Precise human pose estimation based on two-dimensional images for kinematic analysis

Rojas Martínez, Sara, Herrera, Diana Sofia, Arbeláez, Pablo

Sara Rojas Martínez, Diana Sofia Herrera, Pablo Arbeláez, "Precise human pose estimation based on two-dimensional images for kinematic analysis," Proc. SPIE 11330, 15th International Symposium on Medical Information Processing and Analysis, 113300F (3 January 2020); doi: 10.1117/12.2542539

SPIE.

Event: 15th International Symposium on Medical Information Processing and Analysis, 2019, Medelin, Colombia

Precise Human Pose Estimation based on 2-dimensional images for kinematic analysis

Sara Rojas Martínez*[a], Diana Sofia Herrera*[b], and Pablo Arbeláez[c]

[a,b,c]Departamento de Ingeniería Biomédica, Universidad de los Andes, Bogotá, Colombia

ABSTRACT

Human Pose Estimation (HPE) is a Computer Vision problem that has become increasingly popular over the last few years, with multiple applications in the medical field such as therapy using virtual and augmented reality, robot caregivers, virtual physical therapy and kinematic analysis. Nevertheless, all the machine learning algorithms developed for these applications are trained in small datasets with images captured in constrained scenarios and with information given by sensors, bounding the applicability of these methods. We developed a simple yet useful deep learning algorithm for Human Pose Estimation that uses as input only an image of a scene with people. The estimated position of the joints and body parts can be used to retrieve basic kinematic information from the people on the image that can be applied to the aforementioned medical applications. We focus on overcoming the limit of Human Pose Estimation algorithms due to jittering, aiming to preserve more precise pixel location. Thus, we explore different novel approaches to improve the precision of the existing state-of-the-art algorithms in keypoint estimation and evaluate them on COCO keypoint dataset, outperforming the current top methods. We hope our algorithm encourages the academic community to develop simpler but precise HPE algorithms for medical applications based on RGB images.

Keywords: Human Pose Estimation, jitter, kinematics, precise pixel location.

1. INTRODUCTION

Human Pose Estimation (HPE) is a Computer Vision problem where the goal is to estimate keypoint locations in joints and body parts to describe the position and orientation of the bodies (see Figure 1). This information allows us to recognize human actions and their interaction with a machine (5), opening a broad set of potential applications in the medical field such as therapy using virtual and augmented reality (29; 31), robot caregivers (6; 23; 25; 39; 40), virtual physical therapy and kinematic analysis (2; 16; 19; 22; 44).

Nevertheless, all the machine learning algorithms developed for these applications are trained in small datasets with images captured in constrained scenarios with a fixed set of conditions, and the data they use contain information given by sensors such as accelerometers, gyroscopes, electrodes, force sensors, among others. As a consequence, the applicability of these methods is limited to contexts similar to the ones existing in the datasets used to train the algorithms, bounding the potential applications for a broad range of people who could benefit from them but do not have access to advanced health care (17; 24).

We believe that machine learning algorithms developed to address tasks that might have a huge impact on people's life quality, such as human pose estimation for medical purposes, should be based on simpler information, so that they can be used in cheaper and simpler devices, making it available to benefit a broader range of users, including population with limited or difficult access to health services.

Therefore, we developed a simple yet useful deep learning algorithm for HPE that uses as input only an image of a scene with people. The estimated position of the joints and body parts can be used to retrieve basic kinematic information from the people on the image that can be applied to the aforementioned medical

a: E-mail: s.rojas12@uniandes.edu.co
b: E-mail: ds.herrera10@uniandes.edu.co
c: E-mail: pa.arbelaez@uniandes.edu.co
*Equal contribution.

15th International Symposium on Medical Information Processing and Analysis,
edited by Eduardo Romero, Natasha Lepore, Jorge Brieva, Proc. of SPIE Vol. 11330,
113300F · © 2020 SPIE · CCC code: 0277-786X/20/$21 · doi: 10.1117/12.2542539

applications. Our algorithm can be used in video-based tasks by performing the estimation frame by frame and analyzing the changes in the pose through time.

We focused into developing a robust and more precise method for HPE than the ones available in the state-of-the-art, which is why we used the COCO dataset for train and test stages, the largest, wildest, most diverse and thus more challenging dataset available for the task. We compare our results with other state-of-art methods trained with this dataset.

The challenge in the COCO dataset is to develop an "in the wild" pose estimator for images with up to 13 people. Algorithms trained with images taken from everyday environments with multiple instances are useful for real-life applications. However, real-time performance remains a challenge, since computational time increases with the number of people in an image (3). Furthermore, the amount of people in every image is unknown and each person might be in a different position and scale. Moreover, unpredictable interactions between people cause occlusion and obstruction between their limbs, making more difficult the association of parts for each person. These phenomena can be seen in medical applications such as therapy, human-machine interactions and even sports. As a consequence, the errors in HPE can be classified as inversion, swap, miss or jitter, where the last accounts for more than twice the other sources of error, reducing the performance of the state-of-art algorithms by more than 10% (38). The best-performing state-of-the-art algorithms for human pose estimation so far are based on convolutional neural networks (5; 20; 45). Nevertheless, none of the developed methods achieves precise pixel estimation, otherwise jitter would not represent such a massive source of error.

Therefore, we evaluate various novel approaches to improve precision during keypoint estimation, in other words, to reduce jitter in the results, because precision is crucial to provide accurate information about the position of the person for HPE medical applications, otherwise, automated HPE could lead to wrong diagnosis, therapies, indications or movements that might affect the patient.

The organization of the paper is as follows. Related work for this problem is described in Section 2. In Section 3 we describe the approaches we tested for the pose estimation task, Section 4 describes the experiments and evaluation results of our algorithms and, finally Section 5 presents some concluding remarks of our work.

Figure 1. Human Pose Estimation task through keypoint estimation in COCO dataset.

2. RELATED WORK

Various authors explored HPE using only RGB images as input for medical applications such as patient pose estimation in clinical environments (4), telerehabilitation using Virtual Reality (30), human-robot interaction (17) and performance analysis in physical activity (46). All of them identify the potential of pose estimation

Top-down	AP	First Stage	Second Stage	Training Details
CNP (5)	73	Faster-RCNN	2 Networks: +GlobalNet: ResNet Backbone+RefineNet: HyperNet	ResNet-50-based models takes about 1.5 day on eight NVIDIA Titan X Pascal GPUs.
CFN (12)	72,6	Faster-RCNN	2 streams: +Course stream: Inception modules +Fine stream: concatenation of multi-scale representations of the previous Inception modules	Implementation of CFN runs at 48 frames/sec on a TitanX GPU at the inference stage. Their method allows for real-time human pose estimation together with a fast person detector.
G-RMI (32)	64,9	Faster-RCNN	ResNet with 101 in a fully convolutional fashion	2 steps:+For person detector training we use 9 GPUs.+For pose estimator training we use two machines equipped with 8 GPUs.
Mask-RCNN (11)	63,1	Faster-RCNN	FCN	ResNet-50-FPN takes 32 hours in our synchronized 8-GPU and 44 hours with ResNet-101
RMPE (9)	61,8	SSTN + SPPE	Parametric Pose NMS	*

Table 1. Summary of top-down methods. mAP is shown for COCO dataset *test-dev*.

retrieved only from 2D images for everyday medical needs and to improve access and treatment for patients. Some of them also show that based on keypoint estimation of joints and relevant body parts, it is possible to estimate the kinematics of a body, which can be applied for telerehabilitation, physical activity and human-robot interaction (17; 30; 46). Nevertheless, the models they present are customized for small datasets with constrained context, therefore, their algorithms are not generalizable. On the other hand, algorithms developed for general HPE trained in larger datasets could be useful to robustly retrieve patients poses that can be used for any medical application afterwards.

Throughout the years, one of the greatest limits for Human Pose Estimation problem was the lack of quality data, due to the time-consuming annotations required. Although image and video datasets such as MPII Human Pose Dataset(1), Human3.6M (14), VGG Pose Dataset (42) and PennAction (43) were available in the first half of the decade, images are taken from a fixed perspective in constrained and few scenarios, and usually contain a single person, so they do not allow to train a generalizable model. Likewise, medical datasets for HPE were small-sized and built in constrained scenarios, usually created specifically for each algorithm developed (4; 17; 46). In 2016, COCO keypoints dataset and its associated challenge were published (20). They became a huge stimulus for human pose estimation research area in Computer Vision, because COCO is the greatest "in the wild" dataset for the task and became the state-of-art benchmark for Human Pose Estimation in the recent years.

In general, the algorithms developed to address the keypoint estimation task can be divided into two approaches, top-down and bottom-up pipelines. Top-down pipeline starts with human detection, where each instance is separated by a bounding box, and subsequently, the location of the keypoints are estimated for each human detected. On the contrary, bottom-up pipeline starts by finding the keypoints in the image and afterwards, joins the keypoints belonging to each person. Both have key limitations: the first depends on accurate human detection, while in the second, obstruction and occlusion complicates the separation of keypoints from adjacent people. Nevertheless, the first top-down algorithms with considerable results (5; 9; 11; 12; 32) have better performance than contemporary bottom-up approaches (3; 13; 26; 27; 36) in the challenge. The summary of the basic state-of-the-art methods for each pipeline are presented in Tables 1 and 2 respectively.

Additionally, other contemporary methods propose different approaches to tackle the task. One of them is to jointly optimize data augmentation and network training using adversarial networks, to improve the overall performance of existing architectures in the standard datasets (35). On the other hand, Luvizon et al. (21) use Multitask Deep Learning to address pose estimation and action recognition with the same architecture. The

Buttom-up	AP	Single Stage	Training Details
PAFs (3)	61,8	2 braches:+2D confidence maps of the body locations+2D vector fields of part affinities	The runtime analysis is performed on a laptop with one NVIDIA GeForce GTX-1080 GPU. Their method has achieved the speed of 8.8 fps for a video with 19 people.
Associative embedding (26)	65,5	SHN and associative embedding	Training is done from scratch on MS COCO for three days, and then fine tuned on PASCAL VOC train for 12 hours.
DeepCut (36)	*	3 problems jointlt:+Body part detector +Partition of body part belong to the same person.+Label each selected body part.	*
DeeperCut (13)	*	3 new modules.	They use NVIDIA Tesla K40 GPU with 12 GB RAM

Table 2. The summary of the Bottom-up methods. mAP is shown for COCO dataset *test-dev*.

winners of the COCO 2017 keypoint challenge propose a Cascaded Pyramid Network in which the initial stage estimates the easy keypoints, GlobalNet, and the second one, RefineNet, focuses on the hard keypoints (5). Other methods add intermediate information to use the human structure as a constraint; (41) uses a Deeply Learned Compositional Model whereas in (28) the authors develop a Parsing Induced Learner by merging a parsing encoder and a pose model parameter adapter. Likewise, (15) creates a Multi-Scale Structure-Aware Network for Human Pose Estimation to seize the intermediate information and provide consistency through keypoints and scales. As can be seen, most of the architectures used in the task are complex and have a high computational cost, but (45) proposes a simple architecture using the most common backbone network for image feature extraction, ResNet, and a few deconvolutional layers over the last convolution stage, as the simplest method to obtain heatmaps from low resolution, deep features. They achieve the second place in the COCO keypoint challenge 2018.

However, most of these works do not have a publicly available code and present their results in the MPII dataset, which has less than a third of annotated instances than COCO. Therefore, a direct comparison can not be made.

More importantly, according to the results of the COCO challenge, none of the state-of-art methods addresses the keypoint localization task with precision, as seen by an error greater than 10% due to jittering (38). Thereafter, improving the current human pose estimation algorithms with an architecture designed for precision could lead to an algorithm that tackles the jittering issue in human pose estimation problem, outperforming all the state-of-the-art methods.

3. APPROACH

3.1 Dataset description

We use the COCO 2017 keypoints dataset for all our experiments with the official split of 118K train, 5K validation and 40K test images (20). These images contain 155,000 person instances labeled with over 1 million keypoints classified in 17 types (20). Instances have different sizes and images include multiple perspectives, occlusions and multi-instance examples with up to 13 people per image. Thus, COCO is the greatest "in the wild" dataset for the task and became the state-of-art benchmark for Human Pose Estimation in the recent years.

The task on COCO is retrieved as keypoint detection because it involves people detection and keypoint estimation simultaneously. They introduced their own evaluation metric for the challenge based on a similarity measure between ground truth objects and predicted objects, as in a detection task. It is defined as *object*

keypoint similarity (OKS) and ranges from 0 (poor match) to 1 (perfect match), according to the distance between the estimated and annotated keypoints under a Gaussian distribution whose deviation depends on the type of keypoint. To determine correct predictions, a threshold in the OKS is set; by varying this threshold, a precision-recall curve is calculated. The area under the curve is the average precision (AP), the most important metric in the detection task (20).

3.2 Proposed method

In our baseline we use Resnet with 50 layers to produce features and subsequently, three deconvolutions to build heatmaps over those features, inspired by (45). We explore the following modifications to our baseline:

1. Use the same architecture of (45) with dilated convolutions.
2. Introduce a Global Convolution Network (GCN) (34) module between layers of the encoder's (45) architecture.
3. Use an architecture developed for the contour detection task (7) with Resnet backbone (Crisp keypoints).
4. Repeat the architecture of (45) twice in order to refine heatmaps as (11) (FishNet).
5. Add a parsing module (28) to refine the keypoints estimation process by constraining them to be inside the segmentation region of the correct body part.

4. EXPERIMENTAL VALIDATION

4.1 Experimental setup

We implement our method using Pytorch (33) and perform all our experiments using the COCO keypoint 2017 dataset (20). All methods were trained during 140 epochs with a batch size of 32. Data augmentation was performed with rotation, scaling and cropping. We use ADAM for optimization (18).

For all our experiments, we used a ResNet 50 backbone and an image input size of 256x192.

4.2 Network architectures

A detailed description for each network architecture we tested is stated below:

4.2.1 Simple baseline with dilated convolutions

For the first experiment we replace the convolutions in the architecture by dilated convolutions to increase the receptive field of the model and improve its scale invariance (see Figure 2). We tested kernel sizes 2, 3 and 4.

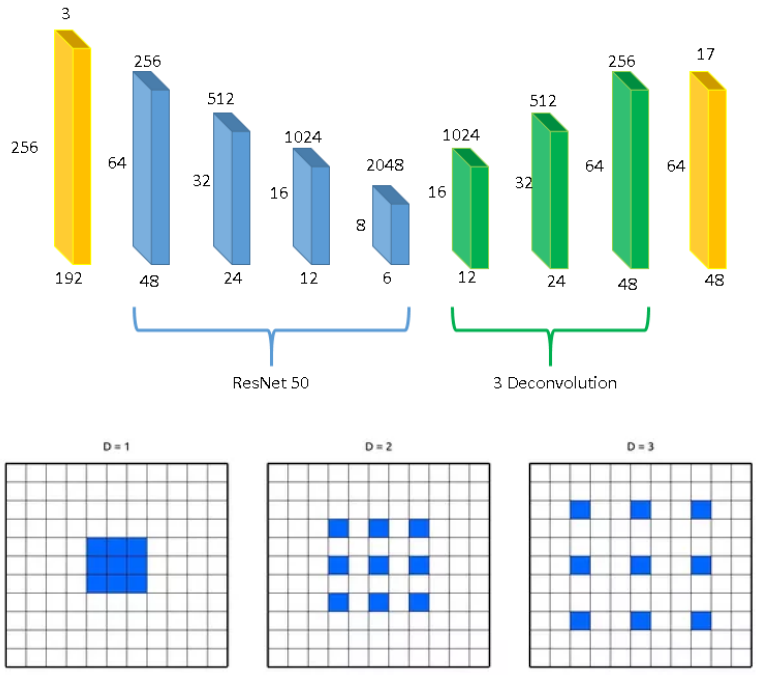

Figure 2. Simple baseline with dilated convolutions. The lower diagram shows the representation of the kernel size in the dilated convolutions.

4.2.2 Simple baseline with GCN

Secondly, we added GCN modules in the downsampling path to increase the receptive field of every stage. Each red circle in figure 3 represents a GCN module. We performed this experiment with different kernel sizes for the module.

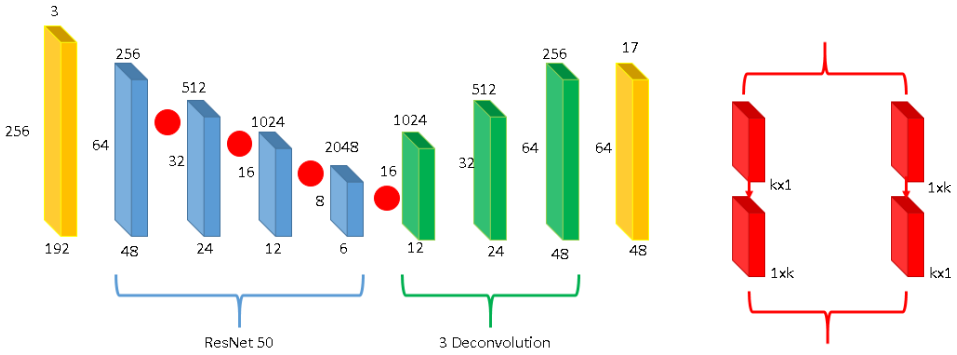

Figure 3. GCN modules within the architecture are shown as red dots. The architecture of the module is shown in the rightmost red diagram.

4.2.3 Crisp keypoints

A task in which an output with a very precise estimation is needed is contour detection, for whom (8) proposes a top-down/bottom-up architecture with deep supervision to predict even the smallest details on the contours. Thus, we adapt their architecture for fine contour detection in our model, creating a bottom-up top-down network. Basically, we introduce skip connections between modules of the downsampling and upsampling paths

as shown in figure 4, aiming to constrain the model to estimate the keypoints correctly in every stage. Two versions of this model were tested, one in which only the final loss was calculated with the output of the last layer in the upsampling path, and one in which in the loss was calculated using the concatenation from the outputs of every layer on the upsampling path.

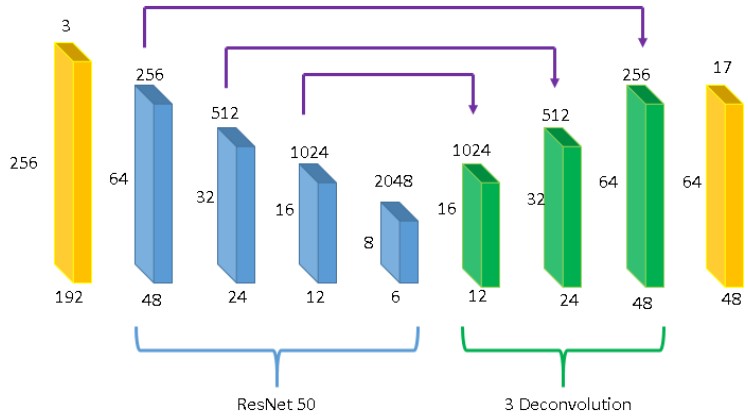

Figure 4. "Crisp keypoints". Bottom-up top-down network with skip connections between modules of the downsampling and upsampling paths.

4.2.4 FishNet

As another approach, we reproduce the baseline architecture twice in sequence to refine the keypoints estimated, inspired by (11). We evaluated to types of input for the second network, one with the 17 channels from the heatmaps estimated and another one containing 20 channels: 17 from the heatmaps estimated in the first network plus the RGB channels from the original image, to provide guidance to the refinement process.

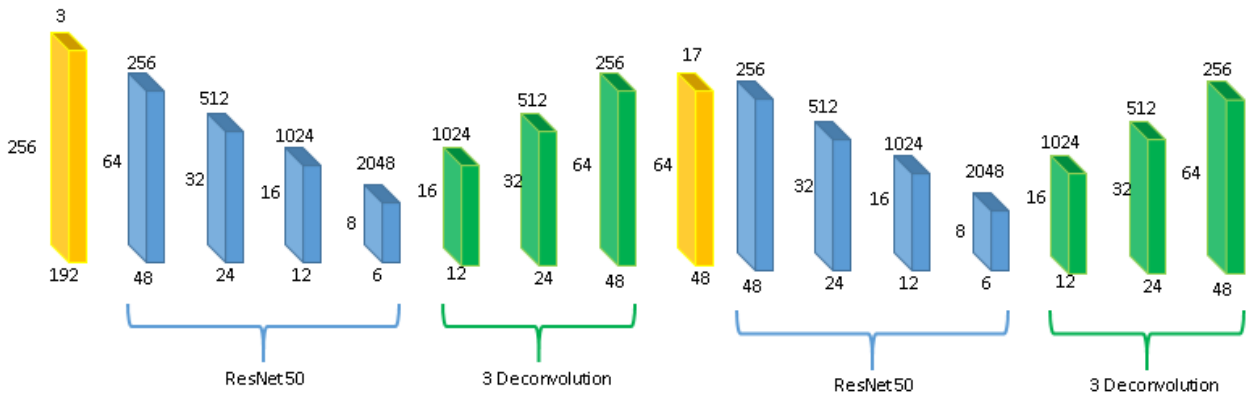

Figure 5. "FishNet". Sequence of the same architecture twice to refine the keypoints estimated.

4.2.5 Parsing module

Lastly, we add a parsing module to the network to take into account parsing information in the pose estimation process, aiming to refine the keypoints estimated by constraining them to be inside the segmentation mask of the correct body part, inspired by (28). To this goal, we used the DensePose dataset (10), because it contains

parsing annotations in COCO images as the one in figure 6. Nevertheless, these annotations include only 39,210 samples instead of the 149,813 instances used in the standard COCO dataset for train and 6,352 samples instead of 2,243 samples for validation. Hence, the performance of this approach is not directly comparable with the others.

Figure 6. An example of the parsing annotations in the full-size and bounding-box-cropped image.

4.3 Results

The results from the experiments performed are shown in table 3.

Among all the methods, using dilated convolutions in the Simple Baseline from (45) performs considerably well when using a kernel of size 3. The advantage of atrous convolutions over traditional ones relies in the fact that it allows to extract information at different scales, improving the detection on keypoints that are occluded or deformed due to perspective.

Using GCN does not improve the performance of the network because overall context is not imperative for keypoint detection, especially considering the fact that instances are initially cropped within their bounding boxes. On the other hand, a careful insight into the source of error of the results (see Fig.7) obtained using the crisp keypoints approach, proves that the method locates with a higher precision the keypoints estimated, as we expected, by means of a reduction in more than 1% in jitter. The remaining error types do not vary, outlining the limitation of the method to improve further its performance.

Adding the parsing module reduces significantly the performance due to the smaller amount of samples available for training, almost to a third of the amount used to train the other approaches. Thus, to perform a fair comparison we use the same subset of COCO with and without the parsing module (See Fig. 3) and the first experiment outperformed the second one, proving the usefulness of parsing in the task. It is interesting to note that the parsing approach reduces significantly the performance of the algorithm for medium instances, whose score is abnormally smaller than the one in large instances, indicating a lack of precision of the method for instances without the whole set of keypoints.

Lastly, the FishNet 20 architecture gives the best results because it produces a refinement process in which the network repeats the pose estimation with the heatmaps estimated in the first halve using the input image as a model again. This network clearly reduces the jitter in the results because the scores in both medium and large scale instances improve, demonstrating the efficacy of the method in improving the precision on keypoint estimation.

According to the results from all the approaches evaluated, our final method merges the FishNet architecture with dilated convolutions. The corresponding results are presented in table 4. Moreover, thanks to the tool of (37) we analyze the main sources of error in our final method and verify that the major improvement is due to a reduction in jitter and miss errors, with an improvement in 0.7 points in both. The summary of the analysis is presented in figure 7. Overall, FishNet architecture uses the simple baseline from (45) twice, proving the usefulness of this simple approach in the task.

Experiment	Epoch	AP	AP .5	AP. 75	AP (M)	AP (L)	AR	AR .5	AR .75	AR (M)	AR (L)
1. Dilated convolution	-	-	-	-	-	-	-	-	-	-	-
Dilated convolution = 2	140	0.718	0.915	0.794	0.689	0.763	0.749	0.926	0.817	0.717	0.798
Dilated convolution = 3	140	**0.723**	0.925	0.794	0.696	0.765	**0.754**	0.932	0.818	0.722	0.802
Dilated convolution = 4	140	0.701	0.905	0.782	0.674	0.745	0.732	0.920	0.801	0.700	0.781
2. GCN	-	-	-	-	-	-	-	-	-	-	-
Large kernel = 7	140	0.701	0.905	0.782	0.674	0.745	0.732	0.920	0.801	0.700	0.781
Large kernel = 9	40	0.660	0.892	0.736	0.635	0.698	0.693	0.900	0.761	0.662	0.740
Large kernel = 13	140	**0.719**	0.925	0.793	0.690	0.762	**0.749**	0.931	0.816	0.718	0.797
Large kernel = 15	140	0.702	0.915	0.781	0.673	0.745	0.735	0.923	0.803	0.702	0.784
3. Crips keypoints	-	-	-	-	-	-	-	-	-	-	-
Crisp_1Loss	109	**0.688**	0.896	0.763	0.666	0.738	**0.727**	0.910	0.792	0.696	0.774
Crisp_cat_1loss	81	0.680	0.893	0.749	0.650	0.724	0.713	0.907	0.775	0.678	0.765
4. FishNet	-	-	-	-	-	-	-	-	-	-	-
FishNet 17	140	0.728	0.925	0.804	0.701	0.771	0.759	0.931	0.827	0.728	0.807
FishNet 20	140	**0.731**	0.925	0.805	0.707	0.774	**0.763**	0.933	0.829	0.733	0.810
5. Parsing	-	-	-	-	-	-	-	-	-	-	-
Parsing module	140	0.442	0.505	0.481	0.069	0.752	0.453	0.503	0.487	0.065	0.776
Without parsing module	140	0.418	0.494	0.471	0.067	0.709	0.430	0.497	0.471	0.061	0.737

Table 3. Results of the experiments performed. AP is shown for COCO dataset *test-dev*.

FishNet 20+dilated convs	Epoch	AP	AP .5	AP. 75	AP (M)	AP (L)	AR	AR .5	AR .75	AR (M)	AR (L)
FishNet 20	140	0.731	**0.925**	0.805	0.707	0.774	0.763	**0.933**	0.829	0.733	0.810
FishNet 20+dilated k=2	140	0.734	**0.925**	**0.814**	0.706	**0.777**	0.764	**0.933**	**0.832**	0.733	**0.812**
FishNet 20+dilated k=3	140	0.732	**0.925**	0.805	0.705	0.773	0.763	0.932	0.828	0.732	0.810
FishNet 20+dilated k=4	140	**0.735**	**0.925**	**0.814**	**0.708**	0.775	**0.765**	0.932	0.831	**0.734**	**0.812**
FishNet 20+dilated k=5	140	0.671	0.904	0.748	0.652	0.706	0.708	0.911	0.777	0.683	0.747

Table 4. Results of our final model which merges the FishNet architecture with dilated convolutions. AP is shown for COCO dataset *test-dev*.

4.4 Comparison with the State-of-the-Art

To make a fair comparison, AP for CPN and Simple Baseline methods are the scores obtained when training the available codes with a ResNet 50 backbone and image input size of 256x192, the same parameters that we used for our experiments.

Method	AP
CPN (5)	0.697
Simple Baselines (45)	0.715
*Simple Baselines with Dilated Convolution	0.723
*Simple Baselines with GCN	0.719
*Crisp keypoints	0.688
*FishNet 20	0.730
*Parsing module	0.442
*FishNet 20 + dilated convolutions k=4	**0.735**

Table 5. Comparison made with state-of-the-art methods. AP is shown for COCO dataset *test-dev*. Our experiments have a * at the beginning of the method.

This comparison demonstrates that using dilated convolutions with a kernel size of 4 in a FishNet architecture with a baseline inspired by (45), outperforms not only the other experiments we explored and the baseline (45) by 2%, but also CPN (5) by 3.8%, the winners of the COCO 2017 keypoint challenge, when their methods are evaluated with the same conditions of our experiments. Hence, we demonstrate that for the keypoint estimation task it is useful to add information of different scales for each instance, plus a refinement process of the resulting heatmaps with the same architecture and the input image as a guide. We hypothesize this works because perspective generates different scales within a 2D image of the human body and refinement constraints the estimated keypoints to locate more precisely.

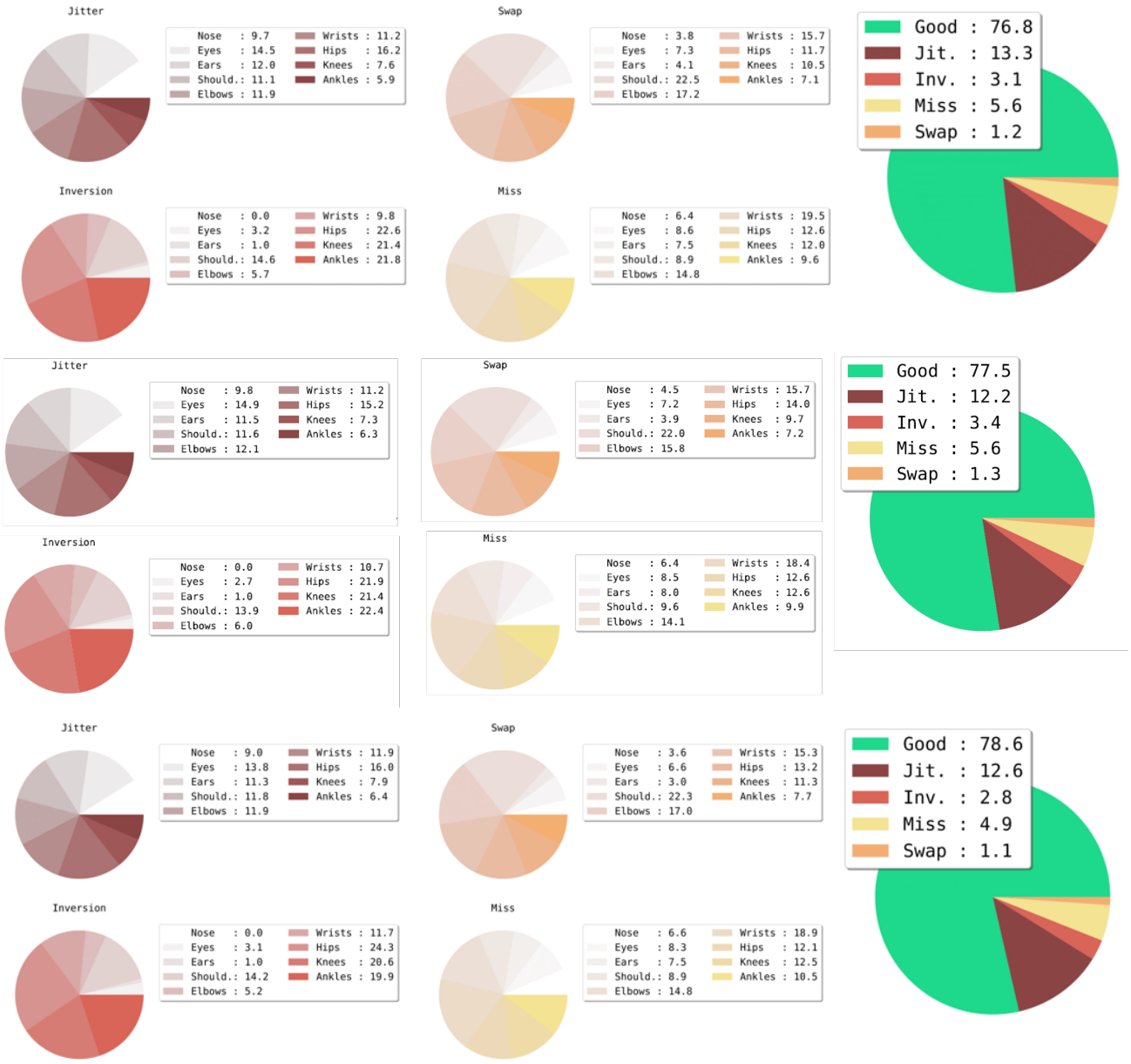

Figure 7. Up: Error analysis of the baseline (45). Middle: Error analysis of the crisp keypoints approach. Down: Error analysis of our final model FishNet 20 with dilated convolutions using a kernel size of 4. The analysis uses a 0.75 threshold for the OKS

5. CONCLUSIONS

By changing the convolutions within the Simple Baseline method for atrous convolutions and repeating the architecture twice to generate a refinement process of the heatmaps initially estimated, we outperform the baseline, the other experiments evaluated and the the winner of the COCO 2017 keypoint challenge, CPN (5), when their available code is evaluated with the same parameters used in our experiments. This fact outlines the usefulness of adding information at different scales and a refinement stage in which the input image is used to force the initial heatmaps estimated to locate more precisely, in order to improve the performance of the existing algorithms in the human pose estimation task due to a reduction in jitter and miss errors.

In this way, we prove the potential of simpler algorithms for HPE. The retrieval of more precise keypoints is essential for medical applications based on HPE. At the same time, the impact and applicability of methods that use only bidimensional images as an input, instead of sensor information in medical contexts is broader. The potential of our algorithm relies on the fact that it is more generalizable than those developed with medical datasets that contain fewer data with fixed settings. Additionally, the novel architecture we propose focuses on precision, which is determinant to give appropriate virtual therapy, caregiving or diagnosis. Overall, we propose an insight into the application of deep learning algorithms trained in larger and general datasets for medical tasks based on HPE. Our approach, developed in such settings, can be applied in applications for broad populations with limited or difficult access to health services, due to its simplicity yet high performance. We hope this work encourages other researchers in the field to use powerful algorithms based on simple information and trained in larger datasets that can be more easily transferred to medical applications for the people who need them.

References

[1] M. Andriluka, L. Pishchulin, P. Gehler, and B. Schiele. 2d human pose estimation: New benchmark and state of the art analysis. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2014.

[2] S. Bragaglia, S. Di Monte, and P. Mello. A distributed system using ms kinect and event calculus for adaptive physiotherapist rehabilitation. In *2014 Eighth International Conference on Complex, Intelligent and Software Intensive Systems*. IEEE, 2014.

[3] Z. Cao, T. Simon, S.-E. Wei, and Y. Sheikh. Realtime multi-person 2d pose estimation using part affinity fields. In *CVPR*, volume 1, page 7, 2017.

[4] K. Chen, P. Gabriel, A. Alasfour, and C. Gong. Patient-specific pose estimation in clinical environments. *IEEE J Transl Eng Health Med*, 6.2101111, 2018.

[5] Y. Chen, Z. Wang, Y. Peng, Z. Zhang, G. Yu, and J. Sun. Cascaded pyramid network for multi-person pose estimation. *arXiv preprint arXiv:1711.07319*, 2017.

[6] K. Dautenhahn and I. Werry. Issues of robot-human interaction dynamics in the rehabilitation of children with autism. *Proc. From animals to animats*, 6:519–528, 2000.

[7] R. Deng, C. Shen, S. Liu, H. Wang, and X. Liu. Learning to predict crisp boundaries. *arXiv preprint arXiv:1807.10097*, 2018.

[8] R. Deng, C. Shen, S. Liu, H. Wang, and X. Liu. Learning to predict crisp boundaries. In *The European Conference on Computer Vision (ECCV)*, September 2018.

[9] H. Fang, S. Xie, Y.-W. Tai, and C. Lu. Rmpe: Regional multi-person pose estimation. In *The IEEE International Conference on Computer Vision (ICCV)*, volume 2, 2017.

[10] R. A. Güler, N. Neverova, and I. Kokkinos. Densepose: Dense human pose estimation in the wild. *CoRR*, abs/1802.00434, 2018.

[11] K. He, G. Gkioxari, P. Dollár, and R. Girshick. Mask r-cnn. In *Computer Vision (ICCV), 2017 IEEE International Conference on*, pages 2980–2988. IEEE, 2017.

[12] S. Huang, M. Gong, and D. Tao. A coarse-fine network for keypoint localization. In *The IEEE International Conference on Computer Vision (ICCV)*, volume 2, 2017.

[13] E. Insafutdinov, L. Pishchulin, B. Andres, M. Andriluka, and B. Schiele. Deepercut: A deeper, stronger, and faster multi-person pose estimation model. In *European Conference on Computer Vision*, pages 34–50. Springer, 2016.

[14] C. Ionescu, D. Papava, V. Olaru, and C. Sminchisescu. Human3.6m: Large scale datasets and predictive methods for 3d human sensing in natural environments. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 36(7):1325–1339, jul 2014.

[15] L. Ke, M.-C. Chang, H. Qi, and S. Lyu. Multi-scale structure-aware network for human pose estimation. In *The European Conference on Computer Vision (ECCV)*, September 2018.

[16] C. Kertesz. Physiotherapy exercises recognition based on rgb-d human skeleton models. In *2013 European Modelling Symposium*. IEEE, 2013.

[17] R. Khonasty, M. Carmichael, D. Liu, and K. Waldron. Upper body pose estimation utilizing kinematic constraints from physical human-robot interaction. *Australian Robotics Automation Association*, 2017.

[18] D. P. Kingma and J. Ba. Adam: A method for stochastic optimization. *CoRR*, abs/1412.6980, 2014.

[19] J. Lin and D. Kulic. Human pose recovery using wireless inertial measurement units. *Physiological measurement*, 33.2:2099, 2012.

[20] T.-Y. Lin, M. Maire, S. Belongie, J. Hays, P. Perona, D. Ramanan, P. Dollár, and C. L. Zitnick. Microsoft coco: Common objects in context. In *European conference on computer vision*, pages 740–755. Springer, 2014.

[21] D. C. Luvizon, D. Picard, and H. Tabia. 2d/3d pose estimation and action recognition using multitask deep learning. In *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2018.

[22] E. Marchand, H. Uchiyama, and F. Spindler. Pose estimation for augmented reality: a hands-on survey. *IEEE transactions on visualization and computer graphics*, 22(12):2633–2651, 2016.

[23] R. Mead, E. Wade, P. Johnson, A. St.Clair, S. Chen, and M. Mataric. An architecture for rehabilitation task practice in socially assistive human-robot interaction. In *19th International Symposium in Robot and Human Interactive Communication*. IEEE, 2010.
[24] D. Mehta, S. Sridhar, O. Sotnychenko, H. Rhodin, M. Shafiei, H.-P. Seidel, W. Xu, D. Casas, and C. Theobalt. Vnect: Real-time 3d human pose estimation with a single rgb camera. *ACM Transactions on Graphics*, 36.4:44, 2017.
[25] M. Mihelj. Human arm kinematics for robot based rehabilitation. *Robotica*, 24.3:377–383, 2006.
[26] A. Newell, Z. Huang, and J. Deng. Associative embedding: End-to-end learning for joint detection and grouping. In *Advances in Neural Information Processing Systems*, pages 2274–2284, 2017.
[27] A. Newell, K. Yang, and J. Deng. Stacked hourglass networks for human pose estimation. In *European Conference on Computer Vision*, pages 483–499. Springer, 2016.
[28] X. Nie, J. Feng, Y. Zuo, and S. Yan. Human pose estimation with parsing induced learner. In *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2018.
[29] S. Obdrzalek, G. Kurillo, R. Bajcsy, E. Seto, H. Jimison, and M. Pavel. Accuracy and robustness of kinect pose estimation in the context of coaching of elderly population. In *2012 Annual International Conference of the IEEE Engineering in Medicine and Biology Society*. IEEE, 2012.
[30] S. Obdrzalek, G. Kurillo, J. Han, T. Abresch, and R. Bajcsy. Real-time human pose detection and tracking for tele-rehabilitation in virtual reality. *Medicine Meets Virtual Reality 19*, 19:320–324, 2012.
[31] S. Obdrzalek, G. Kurillo, J. Han, T. Abresch, R. Bajcsy, et al. Real-time human pose detection and tracking for tele-rehabilitation in virtual reality. *Studies in health technology and informatics*, 173:320–324, 2012.
[32] G. Papandreou, T. Zhu, N. Kanazawa, A. Toshev, J. Tompson, C. Bregler, and K. Murphy. Towards accurate multiperson pose estimation in the wild. *arXiv preprint arXiv:1701.01779*, 8, 2017.
[33] A. Paszke, S. Gross, S. Chintala, G. Chanan, E. Yang, Z. DeVito, Z. Lin, A. Desmaison, L. Antiga, and A. Lerer. Automatic differentiation in pytorch. 2017.
[34] C. Peng, X. Zhang, G. Yu, G. Luo, and J. Sun. Large kernel matters—improve semantic segmentation by global convolutional network. In *Computer Vision and Pattern Recognition (CVPR), 2017 IEEE Conference on*, pages 1743–1751. IEEE, 2017.
[35] 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 *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2018.
[36] L. Pishchulin, E. Insafutdinov, S. Tang, B. Andres, M. Andriluka, P. V. Gehler, and B. Schiele. Deepcut: Joint subset partition and labeling for multi person pose estimation. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 4929–4937, 2016.
[37] M. R. Ronchi and P. Perona. Benchmarking and error diagnosis in multi-instance pose estimation. In *Computer Vision (ICCV), 2017 IEEE International Conference on*, pages 369–378. IEEE, 2017.
[38] M. Ruggero. Coco and places visual recognition challenges workshop. http://presentations.cocodataset.org/COCO17-Keypoints-Overview.pdf, 2017.
[39] E. Seemann, N. Kai, and S. Rainer. Head pose estimation using stereo vision for human-robot interaction. In *Sixth IEEE International Conference on Automatic Face and Gesture Recognition, 2004*. IEEE, 2004.
[40] M. Svenstrup, S. Hansen, H. Andersen, and T. Bak. Pose estimation and adaptive robot behaviour for human-robot interaction. In *2009 IEEE International Conference on Robotics and Automation*. IEEE, 2009.
[41] W. Tang, P. Yu, and Y. Wu. Deeply learned compositional models for human pose estimation. In *The European Conference on Computer Vision (ECCV)*, September 2018.
[42] O. University. Vgg human pose estimation datasets. https://www.robots.ox.ac.uk/~vgg/data/pose/index.html.
[43] M. Z. Weiyu Zhang and K. Derpanis. From actemes to action: A strongly-supervised representation for detailed action understanding. http://dreamdragon.github.io/PennAction/, 2013.
[44] Wrnch. Frictionless motion capture and activity recognition. https://wrnch.ai/, 2018.
[45] B. Xiao, H. Wu, and Y. Wei. Simple baselines for human pose estimation and tracking. *arXiv preprint arXiv:1804.06208*, 2018.
[46] D. Zecha, M. Einfalt, C. Eggert, and L. Rainer. Kinematic pose rectification for performance analysis and retrieval in sports. In *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2018.