Journal of Korea Multimedia Society Vol. 24, No. 5, May 2021(pp. 659-666)
https://doi.org/10.9717/kmms.2021.24.5.659

Multi-Human Behavior Recognition Based on Improved Posture Estimation Model

Zhang Ning[†], Park Jin-Ho[††], Lee Eung-Joo[†††]

ABSTRACT

With the continuous development of deep learning, human behavior recognition algorithms have achieved good results. However, in a multi-person recognition environment, the complex behavior environment poses a great challenge to the efficiency of recognition. To this end, this paper proposes a multi-person pose estimation model. First of all, the human detectors in the top-down framework mostly use the two-stage target detection model, which runs slow down. The single-stage YOLOv3 target detection model is used to effectively improve the running speed and the generalization of the model. Depth separable convolution, which further improves the speed of target detection and improves the model's ability to extract target proposed regions; Secondly, based on the feature pyramid network combined with context semantic information in the pose estimation model, the OHEM algorithm is used to solve difficult key point detection problems, and the accuracy of multi-person pose estimation is improved; Finally, the Euclidean distance is used to calculate the spatial distance between key points, to determine the similarity of postures in the frame, and to eliminate redundant postures.

Key words: Multi-Human Behavior, Posture Estimation, S(D)TN, YOLOv3, OHEM

1. INTRODUCTION

The initial attitude estimation algorithms can be roughly divided into two categories. The first type is to directly solve the pose estimation problem as a classification or regression problem through a global feature[1,2]. The second category is based on a graphical model, such as a commonly used pictorial structure model. There are many subsequent improvements, either in how to extract a better feature representation[3,4], or in modeling a better spatial position relationship[5,6]. Starting from AlexNet in 2012, deep learning began to develop rapidly, from the earliest image classification problem to the later detection and segmentation problems. In 2014, [7] successfully introduced CNN for the first time to solve the problem of single-person pose estimation. Until 2016, with the explosion of deep learning, the problem of pose estimation also attracted prime time. Here we need to focus on two tasks, one is Convolutional Pose Machine (CPM)[8], and the other is Hourglass[9]. After Hourglass, there are also many good works that continue to optimize the single-person pose estimation algorithm, such as [10,11]. In the 2016 COCO competition, the first place at the time was OpenPose[12]. Based on CPM as the component, the CMU team first finds the position of each joint in the picture, and then proposes Part Affinity Field (PAF) to assemble the human body. On the list of

※ Corresponding Author : Lee Eung-Joo, Address: (48520) 428, Sinseon-ro, Nam-gu, Busan, Korea, TEL : +82-51-629-1143, FAX : +82-51-629-1143, E-mail : ejlee@tu.ac.kr
Receipt date : May 20, 2021, Approval date : May 27, 2021

[†] Dept. of Information and Communication Engineering, Tongmyong University, Busan, Korea, Dept. of Duoyuan Technology Co., Ltd, Shenyang, China
(E-mail : jangneyong0829@hotmail.com)
[†††] Dept. of Information and Communication Engineering, Tongmyong University, Busan, Korea
(E-mail : jinobak@hanmail.net)
[†††] Dept. of Information and Communication Engineering, Tongmyong University, Busan, Korea

the 2016 competition, another very important job is the Associative Embedding of the Deng Jia group[13]. In addition to Openpose and Associative Embedding, bottom-up also has a very good job, DeepCut[14] and DeeperCut[15], they use optimization problems to directly optimize the combination of solvers. Simple Baselines[16] is Xiao Bin's work at MSRA. A very concise structure is proposed which can be used for multi-person pose estimation and human body pose estimation tracking problems.

2. IMPROVED POSTURE ESTIMATION MODEL

2.1 Model Structure

The improved multi-human pose estimation model proposed in this paper is based on a top-down framework, and its overall structure is shown in Fig. 1. The model mainly includes four parts: human body detector, STN network and SDTN network, attitude estimation network and attitude back propagation network, and attitude non-maximum suppression network (PoseNMS). First of all, the human detectors in the top-down framework mostly use the two-stage target detection model, which runs slow down. The single-stage YOLOv3 target detection model is used to effectively improve the running speed and the generalization of the model. Depth separable convolution, which further improves the speed of target detection and improves the model's ability to extract target proposed regions; Secondly, based on the feature pyramid network combined with context semantic information in the pose estimation model, the OHEM algorithm is used to solve difficult key point detection problems, and the accuracy of multi-person pose estimation is improved; Finally, the Euclidean distance is used to calculate the spatial distance between key points, to determine the similarity of postures in the frame, and to eliminate redundant postures.

2.2 Space (de-Transformation) Transformation Network (S(D)TN)

The spatial transformation network (STN) gives traditional convolutional cropping, translation, scaling, and rotation characteristics to make the model spatially invariant. It can adaptively transform and align the data to extract a high-quality human body frame. The spatial de-transformation network (SDTN) is used to inversely map the pose estimation results to the original image coordinates. The symmetric spatial transformation network consists of STN and SDTN, and its network structure is shown in Fig. 2, where Θ represents the spatial transformation parameters, λ represents the spatial de-transformation parameter, $T_\theta(G)$ represents the 2D affine transformation function, and $T_\lambda(G)$ represents the 2D de-transformation function. Mathematically, spatial transformation refers to the affine transformation of the corresponding matrix. The affine transformation of the image can be expressed as

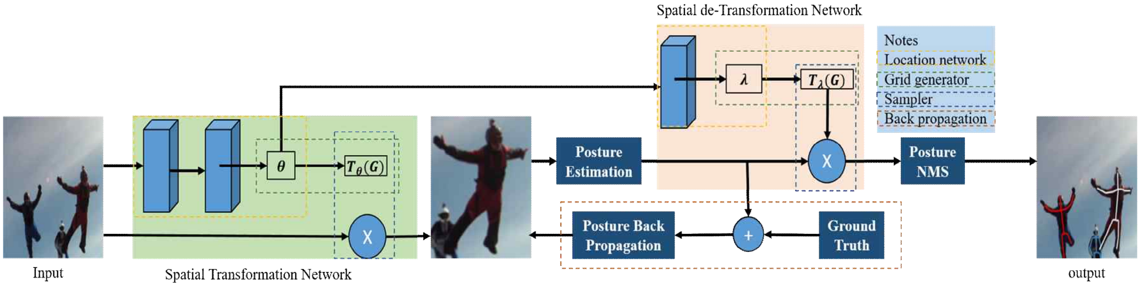

Fig. 1. Overall Structure of Improved multi-human estimation model.

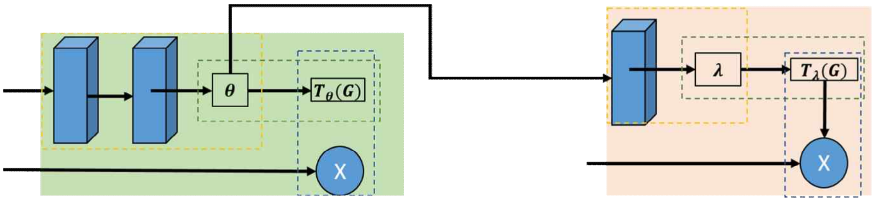
θ
$T_\theta(G)$
X
Spatial Transformation Network
λ
$T_\lambda(G)$
X
Spatial de-Transformation Network

Fig. 2. Overall Structure of Space (de-Transformation) Transformation Network.

$$\begin{bmatrix} x_i^{(s)} \\ y_i^{(s)} \end{bmatrix} = \begin{bmatrix} \theta_{11} & \theta_{12} & \theta_{13} \\ \theta_{21} & \theta_{22} & \theta_{23} \end{bmatrix} \begin{bmatrix} x_i^{(T)} \\ y_i^{(T)} \\ 1 \end{bmatrix} = [\theta_1 \ \theta_2 \ \theta_3] \begin{bmatrix} x_i^{(T)} \\ y_i^{(T)} \\ 1 \end{bmatrix} \tag{1}$$

where, $\theta \rightarrow (\theta_1 \ \theta_2 \ \theta_3)$ is a two-dimensional space vector in the real domain; $(x_i^{(s)}, y_i^{(s)})$ represents the ith coordinate point of the original image; S is the original image before conversion; $(x_i^{(T)}, y_i^{(T)})$ represents the i-th coordinate point of the image after affine transformation, T represents the converted image. First, the affine transformation coefficient Θ representing the coordinate mapping relationship is generated according to the positioning network, the corresponding input point coordinates are calculated, and then the pixel values are filled in the sampler.

$$V_i = \sum_n \sum_m U_{nm} * k(x_i^{(s)} - m; \phi_x) * k(y_i^{(s)} - n; \phi_y) \tag{2}$$

where, n and m will traverse all coordinate points in the original image; U_{nm} refers to the pixel value of the point (n, m) in the original image channel; V_i is the pixel value of the i-th coordinate point; k(•) is a linear interpolation function; ϕ_x and ϕ_y are the parameters of the interpolation function; x and y respectively represent the coordinates of the i-th coordinate point of the original image; ∗ indicates convolution. When the function k(•) uses bilinear interpolation, the filling function becomes.

$$V_i = \sum_n \sum_m U_{nm} * \max(0, 1 - |x_i^{(s)} - m|) * \max(0, 1 - |y_i^{(s)} - n|) \tag{3}$$

The SDTN network is de-transform of the STN network. The parameter λ used to calculate the de-transform can be obtained by the parameter $\theta \rightarrow (\theta_1 \ \theta_2 \ \theta_3)$, and its expression is

$$\begin{bmatrix} x_i^{(T)} \\ y_i^{(T)} \end{bmatrix} = [\lambda_1 \ \lambda_2 \ \lambda_3] \begin{bmatrix} x_i^{(s)} \\ y_i^{(s)} \\ 1 \end{bmatrix} \tag{4}$$

where, $\lambda \rightarrow (\lambda_1 \ \lambda_2 \ \lambda_3)$ is a two-dimensional space vector in the real domain; $(x_i^{(s)}, y_i^{(s)})$ and $(x_i^{(T)}, y_i^{(T)})$ represent the pixels of the original image and the pixels of the image after de-transformation. The relationship of $\lambda_1, \lambda_2, \lambda_3$ as follow:

$$[\lambda_1 \ \lambda_2] = [\theta_1 \ \theta_2]^{-1} \tag{5}$$

$$\lambda_3 = [-\lambda_1 \ \lambda_2]\theta_3 \tag{6}$$

In order to carry out back propagation in SDTN network, $\frac{\partial J(W,b)}{\partial \theta}$ can be decomposed into

$$\frac{\partial J(W,b)}{\partial [\theta_1 \ \theta_2]} = \frac{\partial J(W,b)}{\partial [\lambda_1 \ \lambda_2]} \times \frac{\partial [\lambda_1 \ \lambda_2]}{\partial [\theta_1 \ \theta_2]} + \frac{\partial J(W,b)}{\partial \lambda_3} \times \frac{\partial \lambda_3}{\partial [\lambda_1 \ \lambda_2]} \times \frac{\partial [\lambda_1 \ \lambda_2]}{\partial [\theta_1 \ \theta_2]} \tag{7}$$

$$\frac{\partial J(W,b)}{\partial \theta_3} = \frac{\partial J(W,b)}{\partial \lambda_3} \times \frac{\partial \lambda_3}{\partial \theta_3} \tag{8}$$

where, $J(W,b)$ is the cost function of the symmetric STN model; W and b are both parameter matrices.

2.3 Separable Spatial Convolution

The YOLOv3 model draws on the feature pyramid network and uses logistic regression instead of the Softmax function as the classifier, and the target detection speed is greatly improved. Based on the YOLOv3 target detection model, the depth separable convolution is added to the image convolution process, which can effectively reduce the parameter size of the model and improve the target detection speed.

In standard convolution, each input channel is convolved with a specific convolution kernel, and the sum of the convolution results from all channels is used as the final result. In depth separable convolution, depth convolution is first performed, and each input channel is separately convolved, and then convolution is performed point by point. Compared with standard convolution, this convolution structure can greatly reduce the number of parameters and calculations of the network model, and will not cause significant accuracy loss. For example, the traditional convolution kernel is to convolve 3 channels at the same time, that is, three channels output a convolution value after one convolution. The depth separable convolution is to use three convolution kernels to convolve three channels separately, in this way, 3 convolution values are output after one convolution, then pass a 1×1×3 convolution kernel to get the final convolution value. As the extracted attributes increase, deep separable convolution can save more parameters and reduce the amount of model calculation.

When the size of the input image is $M\times M\times N$, the size of the convolution kernel is $K\times K\times N\times P$, and the step size is 1, the size of the parameters required by the standard convolution, PSC, and the calculation amount of the convolution operation CSC are

$$P_{SC} = K\times K\times N\times P \tag{9}$$

$$C_{SC} = M\times M\times K\times K\times N\times P \tag{10}$$

The size P_{DSC} of the parameters required for deep separable convolution and the calculation amount C_{DSC} of the convolution operation are respectively

$$P_{DSC} = K\times K\times N + N\times P \tag{11}$$

$$C_{DSC} = M\times M\times K\times K\times N + M\times M\times N\times P \tag{12}$$

The definition of its parameter scale change P_C and reduction rate P_{CR} is defined as

$$P_C = \frac{P_{DSC}}{P_{SC}} = \frac{1}{P} + \frac{1}{K^2} \tag{13}$$

$$P_{CR} = \frac{|P_{SDC} - P_{SC}|}{P_{SC}} = \frac{|K^2\times P - K^2 - P|}{K^2 + P} \tag{14}$$

The calculation method of the amount of change in the convolution operation and the calculation method of the reduction rate are the same as (13) and (14).

2.4 Posture Estimation Network

According to the different difficulty levels of human key point detection, combined with online hard-example mining algorithms to detect key

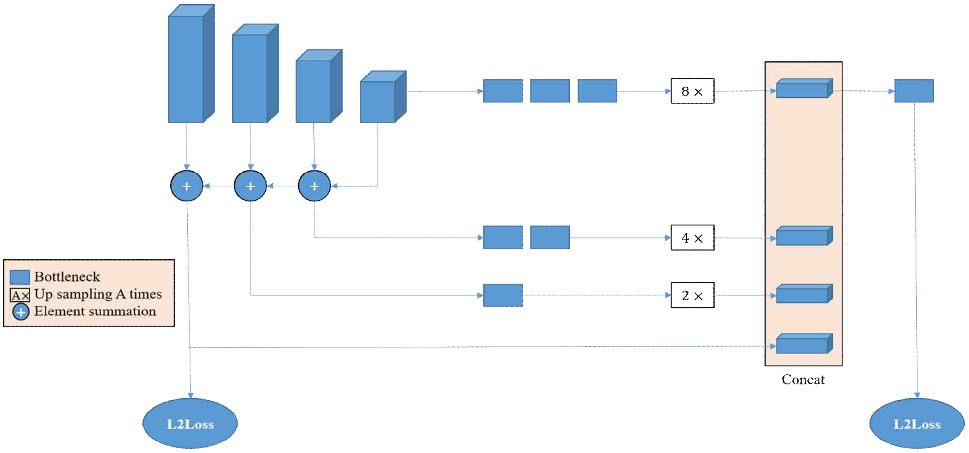

Fig. 3. Attitude estimation network model.

points. First, based on the feature pyramid network, locate key points that are easy to identify, such as head, shoulders, and elbows. Combined with contextual semantic information to locate difficult key points such as ankles, wrists, buttocks, etc. Finally, the key points of the entire human body are detected. The loss function uses the mean square error (MSE, L2Loss), and the network structure is shown in Fig. 3.

3. EXPERIMENT RESULTS AND ANALYSIS

Taking four typical images of campus acquisition images, network images, and 2017 COCO dataset images as examples, multi-person pose estimation is performed to show the generalization ability of the model under four scenarios of scale change, dense crowd, occlusion, and complex poses. The effect of model pose estimation is shown in Fig. 4.

ResNet-18 is used as the positioning network in the spatial transformation network, and the pose estimation network is built on the basis of the feature pyramid network. Add multiple Bottleneck modules to the network design, merge the features of different layers, and add contextual semantic information to achieve difficult key point detection. In model training, the picture is cropped with an aspect ratio of 384:288, and a random flip strategy is used to randomly rotate the picture $(-45^\circ \sim +45^\circ)$ and change the image scale. The scale changes are three different scales: 0.7, 1, 1.35. The model training data set is the 2017MS COCO data set, including 57×103 images and 150×103 human body instances. The Adam algorithm is used during the training process to iteratively update the network weights. After every 3.6 million iterations, the learning rate is reduced by 0.5, and the initial learning rate is $5\times10-4$.

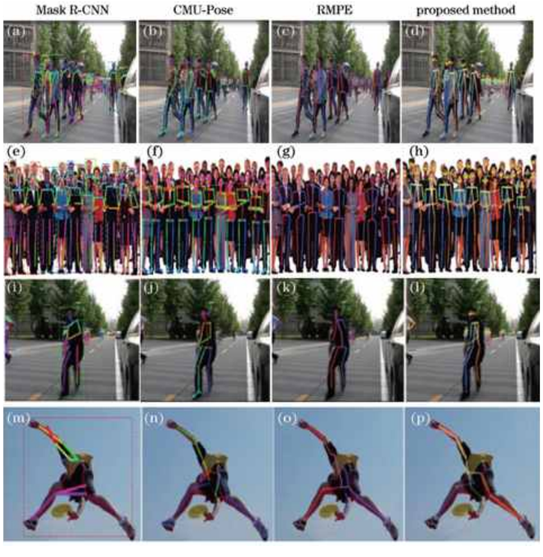

Fig. 4. Comparison of results in different scenarios for each model.

In the MSCOCO evaluation index, the definition of object key point similarity (OKS) is:

$$R_{OKS} = \frac{\sum_i \exp(-d_{ip}^2/2S_p^2\sigma_i^2)\delta(v_{ip}=1)}{\sum_i \delta(v_{ip}=1)} \qquad (16)$$

where, p is the id of the person in the ground truth; i is the id of the key point; d_{ip} is the Euclidean distance between each person's key point and the predicted key point in the ground truth; S_p is the scale factor of the current person, that is, the person is on the ground The square root of the area occupied in the live situation; σ_i represents the normalization factor of the i-th key point; v_{ip} represents whether the i-th key point of the p-th person is visible; δ is a function that selects the visible point for calculation. AP is the average accuracy rate of all 10 OKS thresholds, AR is the average recall rate of all 10 OKS thresholds. $AP_{@0.5}$ indicates the AP value when the OKS is 0.5, $AP_{@0.75}$ indicates the AP value when the OKS is 0.75, AP_m indicates the medium target AP value, and the area size is (322,962), AP_1 represents the large target AP value, the area size range is (962, $+\infty$), and the AR parameter has the same meaning as AP.

The performance of this design model and the current leading attitude estimation model in the 2017 COCO Test-dev data set is shown in Table 1.

Table 1. Comparison of performance of each pose estimation (%).

Model	AP	$AP_{@0.5}$	$AP_{@0.75}$	AP_{m}	AP_{1}
CMU-Pose[17]	61.8	84.9	67.5	57.1	68.2
Mask R-CNN[18]	63.1	87.3	68.7	57.8	71.4
RMPE[19]	72.3	89.2	79.1	68.0	78.6
Improved	74.1	92.5	80.5	70.6	79.5

Table 2. Comparison of parameters of each human detection algorithm.

Model	Data Set	Running Speed (frame/s)	Parameter Size (MB)	Calculated amount/10^9
YOLOv3[20]	MS COCO	51	237	65.86
Improved	MS COCO	64	195	44.32

Table 3. Comparison of performance of each pose estimation (%).

Imput	AP	$AP_{@0.5}$	$AP_{@0.75}$	AP_{m}	AP_{1}	AR	$AR_{@0.5}$	$AR_{@0.75}$	AR_{m}	AR_{1}
256×192	71.2	91.4	78.3	68.5	75.2	74.3	92.2	80.9	71.3	78.9
384×288	74.1	92.5	80.5	70.6	79.5	76.8	93.2	82.5	73.0	82.6

In this paper, a random flip and rotation strategy is used in data preprocessing, and three different image scales are generated, which not only enlarges the data scale, but also makes the data have good scale invariance and rotation invariance. The integration of target features of different layers in the design of the network structure and the addition of contextual semantic information are conducive to improving the detection accuracy of difficult key points. According to the evaluation indicators of the 2017 COCO Test-dev data set, the detection accuracy of the key points of the pose estimation model in this paper has been improved. The average detection accuracy has been improved by 14.84% compared with the Mask R-CNN model and 2.43% compared with the RMPE model.

1) Scale analysis of human body detector parameters

This article analyzes and compares the parameter scale of the proposed model and the YOLOv3 target detection model. The data is shown in Table 2.

It can be seen from Table 2 that after the deep separable convolution operation, the parameter scale is reduced by about 17.72%, the calculation amount is reduced by about 32.71%, and the frame frequency is 64 frame/s.

2) P-R performance analysis of attitude estimation model

In this paper, the AP and AR values are shown in Table 3 when the input image size is 256 pixel × 192 pixel and 384 pixel × 288 pixel.

4. CONCLUSION

The multi-person pose estimation model proposed in this paper effectively improves the key point detection speed of the multi-person pose estimation model by adding a depth separable convolution algorithm to the single-stage YOLOv3 target detection algorithm. And through the online hard example mining algorithm combined with contextual semantic information, the accuracy of key point detection is effectively improved. On the COCO Test-dev dataset in 2017, the detection accuracy of key points in the model designed in this paper is increased by 14.84% compared to Mask R-CNN, and 2.43% higher than the RMPE model. In the experimental environment of this article, the real-time running speed of the multi-person pose estimation network model reaches 22 frames per second.

REFERENCE

[1] G. Rogez, J. Rihan, S. Ramalingam, C. Orrite, and P.H.S. Torr, "Randomized Trees for Human Pose Detection," *IEEE Conference on Computer Vision and Pattern Recognition*, pp. 1-8, 2008.

[2] R. Urtasun and T. Darrell, "Local Probabilistic Regression for Activity-Independent Human Pose Inference," *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 1-8, 2008.

[3] L. Pishchulin, M. Andriluka, P. Gehler, and B. Schiele, "Strong Appearance and Expressive Spatial Models for Human Pose Estimation," *Proceedings of the IEEE International Conference on Computer Vision (ICCV)*, pp. 3487-3494, 2013.

[4] M. Andriluka, S. Roth, and B. Schiele, "Pictorial Structures Revisited: People Detection and Articulated Pose Estimation," *IEEE Conference on Computer Vision and Pattern Recognition*, pp. 1014-1021, 2009.

[5] C. Lonescu, L. Fuxin, and C. Sminchisescu, "Latent Structured Models for Human Pose Estimation," *2011 International Conference on Computer Vision*, pp. 2220-2227, 2011.

[6] L. Pishchulin, M. Andriluka, P. Gehler, and B. Schiele, "Poselet Conditioned Pictorial Structures," *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 588-595, 2013.

[7] A. Jain, J. Tompson, M. Andriluka, G.W. Taylor, and C. Bregler, "Learning Human Pose Estimation Features with Convolutional Networks," *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 1-11, 2014.

[8] W. Shih-En, V. Ramakrishna, T. Kanade, and Y. Sheikh, "Convolutional Pose Machines," *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 4724-4732, 2016.

[9] A. Newell, Y. Kaiyu, and D. Jia, "Stacked Hourglass Networks for Human Pose Estimation," *European Conference on Computer Vision*, pp. 483-499, 2016.

[10] C. Xiao, Y. Wei, O.Y. Wanli, M. Cheng, A.L. Yuille, and W. Xiaogang, "Multi-Context Attention for Human Pose Estimation," *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 1831-1840, 2017.

[11] T. Wei, Y. Pei, and W. Ying, "Deeply Learned Compositional Models for Human Pose Estimation," *Proceedings of the European Conference on Computer Vision (ECCV)*, pp. 190-206, 2018.

[12] C. Zhe, T. Simon, W. Shih-En, and Y. Sheikh, "Realtime Multi-Person 2D Pose Estimation using Part Affinity Fields," *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 7291-7299, 2017.

[13] A. Newell, H. Zhiao, and D. Jia, "Associative Embedding: End-to-End Learning for Joint Detection and Grouping," *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 2274-2284, 2017.

[14] L. Pishchulin, E. Insafutdinov, T. Siyu, B. Andres, M. Andriluka, P.V. Gehler, and B. Schiele, "DeepCut: Joint Subset Partition and Labeling for Multi Person Pose Estimation," *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 4929-4937, 2016.

[15] E. Insafutdinov, L. Pishchulin, B. Andres, M. Andriluka, and B. Schiele, "DeeperCut: A Deeper, Stronger, and Faster Multi-Person Pose Estimation Model," *European Confer-*

ence on Computer Vision, pp. 34-50, 2016.

[16] X. Bin, W. Haiping, and W. Yichen, “Simple Baselines for Human Pose Estimation and Tracking,” *Proceedings of the European Conference on Computer Vision (ECCV)*, pp. 466-481, 2018.

[17] C. Zhe, S. Tomas, W. Shih-En, S. Yaser, “Realtime Multi-Person 2D Pose Estimation Using Part Affinity Fields,” *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 7291-7299, 2017.

[18] H. Kaiming, G. Georgia, D. Piotr, and G. Ross, “Mask R-CNN,” *Proceedings of the IEEE International Conference on Computer Vision (ICCV)*, pp. 2961-2969, 2017.

[19] Z. Ning, F. Yiran, and E.-J. Lee, “Activity Object Detection Based on Improved Faster R-CNN,” *Journal of Korea Multimedia Society*, Vol. 24, No. 3, pp. 416-422, 2021

[20] J. Redmon and A. Farhadi, “YOLOv3: An Incremental Improvement,” *Proceedings of the European Conference on Computer Vision (ECCV)*, 2018.

Ning Zhang

received his B. S. at Dalian Polytechnic University in China (2008-2012), and received his master degree and doctor degree at Busan Tongmyong University in Korea (2014-2016, 2016-2020). Currently, he is working as an information technology research engineer at Shenyang Duoyuan Co., Ltd. in China. His main research areas are image processing,computer vision and intelligent pattern recognition.

Jin-Ho Park

received his B. S. in Business Administration from Tongmyong University, Korea, in 2017; his M. S. in Information and Communication Engineering from Tongmyong University, Korea, in 2019. Currently, he is studying in Department of Information and Communication Engineering from Tongmyong University, Korea for doctoral degree. His main research areas are image processing and pattern recognition.

Eung-Joo Lee

Eung-Joo Lee received his B. S., M. S. and Ph. D. in Electronic Engineering from Kyungpook National University, Korea, in 1990, 1992, and Aug. 1996, respectively. Since 1997 he has been with the Department of Information & Communications Engineering, Tongmyong University, Korea, where he is currently a professor. From 2000 to July 2002, he was a president of Digital Net Bank Inc. From 2005 to July 2006, he was a visiting professor in the Department of Computer and Information Engineering, Dalian Polytechnic University, China. His main research interests include biometrics, image processing, and computer vision.