Training Data Generation Based on Observation Probability Density for Human Pose Refinement

Kazumasa Oniki, Toshiki Kikuchi, and Yuko Ozasa
Keio University, Kanagawa, Japan
Email: {onikikazumasa1017, tkikuchi, yuko.ozasa}@keio.jp

***Abstract*—Human pose estimation is an active research topic since for decades, and it has immediate applications in various tasks such as action understanding. Although accurate pose estimation is an important requirement, joint occlusion and various gestures of a person often result in deviated pose predictions. In this paper, we aim to correct such outliers included in pose estimation results. We propose a method to generate training data which is effective for learning models for outlier correction.**

***Index Terms*—human pose estimation, machine learning, outlier correction**

I. Introduction

Human pose estimation is a fundamental yet challenging problem in computer vision. Recently, remarkable advances have been achieved in human pose estimation because of the appearance of depth sensors like Kinect, and the powerful Deep Convolutional Neural Networks (DCNN) [1], [2]. Understanding of a person's limb articulation location is helpful for high-level vision tasks like action recognition, and also serves as a fundamental tool in fields such as human-computer interaction applications [3], [4].

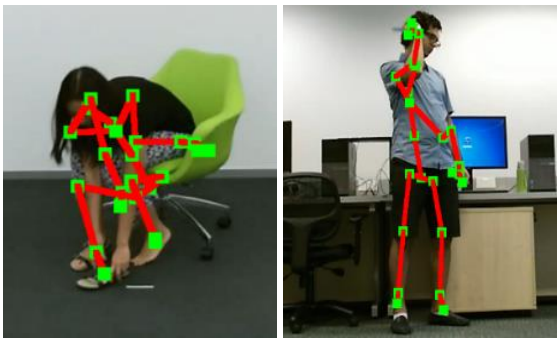


Figure 1. Pose estimation results which seem to include outliers.

Although getting accurate joint locations is crucial in human pose estimation, it remains challenging due to the highly complex joint configuration, partial or even complete joint occlusion, and various gestures of a person. For body parts with heavy occlusion or various posture change, DCNN may have difficulty to locate each body part correctly. For example, some pose estimation results including outliers are shown in Fig. 1. They are captured by Kinect v2, which predicts human pose with implausible configuration. Since accurate pose estimation is an important requirement for activity recognition with diverse applications such as human-computer interaction, correcting such outliers is necessary.

There already exists a wide variety of methods trying to improve pose inference strategy [5], [6]. Although these approaches achieved significant advancement in human pose estimation, they often predict human poses including outliers. Here, an outlier is an observation of joint coordinates that deviate much from a real one. In this paper, we aim to correct outliers included in pose estimation results. The purpose of this research is not to improve pose estimation accuracy like the study [5], [6], but to correct outliers of estimated skeleton. Therefore, outlier correction that we present can be used in conjunction with any existing methods for pose estimation.

The simplest way to correct outliers would be, if the input is a video sequence, to correct outliers of a frame using its front and rear frame elements. However, it depends on time-series data, and there is no guarantee that the pose estimation results of front and rear frames are correct. Thus, we take an approach to extract skeletal features from pose estimation results, and correct outliers based on them.

Since humans perform various poses, it is difficult to select features of skeletons manually which is effective for outlier correction. We should rely on a machine learning method that can automatically extract skeletal features. To the best of our knowledge there have been no reports which aim at outlier correction in human pose estimation, and there is no training data for this task.

If there are any training data, outliers in pose estimation results can be corrected with existing machine learning methods. We propose a method to generate training data for outlier correction. The probability density for the positions of each joint is estimated based on observed skeletal data, and by updating the position of each joint such that its probability density gets lower, skeletal data including plausible outliers is generated. We train the model for outlier correction using generated skeletal data, and correct outliers by taking skeletal data as input to the learned model. Note that the target to correct is an implausible skeleton which is rarely observed. Generated training data is helpful to modify such skeleton to a plausible one with high observation probability.

Manuscript received September 25, 2020; revised March 4, 2021.

doi: 10.18178/joig.9.2.50-54

To evaluate the effectiveness of the training data generated by our proposed method, we train two models for outlier correction, and evaluate the correction accuracy of each model quantitively. Your goal is to simulate the usual appearance of papers in the. We are requesting that you follow these guidelines as closely as possible.

II. Related Work

Multi-person pose estimation methods can be grouped into two types: top-down and bottom-up approaches. Top-down approaches [7], [8] employ a person detector and then perform single -person estimation for each detected person. Bottom-up methods [1], [9] first predict all body joints and then group them into full poses of different people. Although great progress has been made by these methods, there still exist a lot of challenging cases, such as occluded joints and crowded background, which cannot be well localized. In order to deal with such cases, many researches have been made to correct pose estimation results, in addition to ordinary pose estimation.

Most methods perform pose estimation and refinement in one go [2], [5], [10], [11]. Carreria *et al.* [10] proposed a self-correcting model that progressively changes an initial pose estimation by feeding back error predictions. Chen *et al.* [5] proposed a cascaded pyramid network, which integrates global pyramid network and pyramid refined network based on online hard keypoint mining. Likewise, Newell *et al.* [2] and Weii *it et al.* [11] utilized an end-to-end trainable multi-stage architecture-based network. Each stage tries to refine the pose estimation results. All of these methods combine pose estimation and refinement into a single model, and each refinement module is dependent on estimation. Therefore, the refinement modules have different structures, and they are not guaranteed to work successfully when they are combined with other estimation methods. On the other hand, we separate pose estimation and refinement into two parts, and focus only on refinement. Our pose refinement approach is independent of pose estimation, and therefore it is applied to any pose estimation method. Note that what we present is not a kind of pose estimation but pose refinement.

There already exist some researches which aim at pose refinement [12]-[14]. Fieraru *et al.* [12] synthesized the common failure cases of human pose estimators, and proposed a network to refine the pose estimation results using synthesized skeletons. Moon *et al.* [13] pointed out that the data augmentation presented by Fieraru *et al.* [12] is not based on actual error statistics, and proposed a method to generate training data taking statistical error distribution into account. Besides, they designed a coarse-to-fine estimation pipeline which achieves better result than conventional multi-stage architecture-based refinement methods. These approaches [12], [13] are the post-processing step applied to pose estimation results, and work on top of any human pose estimation method like our approach. However, they require the ground truth of skeletal data and true/false labels when generating training data for error correction. They are hard to be applied to the scenes where the ground truth and true/false labels of skeletal data are not prepared. In contrast, our proposed approach generates training data for outlier correction using only pose estimation results. Since it doesn't require any labels and ground truth, it can be easily used in real-world scenes. Wan *et al.* [14] proposed a method that can be applied to any 3D pose estimation approach, and contrary to other methods [12], [13], it doesn't need true/false labels of skeletal data. However, the ground truth of 3D skeletal data is essential in the training of refinement module. It is also hard to be applied when there is not the ground truth of 3D skeletal data.

III. Generation of Skeletal Data Including Outliers

We introduce the composition of training data for outlier correction. Then we present our approach to generate training data, discussing the requirements of outliers included in skeletal data.

It is possible to train the model for outlier correction with training data composed of the pairs of skeletal data including outliers and its ground truth. In the proposed method, we regard skeletal data $\mathbf{z}$ acquired by an existing method as ground truth, and generate skeletal data $\tilde{\mathbf{z}}$ including outliers from $\mathbf{z}$.

Generating outliers doesn't mean the joint coordinates of skeletal data have only to be changed randomly. This method does not take skeletal distribution into account, and the resulting outliers do not imitate outliers included in pose estimation results. It seems that using plausible skeletal data as training data leads to more accurate correction of pose estimation results than using skeletal data generated randomly.

Algorithm 1 shows a procedure to generate skeletal data by our proposed method. There are two requirements to satisfy in generating skeletons with outliers:

- **Observing condition**

A skeleton consists of joints which can actually be observed.

- **Outlier condition**

Some joints locate the position where they are hardly observed.

The proposed method generates skeletons satisfying both the above observing and outlier condition, which can be regarded as outliers.

Algorithm 1 Generating a Skeleton data Including Plausible Outliers

Input: a skeleton with N joints $\mathbf{z} = \{\mathbf{p^k}\}_{k=1}^{N}$
for $k = 1$ to N **do**
  % update the joint position for T_j times
  for $l = 1$ to T_j **do**
    % estimate gradient ∇KDE^k by central difference approximation
    ∇KDE^k =centraldiff($KDE(\mathbf{p^k})$)
    % update the joint position $\mathbf{p^k}$ only once using ∇KDE^k by optimizer (e.g. Adam)
    $\mathbf{p^k}$ =optimize($\mathbf{p^k}, \nabla KDE^k$)
  end for
end for
Output: a skeleton $\tilde{\mathbf{z}} = \{\mathbf{p^k}\}_{j=1}^{N}$

To satisfy such requirements, the frequency of observing joints on each coordinate should be modeled. We prepare a dataset $\{\boldsymbol{z}'\}$ to find characteristics of each joint coordinate, and calculate the observing frequency of each joint thereof. In the proposed method, at first, we estimate the probability density of N joints respectively using kernel density estimation (KDE). We can estimate the probability density on k-th joint position $\boldsymbol{p^k} = (p_1^k, \ldots, p_d^k)$ as:

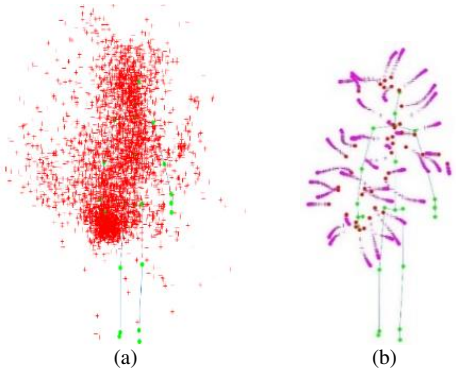
$$KDE(\boldsymbol{p^k}) = \frac{1}{nb_1 \cdots b_d} \sum_{i=1}^{n} \prod_{j=1}^{d} K\left(\frac{p_j^k - p_{ij}'^k}{b_j}\right) \quad (1)$$

where d is the dimension of joint coordinates, $\boldsymbol{p_i'^k} = (p_{i1}'^k, \ldots, p_{id}'^k)$ is the i-th sample included in $\boldsymbol{z}'$, n is the number of observations of the sample, $K(\cdot)$ is the kernel function, and $b_1 \cdots b_n$ are the bandwidth. Based on the estimated probability density, the gradient of probability density is calculated by central difference approximation.

By using an optimization method like Adam [15] with the estimated gradients, joint coordinates of skeletal data can be iteratively updated such that its probability density gets lower. Since the updated coordinates are the joint coordinates with low observation probability, the resulting data can be seen as the skeletal data including plausible outliers.

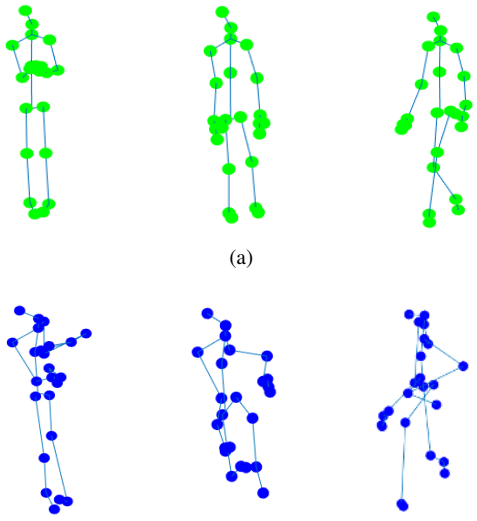
Fig. 2 shows the result of applying the proposed method to joints. We can see that they are shifted to the region where the probability density is low. Fig. 3 shows the example of skeletal data generated by the proposed method. By using the proposed method, the position of each joint is updated to the position not with no prospect of being observed ($KDE(\boldsymbol{p})$ >0), but with lower probability density, which enables us to obtain skeletal data satisfying observing and outlier condition in this paper.

Models can learn the mapping $\tilde{\boldsymbol{z}} \rightarrow \boldsymbol{z}$ using $\tilde{\boldsymbol{z}}$ generated by the proposed method. The learned models output the skeletal data, with outliers being corrected.



(a) (b)

Figure 2. Examples of updating joint location by the proposed method. (a) The distribution of observation points of wrist joint (red). Around the center, joints are densely observed and the probability density is high. The area far from the center has low probability density, since joints are sparsely observed in that area. (b) The process of updating the positions of wrist joints. From the positions where they are observed (red), they are iteratively shifted to the positions with lower probability density (Magenta).



(a) (b)

Figure 3. (a) Examples of skeletal data $\boldsymbol{z}$ as ground truth. (b) Examples of skeletal data $\tilde{\boldsymbol{z}}$ obtained by the proposed method.

IV. Experiment

We don't compare outlier correction with other pose refinement methods because the task is different from others. Existing methods [12]-[14] assume that the ground truth and true/false labels of skeletal data are prepared when training the model to correct pose estimation. By contrast, outlier correction requires neither ground truth nor true/false labels. It can train the model with only pose estimation results. It's not fair to compare methods whose assumptions are different by the same evaluation indices. In the experiment, we compare the training data generated by the proposed method with existing training data for denoising.

To evaluate the effectiveness of the training data generated by the proposed method, we adopt the training data for Denoising Autoencoder (DAE) proposed by Vincent [16] as baseline. DAE is trained to reconstruct a clean input from partially destroyed one which is added with random noises following gaussian or uniform distribution. It is known that the learned DAE removes the noises included in input data, in the process of encoding the input into latent representation and decoding it.

In order to compare the effects of each training data on outlier correction, we performed the following experiments. In addition to the skeletal data generated by our proposed method (*plausible*), we prepared the skeletal data added with random values following uniform distribution (*uniform*) and gaussian distribution (*gaussian*), and compared the results of correcting outliers using each data as training data. In the experiment, we selected training and test data from any one of above three types of skeletal data, and measured the error $d(f(\tilde{\boldsymbol{z}}), \boldsymbol{z})$ between the ground truth $\mathbf{z}$ and the output of the learned model $f(\tilde{\boldsymbol{z}})$. The error $d(f(\tilde{\boldsymbol{z}}), \boldsymbol{z})$ is represented by a sum of all L2 distance between corresponding joints of two skeletons. A small error

means that the model corrects outliers in test data with a high precision. The experiment was performed with every possible combinations of skeletal data as training and test data.

From this evaluation, we can see which training data is more effective for outlier correction: the data used for denoising or the data generated by the proposed method. In addition to DAE, we also experimented with the model based on Graph Convolutional Network (GCN) [17] to demonstrate that the proposed method is effective with any model.

A. Experimental Setting

The skeletal data are estimated by Kinect v2, and composed of three-dimensional coordinates of 25 joints (d = 3, N = 25). Before applying KDE, n = 1000 skeletons used for KDE are translated and rotated following the normalization preprocessing step presented by Shahroudy *et al.* [18] to normalize the position, direction, and size of each skeleton. In KDE, we used Gaussian kernel and allocated bandwidth based on Silverman's rule of thumb [19]. We employed Adam optimizer [15] for updating joint coordinates, and set the update frequency T_j to a random integer between 1 and 100. For training and test data, we randomly selected 10000 and 2000 skeletal data from NTU RGB-D respectively, and generated skeletal data $\tilde{\boldsymbol{z}}$ including outliers from them. The noises used as baseline (*uniform, gaussian*) were generated such that the mean of values to add to each joint coordinate is equal to zero, and the variance is equal to that of outliers generated by the proposed method.

We chose some hyperparameters to tune: the learning rate, the batch size for both DAE and GCN, the number of hidden layer and nodes in each layer for DAE, and the number of convolutional layers and channels in each layer for GCN. They were optimized by Optuna [20]. In the experiment, we used the models with the above hyperparameters optimized on condition that train and test data consist of *plausible* data.

B. Results

Table I shows the result of outlier correction with the use of DAE and GCN. The error $d(f(\tilde{\boldsymbol{z}}), \boldsymbol{z})$ of the model trained with *uniform* and *gaussian* represents the correction results by the normal denoising network, and the rest values correspond to the results of outlier correction. We also set *uniform*, *gaussian*, *plausible* as test data. Among them, the results using *plausible* as test data are the most crucial, since *plausible* includes plausible outliers. These values should be compared.

In the case of using DAE, changing the noise (*uniform*, *gaussian*) used for learning did not make a huge difference to the correction accuracy of test data. When test data are *plausible* data, the model learned with *plausible* data corrected test data with the highest accuracy among all possible training data. This is possibly attributed to the fact that training and test data are the skeletal data following the same distribution. However, in this case, the error $d(f(\tilde{\boldsymbol{z}}), \boldsymbol{z})$ which represents the degree of correcting test data was 2.844, whereas a normal DAE trained and tested on noises had the error of more than 4. This indicates that the learning with *plausible* data is more effective to correct outliers of skeletal data than denoising by a normal DAE. Similar results were obtained from the experiment using GCN as the model.

Fig. 4 (I) (II) shows input-output examples of DAE when test data are *plausible* data. According to (b), we found that the DAE trained with *plausible* data output the skeleton which is the most similar to the ground truth $\boldsymbol{z}$ compared to the one trained with other noises.

TABLE I. CORRECTION PERFORMANCE $d(f(\tilde{\boldsymbol{z}}), \boldsymbol{z})$) OF EACH MODEL

(a) DAE

		test data		
		uniform	*gaussian*	*plausible*
training data	*uniform*	4.239	4.179	5.169
	gaussian	4.253	4.189	5.185
	plausible	5.042	5.042	**2.844**

(b) GCN

		test data		
		uniform	*gaussian*	*plausible*
training data	*uniform*	4.416	4.776	4.077
	gaussian	4.419	4.376	4.329
	plausible	5.128	5.331	**2.693**

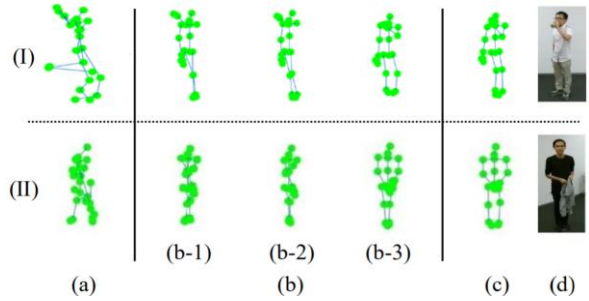


Figure 4. (a) Input: Skeletal data $\tilde{\boldsymbol{z}}$ including outliers. (b) Output: From left to right, skeletal data $f(\tilde{\boldsymbol{z}})$ corrected by DAE trained with (b-1) *uniform*, (b-2) gaussian, and (b-3) *plausible* data respectively. (c) Skeletal data $\boldsymbol{z}$ as ground truth. (d) Corresponding RGB image.

V. CONCLUSION

In this paper, we aim to correct outliers in pose estimation results and proposed a method to generate training data which is effective for machine learning of outlier correction. The experimental results show that the training data generated by the proposed method is more effective for correcting skeletal data with low observation probability than existing training data for denoising. In future work, we will apply outlier correction to the skeletal data captured in real environments, and evaluate the effectiveness of outlier correction qualitatively.

CONFLICT OF INTEREST

The authors declare no conflict of interest.

AUTHOR CONTRIBUTIONS

Kazumasa Oniki conducted the research, analyzed the data, and wrote the paper; Toshiki Kikuchi and Yuko Ozasa considered the proposed method together and refined the paper; All authors had approved the final version.

ACKNOWLEDGMENT

This work was supported by Keio university. The authors are grateful to H. Saito, a professor of Hyper Vision Research Laboratory, for giving us an opportunity of this study.

REFERENCES

[1] Z. Cao, T. Simon, S. E. Wei, and Y. Sheikh, "Realtime multi-person 2d pose estimation using part affinity fields," in *Proc. the IEEE Conference on Computer Vision and Pattern Recognition*, 2017, pp. 7291-7299.
[2] A. Newell, K. Yang, and J. Deng, "Stacked hourglass networks for human pose estimation," in *Proc. European Conference on Computer Vision*, 2016, pp. 483-499.
[3] R. Saini, P. Kumar, B. Kaur, P. P. Roy, D. P. Dogra, and K. Santosh, "Kinect sensor-based interaction monitoring system using the BLSTM neural network in healthcare," *International Journal of Machine Learning and Cybernetics*, vol. 10, no. 9, pp. 1-12, 2018.
[4] S. Suda, Y. Makino, and H. Shinoda, "Prediction of volleyball trajectory using skeletal motions of setter player," in *Proc. the 10th Augmented Human International Conference*, 2019, pp. 1-8.
[5] Y. Chen, Z. Wang, Y. Peng, Z. Zhang, G. Yu, and J. Sun, "Cascaded pyramid network for multi-person pose estimation," in *Proc. the IEEE Conference on Computer Vision and Pattern Recognition*, 2018, pp. 7103-7112.
[6] X. Peng, Z. Tang, F. Yang, R. S. Feris, and D. Metaxas, "Jointly optimize data augmentation and network training: Adversarial data augmentation in human pose estimation," in *Proc. the IEEE Conference on Computer Vision and Pattern Recognition*, 2018, pp. 2226-2234.
[7] K. He, G. Gkioxari, P. Dollar, and R. Girshick, "Mask r-cnn," in *Proc. the IEEE International Conference on Computer Vision*, 2017, pp. 2961-2969.
[8] S. Huang, M. Gong, and D. Tao, "A coarse-fine network for keypoint localization," in *Proc. the IEEE International Conference on Computer Vision*, 2017, pp. 3028-3037.
[9] E. Insafutdinov, L. Pishchulin, B. Andres, M. Andriluka, and B. Schiele, "Deepercut: A deeper, stronger, and faster multi-person pose estimation model," in *Proc. European Conference on Computer Vision*, 2016, pp. 34-50.
[10] J. Carreira, P. Agrawal, K. Fragkiadaki, and J. Malik, "Human pose estimation with iterative error feedback," in *Proc. the IEEE Conference on Computer Vision and Pattern Recognition*, 2016, pp. 4733-4742.
[11] S. E. Wei, V. Ramakrishna, T. Kanade, and Y. Sheikh, "Convolutional pose machines," in *Proc. the IEEE Conference on Computer Vision and Pattern Recognition*, 2016, pp. 4724-4732.
[12] M. Fieraru, A. Khoreva, L. Pishchulin, and B. Schiele, "Learning to refine human pose estimation," in *Proc. the IEEE Conference on Computer Vision and Pattern Recognition Workshops*, 2018, pp. 205-214.
[13] G. Moon, J. Y. Chang, and K. M. Lee, "Posefix: Model-Agnostic general human pose refinement network," in *Proc. the IEEE Conference on Computer Vision and Pattern Recognition*, 2019, pp. 7773-7781.
[14] Q. Wan, W. Qiu, and A. L. Yuille, "Patch-based 3D human pose refinement," arXiv preprint arXiv:1905.08231, 2019.
[15] D. P. Kingma and J. Ba, "Adam: A method for stochastic optimization," arXiv preprint arXiv:1412.6980, 2014.
[16] P. Vincent, H. Larochelle, Y. Bengio, and P. A. Manzagol, "Extracting and composing robust features with denoising autoencoders," in *Proc. the 25th International Conference on Machine Learning*, 2008, pp. 1096-1103.
[17] S. Yan, Y. Xiong, and D. Lin, "Spatial temporal graph convolutional networks for skeleton-based action recognition," in *Proc. Thirty-Second AAAI Conference on Artificial Intelligence*, 2018.
[18] A. Shahroudy, J. Liu, T. T. Ng, and G. Wang, "NTU RGB+ D: A large scale dataset for 3D human activity analysis," in *Proc. the IEEE Conference on Computer Vision and Pattern Recognition*, 2016, pp. 1010-1019.
[19] B. Silverman, "Monographs on statistics and applied probability," *Density Estimation for Statistics and Data Analysis*, vol. 26, 1986.
[20] Preferred networks, define-by-run hyperparameter optimization framework. [Online]. Available: https://optuna.org

Kazumasa Oniki received his B.Eng. degree in Information and Computer Science from Keio University, Japan, in 2018. He is currently a MS student in Machine Learning and Computer Vision at Keio University. His research interests include Human sensing, machine learning, and activity recognition.

Toshiki Kikuchi received his B.Eng. degree in Information and Computer Science from Keio University, Japan, in 2018. He is currently a MS student in Machine Learning and Computer Vision at Keio University. His research interests include machine learning, audio-visual processing, and computer graphics.

Yuko Ozasa received her PhD degree in engineering from Kobe University in 2015. She was a postdoctoral researcher at National Institute of Advanced Industrial Science and Technology (AIST) in 2015. Since 2015 she has been a research associate at Graduate School of Science and Technology, Keio University. Her research interests include object recognition and grounding, multimodal fusion, visual perception, and hyperspectral sensing.