Transfer Learning based Precise Pose Estimation with Insufficient Data

Wonje Choi
Department of Computer Science and Engineering,
Sungkyunkwan University, South Korea
wjchoi1995@g.skku.edu

Honguk Woo*
Department of Computer Science and Engineering,
Sungkyunkwan University, South Korea
hwoo@skku.edu

ABSTRACT

With the recent advance in computer vision techniques and the growing utility of real-time human pose detection and tracking, deep learning-based pose estimation has been intensively studied in recent years. These studies rely on large-scale datasets of human pose images, for which expensive annotation jobs are required due to the complex spatial structure of pose keypoints. In this work, we present a transfer learning-based pose estimation model that leverages low-cost synthetic datasets and regressive domain adaptation, enabling the sample-efficient learning on precise human poses. In evaluation, we demonstrate that our model achieves the high accurate pose estimation on a dataset of golf swing images, which is targeted for a virtual golf coaching application.

CCS CONCEPTS

• **Computing methodologies**; • **Machine learning**; • **Learning paradigms**;

KEYWORDS

Pose estimation, Synthetic data, Domain adaptation

ACM Reference Format:
Wonje Choi and Honguk Woo*. 2022. Transfer Learning based Precise Pose Estimation with Insufficient Data. In *2022 the 5th International Conference on Machine Vision and Applications (ICMVA) (ICMVA 2022), February 18–20, 2022, Singapore, Singapore.* ACM, New York, NY, USA, 6 pages. https://doi.org/10.1145/3523111.3523118

1 INTRODUCTION

Pose estimation for human bodies refers to a computer vision task of processing camera images or videos as input to generate the spatial location information of human body joints or semantic keypoints (e.g., left shoulder) in real-time. Pose estimation is considered fundamental for natural understanding on human behaviors, and it has been adopted in numerous machine learning based applications, e.g., virtual fitting, pedestrian tracking, virtual sports coaching, etc.

Deep learning techniques such as convolutional neural network (CNN) models have been used for 2D human pose estimation [4]. In general, these have been developed based on large-scale datasets of human pose images with many keypoints, for which much annotation labor is required. Thus, when a precise model is needed for new poses but a dataset for the new poses is not available, the effort required to build such a dataset is often an issue.

In this paper, we propose a sample-efficient learning model for 2D human pose estimation for which the two-stage transfer learning scheme with keypoint set extension and synthetic-to-real domain adaptation techniques is employed. Specifically, we generate a pose-aware synthetic dataset to extend the number of pose keypoints from 10 to 21, and then adapt the regressive domain adaptation scheme to transfer the pose knowledge learned on the synthetic dataset of 21 keypoints to our task-specific real data. In that way, we achieve a high-performance pose estimation model that makes inferences on 21 keypoints without a large-scale dataset of photo-realistic images annotated with those 21 keypoints.

Our contributions of this paper are two-folded.

- We present the transfer learning-based pose estimation model by which a set of keypoints can be extended from a source dataset (i.e., 10 keypoints in the source) to a task-specific target model (i.e., 21 keypoints in the target).
- We achieve high accuracy on golf swing pose estimation, e.g., 80.9 mAP, 22.4% higher than a state-of-art method and 4 times higher than the synthetic scratch model.

2 RELATED WORKS

2.1 Whole-Body Pose Estimation

2D pose estimation aims at predicting key parts on detected objects, and it has become a popular research topic for its wide use in computer vision applications. Recently, deep learning-based pose estimation techniques [5, 6, 11, 14, 15, 19–21] have shown a significant progress thanks to the availability of large-scale training datasets such as MPII [2] and COCO keypoint datasets [10]. There have been numerous deep learning models based on CNN for pose estimation, e.g., facial keypoint detection [8, 17] and hand pose estimation [12, 13, 18, 24].

Recently, Zhang et al. [22] explored a task of localizing the dense keypoints of an entire body. Without whole-body annotation, several regional keypoint detection models learned from independent body-part datasets should be assembled to make inferences on whole-body keypoints. Their whole-body annotation on the COCO dataset enables the integration of those independent body-part detection procedures.

2.2 Domain Adaptation for Pose Estimation

In general, domain adaptation deals with the cases where labeled data for a target domain might not be sufficient to learn but the knowledge learned from a source dataset in a similar domain to the

ICMVA 2022, February 18–20, 2022, Singapore, Singapore

ACM ISBN 978-1-4503-9567-0/22/02…$15.00
https://doi.org/10.1145/3523111.3523118

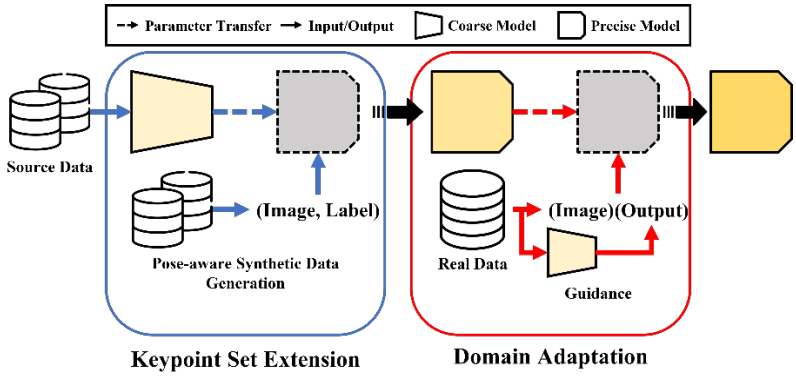

Figure 1: PPE architecture consists of two stages. Keypoint Set Extension, on the left, generates a pose-aware synthetic dataset and conducts model learning. Domain adaptation, on the right, conducts domain adaptation from synthetic data to target data.

target can be leveraged. In pose estimation, adversarial training is used for domain adaptation where inferences from a pre-trained source model can provide the pseudo-labels for target image data.

Cao et al. [3] proposed a novel cross-domain adaptation method to transform the knowledge from labeled animal classes to unlabeled classes. For pose estimation on unseen four-legged animals, they exploited the skeleton similarity between pose-labeled domains (i.e., human and labeled animals) and the adversarial training scheme with a domain discriminator, boosting up the model performance using the pseudo labeling module with progressively increasing reliability. Li et al. [9] presented a multi-scale domain adaptation module and online coarse-to-fine pseudo label updating strategy to reduce the domain gap between synthetic and real data.

Jiang et al. [7] proposed the unsupervised domain adaptation method for regression tasks by exploring the sparsity of regression space in keypoint detection model outputs and narrowing the difference between regression and classification. They treated the regressive domain adaptation problem as a minimax game with an adversarial regressor. We adopt this approach as our baseline for keypoint domain adaptation.

3 METHOD

In this section, we present our method for precise pose estimation (PPE) with two stages, (1) keypoint set extension and (2) synthetic-to-real domain adaptation. Figure 1 illustrates the overall framework architecture.

3.1 Keypoint Set Extension

In keypoint set extension, we first create pose-aware synthetic data and then train a model on the synthetic data. The synthetic data is created to bridge the gap between a source (i.e., COCO keypoint dataset) and a task-specific target dataset (i.e., golf swing dataset), where each synthetic golf swing image is set to have the same keypoint types as the real golf swing image. In doing so, original 10 keypoints of a whole-body are derived from the source and additional task-specific 11 keypoints are automatically annotated by computer-generated imagery (CGI) software according to target data samples. A simulated environment is implemented in CGI where animated characters and their skeleton coordinates are pre-defined. Then, transfer learning is conducted for learning to infer those newly added 11 keypoints, while the knowledge on the source such as the keypoint detection capability on COCO can be properly maintained. This is effective since our synthetic dataset is aligned with both source and target keypoints, although the number of keypoints is different; the source has 10 but the target has 21 keypoints.

In transfer learning, we initialize a model f_{syn} for synthetic dataset D_{syn} with the parameters of pre-trained model f_S from source dataset D_S. The model f_{syn} is then fine-tuned by D_{syn}, i.e., minimizing the loss $\mathcal{L}_{ext}$ based on mean-square error (MSE),

$$\mathcal{L}_{ext}\left(f_{syn}\right) = \mathbb{E}_{(x,y)\sim D_{syn}}\left[\| f_{syn}\left(x\right) - y\|_2\right] \tag{1}$$

where x and y denote synthetic image data and ground truth keypoints, respectively.

3.2 Domain Adaptation from Synthetic to Real

To achieve a high-performance PPE model, we employ a domain adaptation scheme by which a model learned on the synthetic dataset (which is generated by the keypoint set extension) is fine-tuned for target data samples. Specifically, we adapt a state-of-the-art algorithm of domain adaptation for keypoint detection tasks, RegDA [7], to enhance the PPE performance by exploiting the knowledge on original 10 keypoints that are common in both the source and target domains.

Our PPE model $f = f_{syn}$ is structurally divided into regressor r and feature generator ψ, where r corresponds to the last layer of f and ψ corresponds to the other layers, i.e., $f = r \circ \psi$. The same structure is used for adversarial regressor r_{adv}, so $f_{adv} = r_{adv} \circ \psi$. In domain adaptation, we minimize the expected error on target dataset D_T.

$$err_{D_T} = \mathbb{E}_{(x_t,y_t)\sim D_T}\left[\mathcal{L}\left(f\left(x_t\right), y_t\right)\right] \tag{2}$$

In [23], this error is bounded by the sum of empirical error on the synthetic dataset $err_{D_{syn}}(f)$, empirical disparity discrepancy d, ideal error, and complexity terms. Thus, we can optimize the model f with

$$\min\left[err_{D_{syn}}\left(f\right) + d_{f,\mathcal{F}}\left(D_{syn}, D_T\right)\right]. \tag{3}$$

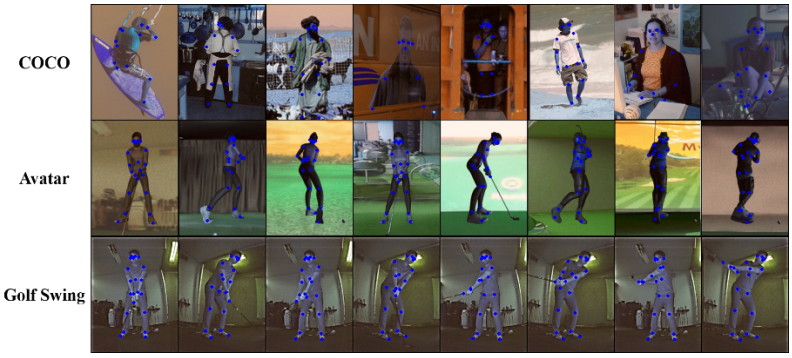

Figure 2: Image samples of three datasets in Table 1, where the first row contains the source (COCO), the second row contains the synthetic (golf swing of avatars), and the last row contains the target (golf swing of humans).

Table 1: Dataset

Features	Dataset		
	Source	Synthetic	Target
Name	COCO [10]	Avatar	Golf Swing
Keypoints	10 (17)	21	21
Samples	156,165	23,766	3,424
Complexity	High	Low	Low

where $\mathcal{F}$ is hypothesis space, and empirical disparity discrepancy d is defined as

$$d_{f,\mathcal{F}}\left(D_{syn}, D_T\right) = sup_{f_{adv}\in\mathcal{F}} \left[\mathbb{E}_{x\sim D_T}\mathcal{L}\left(f_{adv}\left(x\right), f\left(x\right)\right) - \mathbb{E}_{x\sim D_{syn}}\mathcal{L}\left(f_{adv}\left(x\right), f\left(x\right)\right)\right]. \quad (4)$$

To optimize the objective in Eq. (3), we minimize expected error $err_{D_{syn}}(f)$ for synthetic dataset D_{syn}, train f_{adv} to maximize $\mathbb{E}_{x\sim D_T}\mathcal{L}(f_{adv}(x), f(x)) - \mathbb{E}_{x\sim D_{syn}}\mathcal{L}(f_{adv}(x), f(x))$, and train f to minimize empirical disparity discrepancy in Eq. (4), in the same way as [7]. This proceeds through the objective functions below.

Objective 1. To minimize expected error $err_{D_{syn}}(f)$ and the second term of Eq. (4) for synthetic dataset D_{syn}, we use the loss function defined as

$$\mathcal{L}_{obj1}\left(f,\ f_{adv}\right) = \mathbb{E}_{(x_s,y_s)\sim D_{syn}}\left[\mathcal{L}\left(f\left(x_s\right), y_s\right) + \mathcal{L}\left(f_{adv}\left(x_s\right), f(x_s)\right)\right] \quad (5)$$

where $\mathcal{L}$ denotes the KL divergence loss.

Objective 2. To maximize the first term in Eq. (4), we find the function f_{adv} that maximized $\mathbb{E}_{x\sim D_T}\mathcal{L}(f_{adv}(x), f(x))$. Then, we optimize r_{adv} by using ground-false loss $\mathcal{L}_F$such that $\mathcal{L}$ increases when $\mathcal{L}_F$ decreases.

$$\mathcal{L}_{obj2}\left(r_{adv}\right) = \mathbb{E}_{x_t\sim D_T}\mathcal{L}_F\left(f_{adv}\left(x_t\right), f\left(x_t\right)\right) \quad (6)$$

Objective 3. To minimize empirical disparity discrepancy in Eq. (4), we optimize ψ by using the loss defined as

$$\mathcal{L}_{obj3}\left(\psi\right) = \mathbb{E}_{x_t\sim D_T}\mathcal{L}\left(f_{adv}\left(x_t\right), f\left(x_t\right)\right). \quad (7)$$

Following the above three objective functions, we consider an additional loss form to make adaptation more effective. Since source model f_S can provide more accurate outputs for corresponding 10 keypoints of target dataset D_T than the outputs of synthetic model f, we use the outputs of f_S as ground-truth. This minimizes the expected error on D_T directly. Therefore, we can minimize the empirical disparity discrepancy more effectively by optimizing ψ. Then, the guidance loss is defined as

$$\mathcal{L}_{gd}\left(\psi\right) = \mathbb{E}_{x_t\sim D_T}\left[\mathcal{L}\left(f\left(x_t\right), f_S\left(x_t\right)\right) + \mathcal{L}\left(f_{adv}\left(x_t\right),\ f_S\left(x_t\right)\right)\right] \quad (8)$$

which completes the definition of our loss function,

$$\mathcal{L}_{PPE} = \eta\left(\mathcal{L}_{obj1} + \mathcal{L}_{obj2} + \mathcal{L}_{obj3}\right) + \mu\mathcal{L}_{gd} \quad (9)$$

where η and μ are coefficient. The learning algorithm using this loss is in Algorithm 1.

Algorithm 1 Learning for PPE

Dataset: $D_S,\ D_{syn},\ D_T$
Models: f_S, /* pre-trained on D_S */ f_{syn}, $f = r \circ \psi$, $f_{adv} = r_{adv} \circ \psi$
/* Keypoint Set Extension */
$f_{syn} \leftarrow f_S$
loop
for each $x, y \in D_{syn}$ **do**
  $loss = \mathcal{L}_{ext}(f_{syn})$
  $optimize(f_{syn}, loss)$
end for
end loop
/* Domain Adaptation from Synthetic to Real */
$f \leftarrow f_{syn}$, η, μ: coefficient of $loss$
loop
for each $x, y \in D_{syn}$ and $x_t \in D_T$ **do**
  $loss = \eta(\mathcal{L}_{obj1}(f, f_{syn}) + \mathcal{L}_{obj2}(r_{adv}) + \mathcal{L}_{obj3}(\psi)) + \mu\mathcal{L}_{gd}(\psi)$
  $optimize((f, f_{adv}), loss)$
end for
end loop
return f

4 EXPERIMENTS

In this section, we evaluate our method by showing experiment results on three different datasets in Table 1.

4.1 Dataset

Source Dataset. We use the COCO (Microsoft Common Objects in Context) keypoint dataset with 156,165 samples, where the coordinates of human pose keypoints are annotated for each image. It covers a wide variety of viewpoints, backgrounds, sizes, and types of human poses, as exampled in the first row of Figure 2. Note that in our experiments, we use only 10 keypoints as our source data (among existing 17 keypoints in the COCO keypoint dataset).

Synthetic Dataset. We generate the pose-aware synthetic dataset with 23,766 samples using the professional software. It is intended to cover more diversity of viewpoints, backgrounds and types of avatar poses than a task-specific target dataset with a small number of samples, yet avatars generated in this synthetic dataset might appear different from real-world humans in the target dataset; several samples are shown in the second row of Figure 2. Since the software tool that we used for synthetic data generation produces highly precise and accurate coordinates for keypoints in avatars, it is inherently vulnerable to end up with over-fitting when a pose estimation model is learned only on this dataset.

Target Dataset. We use a dataset of real-world, photo-realistic images with 3,424 golf swing pose samples, where each is manually annotated for 21 keypoints. The number of samples is small, compared to the source and synthetic datasets, and this small dataset setting causes the problem of insufficient data for machine learning; several samples are shown in the last row of Figure 2. Note that we use this dataset only for evaluation and use only a small number of the samples restrictively in case of model training.

4.2 Implementation

4.2.1 Implementation for backbone model learning. Table 2 depicts hyperparameter settings for our backbone model with HRNet [16]. We use Adam optimizer and batch size 32 in a single GPU server. For scratch model learning on each dataset, the learning rate is set to 1e-3, and it drops to 1e-4 at 90 epochs and 1e-5 at 120 epochs. The total training is 140 epochs. For keypoint set extension, the learning rate is fixed to 25e-5 during 100 epochs. We use Object Keypoint Similarity (OKS) [1] as our evaluation metric,

$$\text{OKS} = \frac{\sum_i exp\left(-d_i^2/2s^2k_i^2\right)\delta(v_i > 0)}{\sum_i \delta(v_i > 0)} \tag{10}$$

where d_i denotes the Euclidean distance between the detected keypoint and corresponding ground-truth, v_i denotes the visibility flag of ground-truth, and s denotes the object scale. Note that k_i denotes a per-keypoint constant that controls falloff; for newly added keypoints, k_i is set to 0.89 uniformly. We use standard average precision (i.e., mAP; the mean of AP scores at each keypoint) with thresholds of $OKS = 0.50,\ 0.55,\ \ldots,\ 0.90,\ 0.95$).

4.2.2 Implementation for domain adaptation. Table 3 depicts hyperparameter settings for domain adaptation. We adopt mini-batch SGD with momentum of 0.7 and batch size 32 in a single GPU server. To fine-tune with target data, the training of 60 epochs is conducted. The trade-off coefficient η for RegDA [7] is set to 3.5. It is empirically observed that the coefficient is maintained on η of our loss function $\mathcal{L}_{PPE}$ in Eq. (9). Similarly, coefficient μ of $\mathcal{L}_{PPE}$ is set to 1. In whole-body evaluation on keypoints, we use OKS based mAP as metric. For individual keypoint evaluation, we use Percentage of Correct Keypoints (PCK) as metric.

Table 2: Hyperparameters for backbone model learning

Hyperparameter	Value
Optimizer	Adam
Learning Rate	1e-3, 23e-5
Epoch	140,100
Batch Size	32
Per-keypoint Standard Deviation	0.89
Evaluation Metric	OKS(mAP)

Table 3: Hyperparameters for domain adaptation

Hyperparameter	Value
Optimizer	SGD
Learning Rate	1e-1
Epoch	60
Batch Size	32
Regressor	2 convolutional layers
η	3.5
μ	1
Evaluation Metric	PCK, OKS

4.3 Experiment Results

4.3.1 Performance of Scratch Models. Figure 3 shows the difference among datasets, where our backbone HRNet model is learned individually on the three datasets described in Table 1. Both the target scratch model (Target in the x-axis) and the synthetic model (Synthetic) achieve high accuracy of 95.2 and 100 mAP, respectively. Their performance degrades close to zero for different datasets (i.e., Target, Synthetic, Source in the legend), indicating that the synthetic (i.e., Avatar) and target data (i.e., Golf Swing) do not share the same feature space. The source models (Source in the x-axis) show high performance of 76.5 mAP on the source dataset (Source in the legend) but show relatively low performance for the other datasets. However, if we consider the accuracy only for the common keypoints of the source and target datasets, we achieve relatively high performance of 94.1 mAP (by Target* in the legend). It is because the source and target features share similarity. This implies that the source (i.e., COCO) and target data (i.e., Golf Swing) share the feature space partially in common.

4.3.2 PPE Performance. Table 4 illustrates the PPE performance in both PCK (for individual keypoint detection) and OKS (for whole-body keypoint detection) on the evaluation dataset of realistic golf swing data, where each row corresponds to different models in comparison.

Our model (**Ours**) has significantly improved the performance in both PCK and OKS, by 22.4% over RegDA and by more than 4 times over the synthetic scratch model.

In Figure 4, we also visualize the results before and after learning PPE models, synthetic scratch model and ground-truth. The

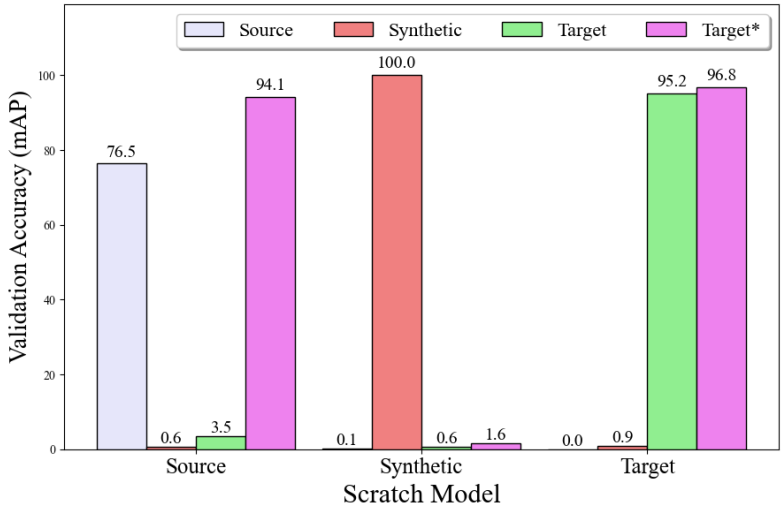

Figure 3: Comparison of scratch models. The x-axis denotes the scratch model from certain dataset and the y-axis denotes the achieved performance on each dataset. The data labeled Target* indicates the accuracy of keypoints that exist in both the source and target dataset.

Table 4: Accuracy on PPE

Method	Augmented Keypoints (PCK)							Total (OKS)
	nose	eyes	ears	ankles	toes	spine	club	Target
Synthetic scratch	0.26	0.15	0.09	0.39	0.26	0.05	0.12	14
RegDA	0.95	0.95	0.95	0.76	0.46	0.34	0.53	66.1
Ours	**0.99**	**0.99**	**0.97**	**0.98**	**0.55**	**0.53**	**0.57**	**80.9**
Target scratch	0.99	0.99	0.98	1	0.64	0.63	0.63	0.92

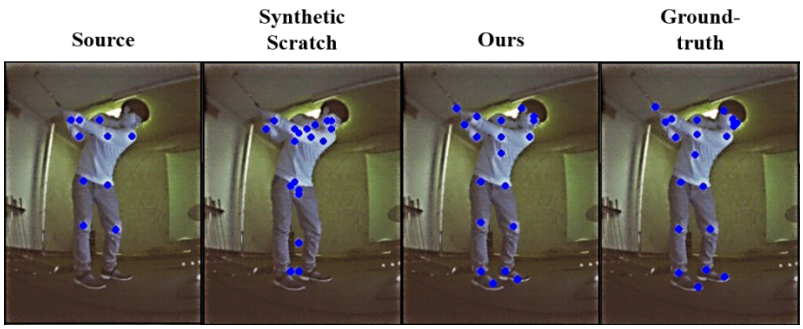

Figure 4: Qualitative results for the target dataset by each method. From the first column to the last, we have the inference outputs from the source model, the scratch model for synthetic, and our PPE model, and ground-truth samples.

first column shows the prediction of the source model. The model from source cannot detect additional keypoints. The second column shows the prediction of the scratch model on the synthetic dataset. The model often fails detecting several keypoints, yielding inaccurate prediction results. The third column represents the output of our model, showing no difference from the ground truth in the fourth column.

4.4 Ablation Study

We also conduct an ablation study to illustrate how much each stage affects the PPE model performance. The configuration in Table 5 includes a backbone model for synthetic dataset and loss used for adaptation to the target dataset (i.e., **Ours** at the first row consists of backbone model from keypoint set extension and the loss of PPE,

Table 5: Ablation study

Method	Configuration		Augmented Keypoints (PCK)							Total (OKS)
	Backbone	Loss	nose	eyes	ears	ankles	toes	spine	club	Target
Ours	EXT	$\mathcal{L}_{PPE}$	**0.99**	**0.99**	**0.97**	**0.98**	**0.55**	**0.53**	**0.57**	**80.9**
	Syn. scratch	$\mathcal{L}_{PPE}$	0.97	0.97	0.96	0.93	0.45	0.45	0.56	77.9
	EXT	$\mathcal{L}_{RegDA}$.	0.98	0.98	0.96	0.98	0.57	0.42	0.57	76
RegDA	Syn. scratch	$\mathcal{L}_{RegDA}$.	0.95	0.95	0.95	0.76	0.46	0.34	0.53	66.1

denoted as $\mathcal{L}_{PPE}$). From second to fourth row, we replace each configuration from one of RegDA [7] where backbone is scratch model trained on synthetic dataset (i.e., Syn. scratch at the second row) and loss function of RegDA is used, denoted as $\mathcal{L}_{RegDA}$. The results show that each stage of ours affects performance improvement independently. The whole-body accuracy increases through both stages, but the performance improvement of individual keypoint between both stages has trade-off.

5 CONCLUSION

In this paper, we addressed the problem of limited data on precise pose estimation (PPE) and presented a novel deep learning-based model for PPE by employing transfer learning and domain adaptation techniques. For real-world applications, we validated the performance gain of our model, demonstrating the high accuracy on golf swing pose estimation with augmented keypoints up to 21 from original 10 keypoints for the task-specific target domain.

The direction of our future work is to develop a PPE model based on generative adversarial networks (GAN) with robust accuracy on various image types.

ACKNOWLEDGMENTS

This work was supported by the Institute for Information and Communications Technology Planning and Evaluation (IITP) under Grant 2021-0-00875 and 2021-0-00900.

REFERENCES

[1] [n.d.]. COCO Keypoints Evaluation. [Online]. Available: https://cocodataset.org/#keypoints-eval. Accessed on: Dec 20, 2021.

[2] Mykhaylo Andriluka, Leonid Pishchulin, Peter Gehler, and Bernt Schiele. June.2014. 2D Human Pose Estimation: New Benchmark and State of the Art Analysis. In IEEE Conference on Computer Vision and Pattern Recognition (CVPR). Columbus, Ohio, 3686–3693.

[3] Jinkun Cao, Hongyang Tang, Hao-Shu Fang, Xiaoyong Shen, Cewu Lu, and Yu-Wing Tai. OCT. 2019. Cross-Domain Adaptation for Animal Pose Estimation. In Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV). Seoul, Korea, 9498–9507.

[4] Yucheng Chen, Yingli Tian, and Mingyi He. MAR. 2020. Monocular human pose estimation: A survey of deep learning-based methods. Computer Vision and ImageUnderstanding192 (MAR. 2020), 102897–102919.

[5] Yilun Chen, Zhicheng Wang, Yuxiang Peng, Zhiqiang Zhang, Gang Yu, and JianSun. June. 2018. Cascaded pyramid network for multi-person pose estimation. In Proceedings of the IEEE conference on computer vision and pattern recognition. Salt Lake City, Utah, 7103–7112.

[6] Xiao Chu, Wei Yang, Wanli Ouyang, Cheng Ma, Alan L Yuille, and Xiaogang Wang.July. 2017. Multi-context attention for human pose estimation. In Proceedings of the IEEE conference on computer vision and pattern recognition. Honolulu, Hawaii,1831–1840.

[7] Junguang Jiang, Yifei Ji, Ximei Wang, Yufeng Liu, Jianmin Wang, and Mingsheng Long. June. 2021. Regressive Domain Adaptation for Unsupervised Keypoint Detection. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR). virtual only, 6780–6789.

[8] Masatoshi Kimura, Takayoshi Yamashita, Yuji Yamauchi, and Hironobu Fujiyoshi. SEP. 2015. Facial point detection based on a convolutional neural network with optimal mini-batch procedure. In2015 IEEE International Conference on Image Processing (ICIP). Quebec City, Canada, 2860–2864.

[9] Chen Li and Gim Hee Lee. June. 2021. From Synthetic to Real: Unsupervised Domain Adaptation for Animal Pose Estimation. In Proceedings of the IEEE/CVFConference on Computer Vision and Pattern Recognition (CVPR). virtual only, 1482–1491.

[10] Tsung-Yi Lin, Michael Maire, Serge J. Belongie, Lubomir D. Bourdev, Ross B. Gir-shick, James Hays, Pietro Perona, Deva Ramanan, Piotr Dollár, and C. Lawrence Zitnick. 2014. Microsoft COCO: Common Objects in Context. CoRRabs/1405.0312(2014). arXiv:1405.0312 http://arxiv.org/abs/1405.0312

[11] A. Newell, K. Yang, and J. Deng. OCT. 2016. Stacked Hourglass Networks for Human Pose Estimation. In Proceedings of the European Conference on Computer Vision (ECCV). Amsterdam, Netherlands, 483–499.

[12] Markus Oberweger, Paul Wohlhart, and Vincent Lepetit. DEC. 2015. Hands deep in deep learning for hand pose estimation. arXiv:1502.06807(DEC. 2015).

[13] Markus Oberweger, Paul Wohlhart, and Vincent Lepetit. DEC. 2015. Training a feedback loop for hand pose estimation. In Proceedings of the IEEE international conference on computer vision. Santiago, Chile, 3316–3324.

[14] George Papandreou, Tyler Zhu, Nori Kanazawa, Alexander Toshev, Jonathan Tompson, Chris Bregler, and Kevin Murphy. July. 2017. Towards accurate multi-person pose estimation in the wild. In Proceedings of the IEEE conference on computer vision and pattern recognition. Honolulu, Hawaii, 4903–4911.

[15] Leonid Pishchulin, Eldar Insafutdinov, Siyu Tang, Bjoern Andres, Mykhaylo Andriluka, Peter V Gehler, and Bernt Schiele. June. 2016. Deepcut: Joint subsetpartition and labeling for multi person pose estimation. In Proceedings of the IEEE conference on computer vision and pattern recognition. Las Vegas, Nevada,4929–4937.

[16] Ke Sun, Bin Xiao, Dong Liu, and Jingdong Wang. June. 2019. Deep High-Resolution Representation Learning for Human Pose Estimation. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR). Long Beach, California, 5693–5703.

[17] Yi Sun, Xiaogang Wang, and Xiaoou Tang. June. 2013. Deep convolutional network cascade for facial point detection. In Proceedings of the IEEE conference on computer vision and pattern recognition. Portland, Oregon, 3476–3483.

[18] Jonathan Tompson, Murphy Stein, Yann Lecun, and Ken Perlin. 2014. Real-time continuous pose recovery of human hands using convolutional networks. ACM Transactions on Graphics (ToG)33, 5 (2014), 1–10.

[19] Shih-En Wei, Varun Ramakrishna, Takeo Kanade, and Yaser Sheikh. June. 2016.Convolutional pose machines. In Proceedings of the IEEE conference on Computer Vision and Pattern Recognition. Las Vegas, Nevada, 4724–4732.

[20] Bin Xiao, Haiping Wu, and Yichen Wei. SEP. 2018. Simple Baselines for Human Pose Estimation and Tracking. In Proceedings of the European Conference on Computer Vision (ECCV). Munich, Germany, 466–481.

[21] Wei Yang, Shuang Li, Wanli Ouyang, Hongsheng Li, and Xiaogang Wang. OCT.2017. Learning feature pyramids for human pose estimation. In proceedings of the IEEE international conference on computer vision. Venice, Italy, 1281–1290.

[22] Yizhai Zhang, Kuo Chen, Jingang Yi, Tao Liu, and Quan Pan. FEB. 2016. Whole-Body Pose Estimation in Human Bicycle Riding Using a Small Set of Wearable Sensors. IEEE/ASME Transactions on Mechatronics21, 1, 163–174.

[23] Yuchen Zhang, Tianle Liu, Mingsheng Long, and Michael Jordan. June. 2019.Bridging theory and algorithm for domain adaptation. In International Conference on Machine Learning. Long Beach, California, 7404–7413.

[24] Xingyi Zhou, Qingfu Wan, Wei Zhang, Xiangyang Xue, and Yichen Wei. June.2016. Model-based deep hand pose estimation. arXiv:1606.06854(June. 2016).