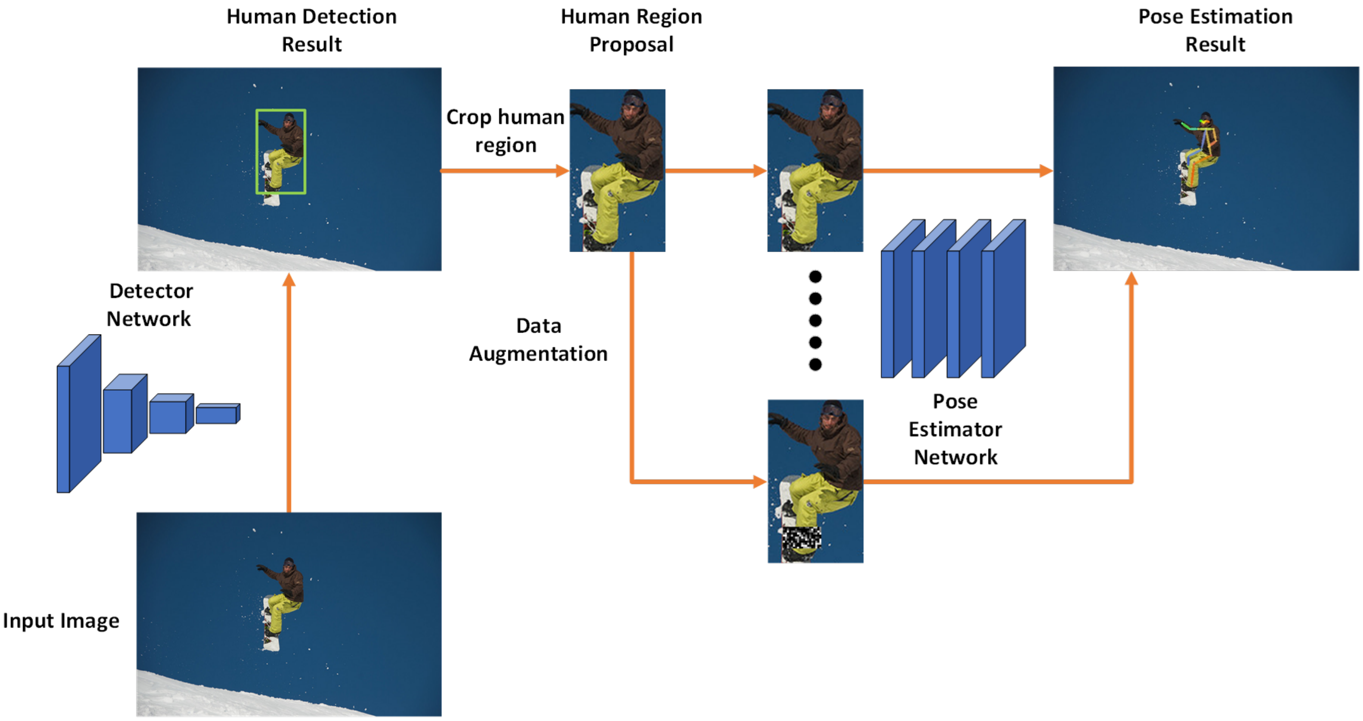

Fig. 2. Full system of 2D human pose estimation from input to pose estimation. The proposed approach split the system into 2 stages, the first stage is the human detector and the second stage is the pose estimator

- We comprehensively evaluate and compare the proposed method with the original method on the COCO benchmark dataset, which is the most popular dataset for keypoint.

II. Related work

2D-Human Pose Estimation Joint detection and its relationship to spatial space are the most crucial elements of human pose estimation, as shown in Fig. 2. The bottom-up method and the top-down method are the two basic approaches used for estimating human pose. Simple baseline uses joint prediction for the bottom-up technique, Deeppose [11], employing an end-to-end network with a higher parameter. Later, Newell minimizes the number of settings while keeping high accuracy by using the Stacked hourglass network [12]. All the approaches used Gaussian distributions to model local joints. An estimation of human posture was then performed using a convolution neural network. For the top-down method, first, we apply a detector for the human proposal region, and after that, we use the crop region for pose estimation. Because the top-down method uses the detector the accuracy can be better than the bottom-up. And bottom-up is an end-to-end method so the inference time can be better than the top-down.

In the proposed paper, we apply the top-down method for the whole architecture which is illustrated in Fig.2, From the input images, the model utilizes the existing detector for human detection. YOLO [**?**] is of diversity kind of detector, which has many versions for different cases such as real-time, high accuracy, or for mobile devices. To balance everything, the proposed method utilizes the YOLO-V3. After applying

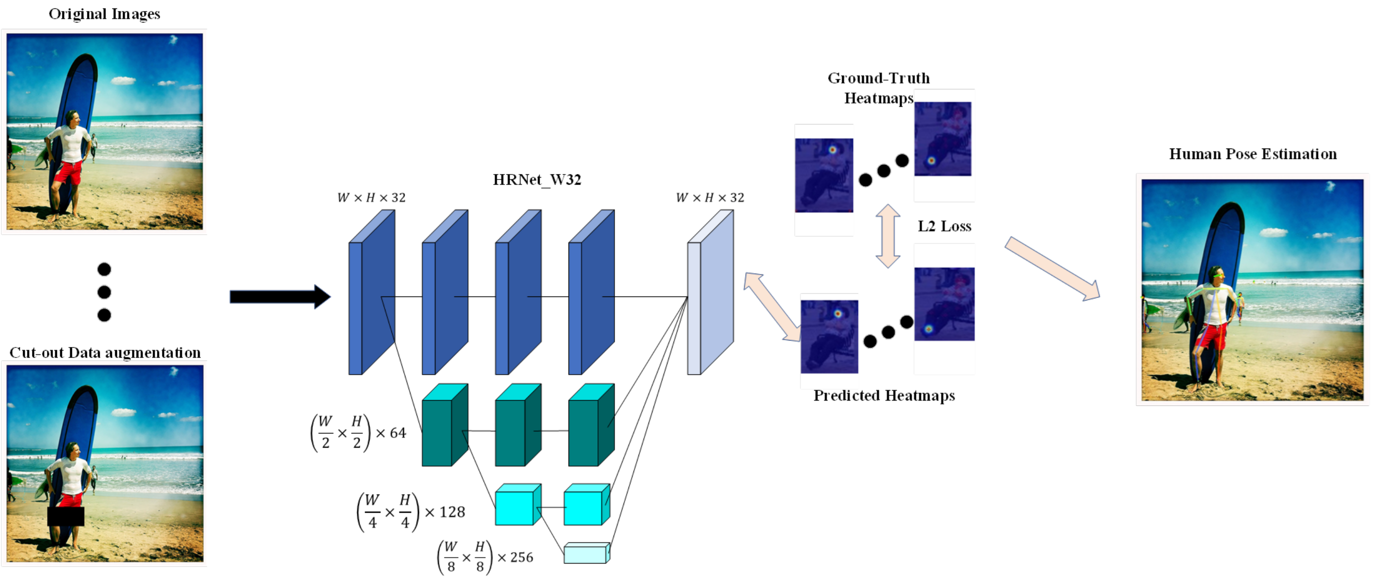
Original Images
Cut-out Data augmentation
HRNet_W32
$W \times H \times 32$
$W \times H \times 32$
$\left(\frac{W}{2} \times \frac{H}{2}\right) \times 64$
$\left(\frac{W}{4} \times \frac{H}{4}\right) \times 128$
$\left(\frac{W}{8} \times \frac{H}{8}\right) \times 256$
Ground-Truth Heatmaps
L2 Loss
Predicted Heatmaps
Human Pose Estimation

Fig. 3. Illustrating the architecture of the proposed 2D-human-pose estimator. The proposed network training with the original images and transformation images

the detector to the human region, the whole network utilizes the pose estimator to perform training tasks in the human region. Additionally, data augmentation will apply in this stage. In comparison, the top-down strategy employs enough viewpoint for implementing a network, which makes the network increase the accuracy but lose the sufficient speed

Data Augmentation: Data augmentation play an important role in computer vision task which compensate for the lack of data in real life. From the original input image, data augmentation makes more data for the network can learn from many perspective views. Notice that the more diversity in the dataset, the more accuracy for human detection. Moreover, data augmentation did not increase the number of parameters so the computational cost will not increase. However, data augmentation also can drive the detector worth [**?**] which make the detector hard to learn the feature of images, especially for occluded keypoint.

Most detector networks used the same data augmentation such as Flip, Rotation, Scale zoom in and zoom out or half body transforms with a probability of 0.3. However, this kind of data augmentation does not work well with occluded keypoint. Hence, we apply the mosaic and mix-up augmentation to show the real case to build the network can learn more about the occluded keypoint. Furthermore, the occluded and invisible keypoint appear more in the data so that the network learns better. To improve accuracy, the mix-up and mosaic method shows better performance in the data augmentation tasks.

III. Methodology

A. Network architecture

Detector The human detector plays an important role in the whole system. First, input image matrix $\alpha\left(\mathbf{X}\right)$ feed to the human detector. After that, the detector gives the result of the human region $\beta\left(\mathbf{X}'\right)$ which is the subset of $\alpha\left(\mathbf{X}\right)$.

$$D\left\{\alpha\left(\mathbf{X}\right)\right\} = \beta\left(\mathbf{X}'\right) \tag{1}$$

Following resize function make $\beta\left(\mathbf{X}'\right)$ into 256×192 images which can call $\gamma\left(\mathbf{X}'\right)$

$$\gamma\left(\mathbf{X}'\right) = Resize\left(256 \times 192, \beta\left(\mathbf{X}'\right)\right) \tag{2}$$

The proposed study utilizes YOLO-V3 [**?**] for the main detector in the whole architecture. The YOLO-V3 is the medium detector that can balance the computational cost and accuracy.

Data Augmentation In the proposed paper, we apply one more data augmentation for the bounding box $\gamma\left(\mathbf{X}'\right)$ after the detector stage. Besides the original data augmentation which includes Flip, Rotate, Scale, and Half Body Transform, additional mix-up and mosaic augmentation is applied. First, the mix-up and mosaic function can be understood

$$C\left\{\gamma\left(\mathbf{X}'\right)\right\} = Mosaic(\gamma\left(\mathbf{X}'\right), n, p) \tag{3}$$

with p is the padding fraction for data augmentation, which is the number of pixels applied. n is the number of mix-ups or mosaic pads in the human region. In the training process, we set n equal to 1. For more detail, the padding p is set random base on the size of $\gamma\left(\mathbf{X}'\right)$ which is 256×192. The human region $\gamma\left(\mathbf{X}'\right)$ have the coordinate of x_{min}, y_{min} and x_{max}, y_{max} which is the coordinate of the human region in the images. the padding P will take random with the condition $x_{min} \leq P_x \leq x_{max}$ and $y_{min} \leq P_y \leq y_{max}$.This research applies the Clamp function in Pytorch [**?**] to make the border for the mix-up and mosaic pad inside the human region $\gamma\left(\mathbf{X}'\right)$. After having the pad for mix-up and mosaic we use the replace function to apply the pad to the human region. Finally, the mix-up and mosaic image and the original pad will apply for the training part in the pose estimator

Pose estimator The pose estimator use backbone mainly HRNet-W32 and HRNet-W48 [10]. Fig.3 shows our proposed architecture for the estimator which is based on the backbone. The estimator HRNet includes 4 stages, which consist of residual blocks and connections. The input is the human region proposal from the detector resize the size to 256 × 192 for both HRNet. After that, each residual block is traversed by the feature maps, and each stage's $W \times H$ resolution is reduced twice. The size of the output tensor is finally reduced to $\frac{W}{16} \times \frac{H}{16}$ with the number of channels is 256 at the final bottom layer. The first subnetwork, whose size is $W \times H$, is the only one that the backbone network will employ during the regression. Additionally, each stage would see a doubling of the channel size. After the first block, it increases from 32 to 256 in the last layer. In order for the Training System to predict the human joints, the backbone network must gather data and feature maps from the input image by utilizing the cross entropy loss which describes in the Loss function part.

After extracting the information using the HRNet as the backbone, the feature map will be trained with Ground-truth Heat Maps. We setting heat map size is a quarter with input original images 256×192 for HRNet-W32 and 384×288 for HRNet-W48. However, we resize the input image for HRNet-W48 into 256×192 to save the parameter and time for training. For regression, the proposed study uses these predict map and the ground truth heatmap to create the corrected keypoint. This article employs HRNet so the feature maps are kept to the shape with the original input (in Fig.3). The residual block contains both batch normalization and ReLU [13].

B. Loss Function

The Proposed network utilizes Heat maps to demonstrate body Keypoint locations in whole Loss Function. In Fig. 3 the GrouthTruth coordinate by $a = \{a_k\} k = 1^K$, where $a_k = (x_k, y_k)$ is the spatial position of the kth keypoiny in the trained sample. The heat map value H_k of groundtruth is then constructed after applying the Gaussian function with variance $\sum$ and the mean a_k as shown below.

$$H_k(p) \sim N(a_k, \sum) \tag{4}$$

where $\mathbf{p} \in \mathbb{R}^\mathbf{2}$ illustrate the coordinate, and $\sum$ is experimentally defined as an identity matrix $\mathbf{I}$. In the final process of training, the network will predict K heat maps, *i.e.,* $\hat{S} = \left\{\hat{S}_k\right\} k = 1^K$ for K body joints. Mean Square error is the main Loss, which is calculated as follows:

$$L = \frac{1}{NK} \sum_{n=1}^{N} \sum_{k=1}^{K} \left\| S_k - \hat{S}_k \right\|^2 \tag{5}$$

N denotes the total of images in the training process. Using information from the backbone network's last layer, The proposed architecture generated the predicted heatmap keypoint by using the ground truth.

TABLE I
THE RESULT FOR APPLYING DIFFERENT KINDS OF DATA AUGMENTATION IN HRNET

Backbone	Data augmentation	mAP
HRNet-W32	Without	72.9
HRNet-W32	Flip	73.6
HRNet-W32	Rotate	73.4
HRNet-W32	Scale	73.3
HRNet-W32	Half body transform	73.7
HRNet-W32	Mix-Up (Our)	74.7
HRNet-W32	Mosaic (Our)	74.9
HRNet-W32	Mix-Up+Mosaic	75.1

IV. EXPERIMENTS

A. Experiment Setup

COCO datasets result Dataset. In the proposed architecture, Microsoft COCO 2017 [14] is used during the studies. The data collection contains 250K human samples and 200K images, each human identity has 17 keypoint labels. The dataset includes three folders with labels. Training folder, validation folder, and test-dev folder contain training, validation, and testing sample respectively. Hence, the original is available to view, and the training and validation set are annotated as well.

This study also made use of a commercial dataset that records footage of individuals working in a commercial laboratory setting. The dataset consists of 4 films with frame rates ranging from 4000 to 6000. There are several difficulties in the video, including overlapped people, crowded at scenes, and little people. Therefore, it is possible to test how effective the suggested strategy is at tracking.

Evaluation metrics. In the proposed study COCO [14], we utilized Object Keypoint Similarity (OKS) using $OKS = \frac{\sum_i exp(-{d_i}^2/2s^2k_i^2)\delta(v_i>0)}{\sum_i \delta(v_i>0)}$. Specifically, d_i represents the Euclidean distance between the ground truth and the predicted joint, v_i represents the visibility label flag, s represents the scale of the object, and k_i represents the keypoint for human joint. Moreover, the average recall (AR) and average precision (AP) scores are calculated. Table II shows the AR and AP averages from OKS=0.5 to OKS=0.95, with AP^L for the large objects and AP^M standing for medium objects.

Implementation details The proposed experiments are conducted using the AlphaPose [9] codebase and implemented on two poplular datasets. The resolution of the image input was lowered to 256×192. The model was trained on a single NVIDIA GTX 1080Ti GPU with CUDA 11.0 and CuDNN 7.5. The strategy involved data augmentation in model training, such as flipping, rotating at 40 degrees by design, and scaling with the factor set to 0.3. When utilizing training photographs, set the number of batchsize is 4 and shuffle is utilized. There are 210 total epochs in whole training process, and the base learning rate is set at 0.001 before being decreased by 10 (learning rate factor = 0.1) at the $170th$ and $200th$ epochs. We set momentum is 0.9 base on the Adam optimizer [19].

TABLE II
COMPARISON ON COCO VALIDATION DATASET

Method	Input size	Backbone	#Params	AP	AP^{50}	AP^{75}	AP^{M}	AP^{L}	AR
8-Stage Hourglass [12]	256×192	8-Stage Hourglass	25.1M	66.9	-	-	-	-	-
Mask-RCNN [15]	256×192	ResNet-50-FPN	-	63.1	87.3	68.7	57.8	71.4	-
OpenPose [16]	-	-	-	61.8	84.9	67.5	57.1	68.2	66.5
PersonLab [17]	-	-	-	78.7	89.0	75.4	64.1	75.5	75.4
SimpleBaseline [18]	256×192	ResNet-50	34.0M	70.4	88.6	78.3	67.1	77.2	76.3
SimpleBaseline [18]	256×192	ResNet-101	53.0M	71.4	89.3	79.3	68.1	78.1	77.1
SimpleBaseline [18]	256×192	ResNet-152	68.6M	73.7	91.9	81.1	70.3	80.0	79.0
HRNetBaseline [10]	256×192	HRNet-W32	28.5M	74.4	90.5	81.9	70.8	81.0	79.8
HRNetBaseline [10]	256×192	HRNet-W48	63.6M	75.1	90.6	82.2	71.5	81.8	80.4
HRNet + our Data Augmentation	256×192	HRNet-W32	28.5M	75.1	90.5	82.3	71.3	82.1	80.3
HRNet + our Data Augmentation	256×192	HRNet-W48	63.6M	75.9	91.0	82.3	72.1	82.3	81.2

B. Experiment Result

The proposed method experiments with each circumstance while implementing different kinds of data augmentation for the pose estimator. In Tab.1, The accuracy illustrates that using the proposed method for Mix-up and Mosaic gains 1.8 and 2.0 in mAP (mean Average Precision), which enhances the AP increase by around 2 percent. Furthermore, this study also investigates again another data augmentation [14], which is set up in almost a training process for pose estimator. The default data augmentation including Flip, Rotation, Scale, and Half body transform is trained again separately and shown in Tab.1. In total, when combining all of the data augmentations and applying the mix-up and mosaic the AP increases with 2.2 AP.

The suggested result in Table 2 was approximated using the COCO validation dataset. In all instances, the accuracy in the our technique is larger than the Benchmark High-Resolution Network of 0.2 AP in both backbone HRNet-32 and HRNet-W48. In addition, the average recall (AR) for HRNet-W32 is 0.3 points higher and 0.2 points higher for HRNet-W48. Overall, the experiment outcomes improved modestly in both AP and AR, but dramatically in the event of an obscured keypoint.

V. CONCLUSION

This research shows the effect of the data augmentation on CNNs especially for occluded human keypoint, focusing on mosaic and mix-up for human proposals. Furthermore, our work demonstrates that not increasing the computation cost, the data augmentation utilized has a more considerable effect. Moreover, the mosaic and mix-up focused more on the essential feature map than the other element. The network will become more effective as a consequence, particularly for various computer vision-related tasks.

Besides, human pose estimation has several problems that need to be solved. First, the occluded joints were challenging to train and predict for the architecture. Second, human key points appear in the low-resolution images. The next issue is the sample has a crowd, which is usually difficult to identify where each participant's joint location. Last but not least, The lacking of data with partial body part appear with human posture. The proposed method tries to solve the first problem is also the most complex case compared to all of the issues. Hence, future research will try to focus on the remaining problem and also try to apply the technique to other state-of-the-art pose estimators.

ACKNOWLEDGEMENT

This results was supported by "Regional Innovation Strategy (RIS)" through the National Research Foundation of Korea(NRF) funded by the Ministry of Education(MOE) (2021RIS-003)

REFERENCES

[1] S.-E. Wei, V. Ramakrishna, T. Kanade, and Y. Sheikh, "Convolutional pose machines," 2016.

[2] C.-J. Chou, J.-T. Chien, and H.-T. Chen, "Self adversarial training for human pose estimation," 2017.

[3] Z. Hussain, M. Sheng, and W. E. Zhang, "Different approaches for human activity recognition: A survey," 2019.

[4] E. Kim, S. Helal, and D. Cook, "Human activity recognition and pattern discovery," *IEEE Pervasive Computing*, vol. 9, no. 1, pp. 48–53, Jan 2010.

[5] X. Yang, M. Wang, and D. Tao, "Person re-identification with metric learning using privileged information," *CoRR*, vol. abs/1904.05005, 2019. [Online]. Available: http://arxiv.org/abs/1904.05005

[6] W. Li, R. Zhao, and X. Wang, "Human reidentification with transferred metric learning," in *Asian Conference on Computer Vision (ACCV)*, 11 2012, pp. 31–44.

[7] C. Chen and D. Ramanan, "3d human pose estimation = 2d pose estimation + matching," in *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, July 2017, pp. 5759–5767.

[8] S. Li, L. Ke, K. Pratama, Y. Tai, C. Tang, and K. Cheng, "Cascaded deep monocular 3d human pose estimation with evolutionary training data," *CoRR*, vol. abs/2006.07778, 2020. [Online]. Available: https://arxiv.org/abs/2006.07778

[9] H.-S. Fang, J. Li, H. Tang, C. Xu, H. Zhu, Y. Xiu, Y.-L. Li, and C. Lu, "Alphapose: Whole-body regional multi-person pose estimation and tracking in real-time," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2022.

[10] K. Sun, B. Xiao, D. Liu, and J. Wang, "Deep high-resolution representation learning for human pose estimation," 2019.

[11] A. Toshev and C. Szegedy, "Deeppose: Human pose estimation via deep neural networks," *CoRR*, vol. abs/1312.4659, 2013. [Online]. Available: http://arxiv.org/abs/1312.4659

[12] A. Newell, K. Yang, and J. Deng, "Stacked hourglass networks for human pose estimation," *CoRR*, vol. abs/1603.06937, 2016. [Online]. Available: http://arxiv.org/abs/1603.06937

[13] S. Ioffe and C. Szegedy, "Batch normalization: Accelerating deep network training by reducing internal covariate shift," 2015.

[14] T. Lin, M. Maire, S. J. Belongie, L. D. Bourdev, R. B. Girshick, J. Hays, P. Perona, D. Ramanan, P. Dollár, and C. L. Zitnick, "Microsoft COCO: common objects in context," *CoRR*, vol. abs/1405.0312, 2014. [Online]. Available: http://arxiv.org/abs/1405.0312

[15] K. He, G. Gkioxari, P. Dollár, and R. Girshick, "Mask r-cnn," 2017.

[16] Z. Cao, G. Hidalgo Martinez, T. Simon, S. Wei, and Y. A. Sheikh, "Openpose: Realtime multi-person 2d pose estimation using part affinity fields," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2019.

[17] G. Papandreou, T. Zhu, L.-C. Chen, S. Gidaris, J. Tompson, and K. Murphy, "Personlab: Person pose estimation and instance segmentation with a bottom-up, part-based, geometric embedding model," 2018.

[18] B. Xiao, H. Wu, and Y. Wei, "Simple baselines for human pose estimation and tracking," *CoRR*, vol. abs/1804.06208, 2018. [Online]. Available: http://arxiv.org/abs/1804.06208

[19] D. Kingma and J. Ba, "Adam: A method for stochastic optimization," *International Conference on Learning Representations*, 12 2014.