OpenPose: Whole-Body Pose Estimation

Ginés Hidalgo Martínez

April 2019

Technical Report Number:
CMU-RI-TR-19-15

Thesis Committee:
Yaser Sheikh
Kris Kitani
Aayush Bansal

Robotics Institute
Carnegie Mellon University
Pittsburgh, Pennsylvania 15213

Acknowledgements

First of all, I would like to thank my current and former advisors, Yaser Sheikh and Fernando De La Torre, who have supported my research career and provided me with an invaluable source of insight, ideas, and practical guidance from beginning to end. They have consistently put in their time and energy through both successes and failures. I am grateful that I had the chance to work with them, and I am proud of the work we have done together over the last 5 years at CMU.

I also had the privilege of working and discussing with many other researchers, including my thesis committee members Kris Kitani and Aayush Bansal. I have been fortunate to be surrounded by many supportive friends and colleagues at CMU and elsewhere. I would like to thank Francisco Vicente, Stanislav Panev, Yaadhav Raaj, Hanbyul Joo, Tomas Simon, Ramon Sanabria, Donglai Xiang, Xiu Li, Haroon Idrees, and Xuehan Xiong for many helpful discussions. I would also like to thank my parents, Gines and Maria Concepcion, and my siblings, Magdalena and Miguel Angel, for their support in my life choice and for their years of love. Finally but not least, I would like to especially thank my fiancee Angelica Herrera, who has supported me through successes and failures, and who has been a key element in most of my successes.

Abstract

We present the first single-network approach for 2D whole-body (body, face, hand, and foot) pose estimation, capable of detecting an arbitrary number of people from in-the-wild images. Our method maintains constant real-time performance regardless of the number of people in the image. This network is trained in a single stage using multi-task learning and an improved architecture, which account for the inherent scale difference between body/foot and face/hand keypoints. Our approach considerably improves upon the only known work in whole-body pose estimation (our previous work, the original OpenPose [1]) in both speed and global accuracy. Unlike the original OpenPose, our new method does not need to run an additional network for each hand and face candidate, making it substantially faster for multi-person scenarios. This work directly results in a reduction of computational complexity for applications that require 2D whole-body information (e.g., re-targeting). In addition, it yields higher accuracy, especially for occluded, blurry, and low resolution faces and hands. Our code, trained models, and validation benchmarks will be publicly released as a baseline for future work in the area.

Contents

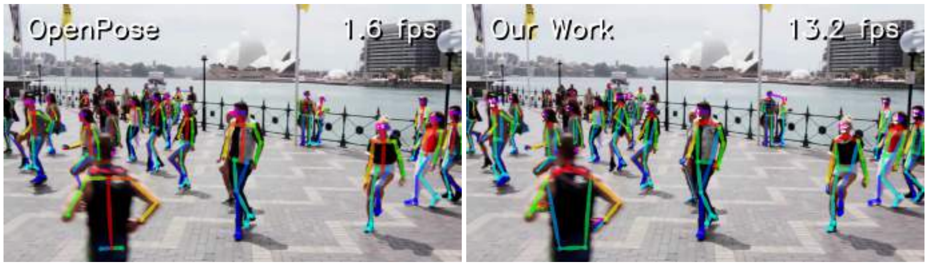


Figure 2: We present the first single-network approach for whole-body pose estimation, with real-time performance that is independent of the number of people in the image. Our work considerably outperforms the current state-of-the-art (our previous work, the original OpenPose [1]) in runtime performance, while it also slightly improves its keypoint accuracy. To avoid confusion, we will denote the original OpenPose as simply "OpenPose", and our new work as "Our work".

1 Introduction

Human keypoint estimation has been an open problem for decades in the research community. Initially, efforts were focused on facial alignment (i.e., face keypoint detection) [2, 3, 4, 5, 6, 7]. Gradually, the problem evolved into single and multi-person human pose estimation in-the-wild, including body and foot keypoints [8, 9, 10, 11, 12]. A more recent and challenging problem has targeted hand keypoint detection [13, 14, 15]. The next logical step is the integration of all of these keypoint detection tasks within the same algorithm, leading to "whole-body" or "full-body" (body, face, hand, and foot) pose estimation [16, 1].

There are several applications that can immediately take advantage of whole-body keypoint detection, such as re-targeting and 3D human keypoint and mesh reconstruction [17, 18, 19, 20, 21]. In general, almost any method that uses body information could also benefit from face, hand, and foot detection, such as person re-identification, tracking, or action recognition [22, 23, 24, 25]. Despite these needs, the only existing method providing whole-body pose estimation is our previous work, the original OpenPose [1], which follows a multi-stage approach. It first obtains all body poses from an input image in a bottom-up fashion [10] and then runs additional face and hand keypoint detectors [13] for each detected person. As a multi-network approach, it directly uses the existing body, face, and hand keypoint detection algorithms. However, it suffers from early commitment: if the body-only detector fails, there is no recourse to recovery, and it is prone to do so when only a face or a hand are visible in the image. In addition, its runtime is proportional to the number of people in the image, making whole-body OpenPose prohibitively costly for multi-person and real-time applications. A single-stage method, estimating whole-body parts of multiple people in a single inference, would be more attractive as it would yield a fixed inference runtime, independent to the number of people in the scene.

Unfortunately, there is an inherent scale difference between body/foot and face/hand keypoint detection. The former requires a large receptive field to learn the complex interactions across people (contact, occlusion, limb articulation), while the latter requires high facial and hand resolution. This scale issue has two critical consequences. First, datasets with full-body annotations in-the-wild do not currently exist. The characteristics of each set of keypoints result in different kinds of datasets. Body datasets predominately contain images with multiple people, usually resulting in fairly low facial and hand resolution, while face and hand datasets mostly contain images with a single, cropped face or hand. Secondly, the architecture design of a single-network model must differ from that of the state-of-the-art keypoint detectors in order to simultaneously offer high-resolution and a large receptive field while improving the inference runtime of multi-network approaches.

To overcome the dataset problem, we resort to multi-task learning (MTL). MTL is a classic machine learning technique [26, 27, 28] in which related learning tasks are solved simultaneously, while exploiting commonalities and differences across them. MTL has been successful in training a combined body-foot keypoint detector [1]. Nevertheless, this approach does not generalize to whole-body estimation because of the underlying scale problem. The major contributions of this paper are summarized as follows:

- **Novelty:** We present a MTL approach which, combined with an updated model architecture design, is able to train a united model out of various keypoint detection tasks with different scale properties. This results in the first single-network approach for whole-body multi-person pose estimation.

- **Speed:** At test time, our single-network approach provides a constant real-time inference regardless of the number of people detected, and it is approximately p times faster than the state-of-the-art (original OpenPose [1]) for images with p people. In addition, it is trained in a single stage, rather than requiring independent network training for each individual task. This reduces the total training time approximately by half.

- **Accuracy:** Our new approach also yields higher accuracy than that of the previous OpenPose, especially for face and hand keypoint detection, generalizing better to occluded, blurry, and low resolution faces and hands.

2 Related Work

2.1 Face Keypoint Detection

Also known as landmark detection or face alignment. It has a long history in computer vision, and many approaches have been proposed to tackle it. These approaches can be divided into two categories: template fitting [2, 29, 4, 6, 30] and regression-based methods [3, 5, 7]. Template fitting methods build face templates to fit input images, usually exploiting a cascade of regression functions. Regression methods, on the other hand, are based on Convolutional Neural Networks (CNNs), usually applying convolutional heatmap regression. They operate in a similar fashion to that of body pose estimation.

2.2 Body Keypoint Estimation

With the face alignment problem solved, efforts have moved toward single-person pose estimation. The initial approaches performed inference over both local observations on body parts and their spatial dependencies, either based on tree-structured graphical models [8, 31, 32, 33, 34] or non-tree models [12, 35, 36, 37, 38]. The popularity of CNNs and the release of massive annotated datasets (COCO [39] and MPII [40]) have resulted in a significant boost of the accuracy of single-person estimation [9, 41, 42, 43, 44, 45, 46, 47], and have enabled multi-person estimation. The latter is traditionally divided into top-down [11, 48, 49, 50, 51, 52, 53, 54] and bottom-up [10, 55, 56, 57, 58] approaches.

2.3 Foot Keypoint Estimation

In our previous work, OpenPose [1], we released the first foot dataset, annotated from a subset of images of the COCO dataset. We also trained the first combined body-foot keypoint detector, by applying a naive multi-task learning loss technique. Our new method is an extension of this work, mitigating its limitations and enabling it to generalize to both large-scale body and foot keypoints as well as the more subtle face and hand keypoints.

2.4 Hand Keypoint Detection

With the exciting improvements in face and body estimation, recent research is targeting hand keypoint detection. However, its manual annotation is extremely challenging and expensive, due to heavy self-occlusion [13]. As a result, large hand keypoint datasets in-the-wild do not exist. To alleviate this problem, early work is based on depth information [59, 60, 61, 14], but is limited to indoor scenarios. Most of the work in RGB-based hand estimation is focused on 3D estimation [62, 63, 64, 15], primarily based on fitting complex 3D models with strong priors. In 2D RGB domain, Simon *et al.* [13] exploit multi-view bootstrapping to create a hand keypoint dataset and train a 2D RGB-based hand detector. First, a naive detector is trained on a small subset of manually labeled annotations. Next, this detector is applied into a 30-camera multi-view dome structure [65, 19] to obtain new annotations based on 3D reconstruction. Unfortunately, most of the methods have only demonstrated results in controlled lab environments.

2.5 Whole-Body Keypoint Detection

OpenPose [1, 10, 13] is the only known work able to provide all body, face, hand, and foot keypoints in 2D. It operates in a multi-network fashion. First, it detects the body and foot keypoints based on [10, 46]. Then, it approximates the face and hand bounding boxes based on the body keypoints, and applies a keypoint detection network for each subsequent face and hand candidate [13]. Recent work is also targeting 3D mesh reconstruction [19, 66], usually leveraging the lack of 3D datasets with the existing 2D datasets and detectors, or reconstructing the 3D surface of the human body from denser 2D human annotations [16].

2.6 Multi-Task Learning

To overcome the problems of state-of-the-art whole-body pose estimation, we aim to apply multi-task learning (MTL) to train a single whole-body estimation model out of the four different annotation tasks (body, face, hand, and foot detection). MTL applied to deep learning can be split into soft and hard parameter sharing of hidden layers. In soft parameter sharing, each task has its own model, but the distance between the parameters is regularized to encourage them to be similar between models [67, 68]. Hard parameter sharing is the most commonly used MTL approach in computer vision, applied in many applications, such as facial alignment [28] or surface normal prediction [27]. Particularly, it has had a critical impact on object detection, where Fast R-CNN [26] exploits MTL in order to merge all the previously independent object detection tasks into a single and improved detector. It considerably improved training and testing speed as well as detection accuracy. Analogously to Fast R-CNN, our work brings together multiple and, currently, independent keypoint detection tasks into a unified framework. See [69] for a more detailed survey of multi-task learning literature.

2.7 PAF-based Body Pose Estimation

The network architecture of the initial body keypoint detector used by OpenPose could have been based on any state-of-the-art body-only keypoint detector technique. In our case, we built OpenPose upon the Part Affinity Field (PAF) network architecture, based on the work by Cao *et al.* [10]. Here, we review the main details of this method. We refer the reader to [10, 1] for a full description. This approach iteratively predicts Part Affinity Fields (PAFs), which encode part-to-part association, and detection confidence maps. Each PAF is defined as a 2D orientation vector that points from one keypoint to another. The input image I is initially analyzed by a convolutional network (pre-trained on VGG-19 [70]), generating a set of feature maps $\mathbf{F}$. Next, $\mathbf{F}$ is fed into the first stage $\phi^{(1)}$ of the network ϕ, which predicts a set of PAFs $\mathbf{L^{(1)}}$. For each subsequent stage i, the PAFs of the previous stage $\mathbf{L^{(t-1)}}$ are concatenated to $\mathbf{F}$ and refined to produce

$\mathbf{L}^{(t)}$. After N stages, we obtain the final set of PAF channels $\mathbf{L} = \mathbf{L}^{(N)}$. Then, $\mathbf{F}$ and $\mathbf{L}$ are concatenated and fed into a network ρ, which predicts the keypoint confidence maps $\mathbf{S}$.

$$\mathbf{L}^{(1)} = \phi^{(1)}\left(\mathbf{F}\right) \tag{1}$$

$$\mathbf{L}^{(t)} = \phi^{(t)}\left(\mathbf{F}, \mathbf{L}^{(i-1)}\right), \; \forall\, 2 \leq t \leq N \tag{2}$$

$$\mathbf{L} = \mathbf{L}^{(N)} \tag{3}$$

$$\mathbf{S} = \rho\left(\mathbf{F}, \mathbf{L}\right) \tag{4}$$

A L2 loss function is applied at the end of each stage, which compares the estimated predictions and the groundtruth maps ($\mathbf{S}^*$) and fields ($\mathbf{L}^*$) for each pixel (p) on each confidence map (c) and PAF (f) channel:

$$f_{\mathbf{L}} = \sum_{f=1}^{F}\sum_{p}\left(W(p) \cdot \|\mathbf{L}_f(p) - \mathbf{L}_f^*(p)\|_2^2\right) \tag{5}$$

$$f_{\mathbf{S}} = \sum_{c=1}^{C}\sum_{p}\left(W(p) \cdot \|\mathbf{S}_c(p) - \mathbf{S}_c^*(p)\|_2^2\right) \tag{6}$$

where C and F are the number of stages for confidence map and PAF prediction, and W is a binary mask with $W(p)$=0 when an annotation is missing at a pixel p. Non-maximum suppression is performed on the confidence maps to obtain a discrete set of body part candidate locations. Finally, bipartite graph matching [71] is used to assemble the connections that share the same part detection candidates into full-body poses for each person in the image.

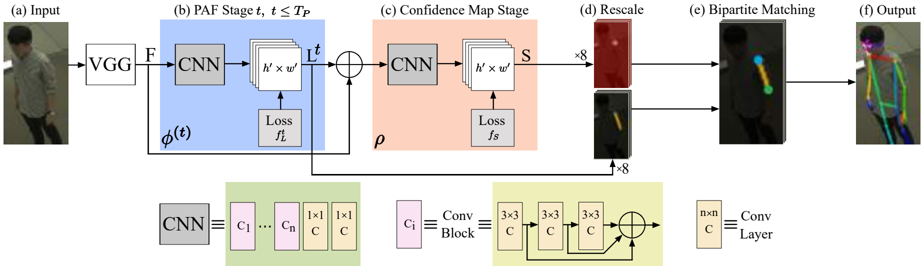


Figure 3: Overall pipeline. (a) A RGB image is taken as input. (b/c) Architecture of the whole-body pose estimation network, consisting of multiple stages predicting refined PAFs (L) and confidence maps (S) for body, face, hand and foot. It is trained end-to-end with a multi-task loss that combines the losses of each individual keypoint annotation task. Each *Conv Layer* shown corresponds to the sequence Convolution-PReLU. (d) At test time, the most refined PAF and confidence map channels are resized to increase the accuracy. (e) The parsing algorithm uses the PAFs to find all the whole-body parts belonging to the same person by bipartite matching. (f) The final whole-body poses are returned for all people in the image.

3 Method

Our system follows a streamlined approach, using a RGB image to generate a set of whole-body human keypoints for each person detected. This global pipeline is illustrated in Fig. 3. The extracted keypoints contain information from the face, torso, arms, hands, legs, and feet.

3.1 OpenPose: Multi-Network Pose Estimation

A growing number of computer vision and machine learning applications require 2D human pose estimation as an input for their systems [17, 18, 22, 19, 23, 20, 25]. To help the research community boost their work, we publicly released OpenPose [1], the first real-time multi-person system to jointly detect human body, foot, hand, and facial keypoints (in total 135 keypoints) on single images. Here, we review the details of this previous work. Available 2D body pose estimation libraries, such as Mask R-CNN [50] or Alpha-Pose[48], require their users to implement most of the pipeline, their own frame reader (e.g., video, images, or camera streaming), a display to visualize the results, output file generation with the results (e.g., JSON or XML files), etc. In addition, existing facial and body keypoint detectors are not combined, requiring a different library for each purpose. OpenPose overcome all of these problems. It can run on different platforms, including Ubuntu, Windows, Mac OSX, and embedded systems (e.g., Nvidia Tegra TX2). It also provides support for different hardware, such as CUDA GPUs, OpenCL GPUs, and CPU-only devices. The user can select an input between images, video, webcam, and IP camera streaming. He can also select whether to display the results or save them on disk, enable or disable each detector (body, foot, face, and hand), enable pixel coordinate normalization, control how many GPUs to use,

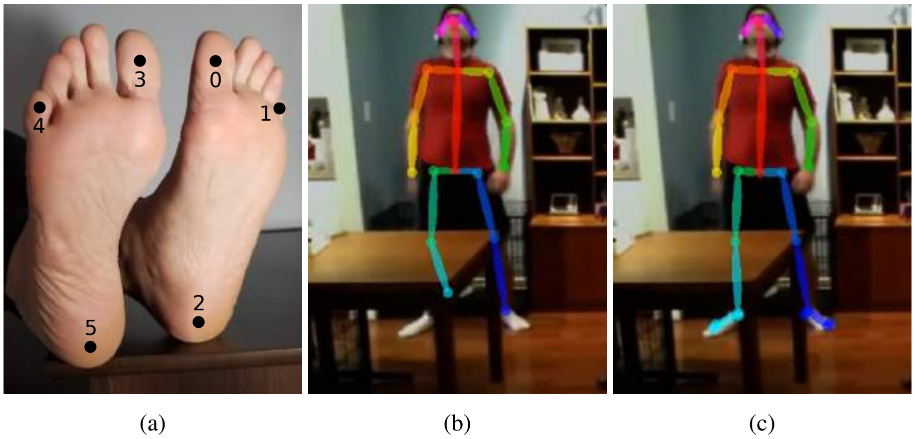
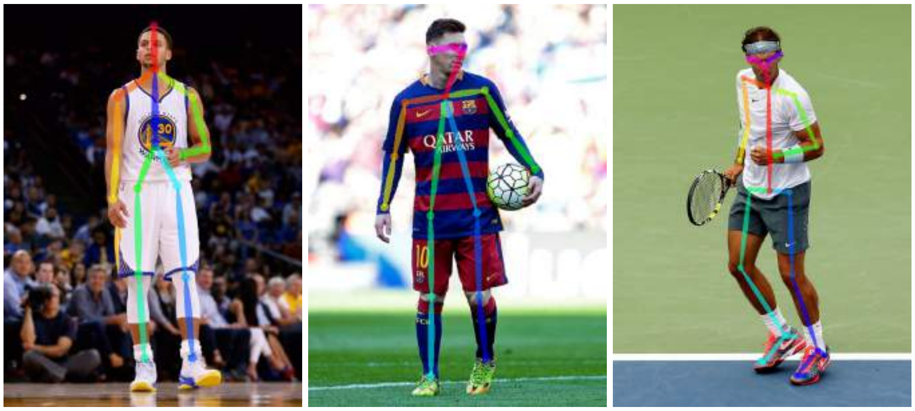


Figure 4: Foot keypoint analysis. (a) Foot keypoint annotations, consisting of big toes, small toes, and heels. (b) Body-only model example at which right ankle is not properly estimated. (c) Analogous body-foot model example, the foot information helped predict the right ankle location.

skip frames for a faster processing, etc.

OpenPose consists of three different blocks: (a) body-foot detection, (b) hand detection [13], and (c) face detection. The core block is the combined body-foot keypoint detector (detailed in Sec. 3.3). It can alternatively use the original body-only detectors [10] trained on COCO and MPII datasets. Based on the output of the body detector, facial bounding box proposals can roughly be estimated from some body part locations, in particular ears, eyes, nose, and neck. Analogously, the hand bounding box proposals are generated with the arm keypoints. The hand keypoint detector algorithm is explained in further detail in [13], while the facial keypoint detector has been trained in the same fashion as that of the hand keypoint detector. The library also includes 3D realtime single-person keypoint detection, able to predict 3D pose estimation out of multiple synchronized camera views. It performs 3D triangulation with non-linear Levenberg-Marquardt refinement [72].

The inference time of OpenPose outperforms all state-of-the-art methods, while preserving high-quality results. Its combined body-foot model is able to run at about 22 FPS in a machine with a Nvidia GTX 1080 Ti while preserving high accuracy. OpenPose has already been used by the research community for many vision and robotics topics, such as person re-identification [25], GAN-based video retargeting of human faces [17] and bodies [18], Human-Computer Interaction [22], 3D human pose estimation [20], and 3D human mesh model generation [19]. In addition, the OpenCV library [73] has included OpenPose and our PAF-based network architecture within its Deep Neural Network (DNN) module.



(a) MPII (b) COCO (c) COCO+Foot

Figure 5: Keypoint annotation configuration for the 3 datasets.

3.2 Improved Body Network Architecture

In the original work by Cao *et al.* [10], both the affinity field and confidence map branches were refined at each stage. However, in our updated approach, we only refine over PAF stages, and only predict confidence maps in 1 stage. Hence, the amount of computation per stage is reduced by half. We empirically observe that refined affinity field predictions improve the confidence map results, while the opposite does not hold. Intuitively, if we look at the PAF channel output, the body part locations can be guessed. However, if we see a bunch of body parts with no other information, we cannot parse them into different people.

In addition, the network depth is also increased. In the original approach [10], the network architecture included several 7x7 convolutional layers. In our current model, the receptive field is preserved while the computation is reduced, by replacing each 7x7 convolutional kernel by 3 consecutive 3x3 kernels. While the number of operations for the former is $2 \times 7^2 - 1 = 97$, it is only 51 for the latter. Additionally, the output of each one of the 3 convolutional kernels is concatenated, following an approach similar to DenseNet [74]. The number of non-linearity layers is tripled, and the network can keep both lower level and higher level features. Batch normalization is required to help our deeper architecture converge. However, it introduces a slow down of about 20%. Instead, we replace the ReLU layers by PReLU layers, which help convergence in a similar way to batch normalization.

3.3 Single-Network Body-Foot Pose Estimation

Existing human pose datasets ([40, 39]) contain limited body part types. The MPII dataset [40] annotates ankles, knees, hips, shoulders, elbows, wrists, necks, torsos, and head tops, while COCO [39] also includes some facial keypoints. For both of these datasets, foot annotations are limited to ankle position only. However, graphics applications such as avatar retargeting or 3D human shape reconstruction ([21, 19]) require foot keypoints such as big toe and heel. Without foot keypoint information, these approaches suffer from problems such as the candy wrapper effect, floor penetration, and foot skate. To address these issues, a small subset of about 15K human foot instances has been labeled using the Clickworker annotation platform. The dataset is obtained out of the over 100K person annotation instances available in the COCO dataset. It is split up with 14K annotations from the COCO training set and 545 from the validation set. A total of 6 foot keypoints have been labeled (see Fig. 4a). We consider the 3D coordinate of the foot keypoints rather than the surface position. For instance, for the exact toe positions, we label the area between the connection of the nail and skin, and also take depth into consideration by labeling the center of the toe rather than the surface.

Using our dataset, we train a foot keypoint detection algorithm. A näive foot keypoint detector could have been built by using a body keypoint detector to generate foot bounding box proposals, and then training a foot detector on top of it. However, this method suffers from the top-down problems discussed previously. Instead, the same architecture previously described for body estimation is trained to predict both the body and foot locations. Fig. 5 shows the keypoint distribution for the three datasets (COCO, MPII, and COCO+foot). The body-foot model also incorporates an interpolated point between the hips to allow the connection of both legs even when the upper torso is occluded or out of the image. We find evidence that foot keypoint detection implicitly helps the network to more accurately predict some body keypoints, in particular leg keypoints, such as ankle locations. Fig. 4b shows an example where the body-only network was not able to predict ankle location. By including foot keypoints during training, while maintaining the same body annotations, the algorithm can properly predict the ankle location in Fig. 4c.

3.4 Single-Network Whole-Body Pose Estimation

We want whole-body pose estimation to be accurate but also fast. Training an individual PAF-based network to predict each individual set of keypoints would achieve the first goal, but would also be computationally inefficient. Instead, we extend the body-only PAF framework to whole-body pose estimation, requiring various modifications of the training approach and network architecture.

Multi-task learning training: We modify the definition of the keypoint confidence maps $\mathbf{S}$ as the concatenation of the body (S_B), face (S_F), hand (S_H), and foot (S_O) confidence maps. Analogously, the set of PAFs at stage i, $\mathbf{L^{(i)}}$, is defined as the concatenation of the body ($L_B^{(i)}$), face ($L_F^{(i)}$), hand ($L_H^{(i)}$), and foot ($L_O^{(i)}$) PAFs. An interconnection between the different annotation tasks must be created in order to allow the different set of keypoints of the same person to be assembled together. For instance, we join the body and foot keypoints through the ankle keypoint, which is annotated in both datasets. Analogously, the wrists connect the body and hand keypoints, while the eyes relate body and face. The rest of the pipeline (non-maximum suppression over confidence maps and bipartite matching to assemble full people) is not further modified. As opposed to having a dedicated network for each keypoint annotation task, all the keypoints are now defined within the same model architecture. This is an extreme

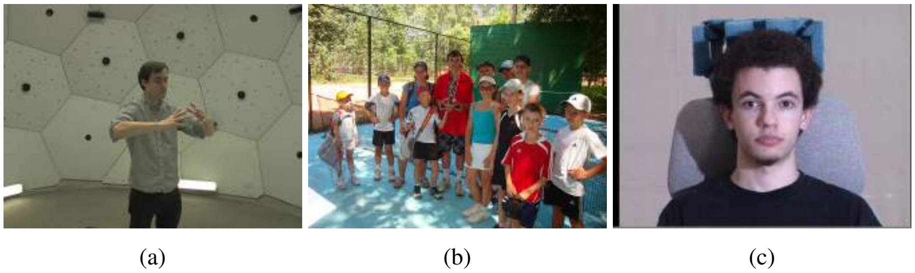


Figure 6: Different kinds of datasets for each set of keypoints present different properties (number of people, occlusion, person scale, etc.). We show typical examples from the hand (left), body (center), and face (right) datasets.

version of hard parameter sharing, in which only the final layer is task-specific.

Dataset-based batch and masking: If we had a whole-body dataset, we could train a combined model following the body-only training approach. Unfortunately, each available dataset only contains annotations for a subset of keypoints. To overcome the lack of a combined dataset, we follow the probability ratio idea of our single-network body-foot detector [1], which was trained from a body-only and body-foot datasets. Batches of images are randomly picked from each available dataset, and the losses for the confidence map and PAF channels associated to non-labeled keypoints are masked out (i.e., their binary mask W is set to 0). The probability ratio p^d is defined as the probability of picking the next annotated batch of images from the dataset d. This probability is distributed across the different datasets depending on the number of images in each dataset. When applied to keypoints with similar scale properties (e.g., body and foot [1]), it results in a robust keypoint detector. However, when applied to whole-body estimation, face and hand keypoint detection does not converge. Additionally, the accuracy of the body and foot detectors is considerably reduced. Solving the face and hand convergence problem requires a deeper understanding of the properties and differences of each set of keypoints.

Dataset-based augmentation: There is an inherent scale difference between body-foot and face-hand keypoints, which results in different kinds of datasets for each set of keypoints. Body datasets predominately contain images with multiple people and low facial and hand resolution; face datasets focus on images with a single person or cropped face; and hand datasets usually contain images with a single full-body person. Fig. 6 shows typical examples from each dataset. To solve this problem, different augmentation parameters are applied to each set of keypoints. For instance, the minimum possible scale augmentation for face datasets is enlarged to expose our model to small faces, recreating in-the-wild environments. Oppositely, the maximum scale augmentation for hand datasets is expanded so that full-sized hands appear more frequently, allowing the network to generalize to high resolution hands.

Overfitting: The face and hand detectors finally converge and we can build an initial whole-body pose detector. However, we observe a large degree of over-fitting in some validation sets, particularly in the face and lab-recorded datasets. Even though the initial probability ratio p^d is evenly distributed depending on the number of images in each dataset, the data complexity of these datasets is lower than the complexity of the challenging multi-person and in-the-wild datasets. In addition, the range of possible facial gestures is much smaller than the number of possible body and hand poses. Thus, the probability ratio of picking a batch from one of the facial and lab-recorded datasets must be additionally reduced. Empirically, we fine-tune the probability ratios between datasets so that the validation accuracy of each one converges at the same pace.

High false positive rate: Face, hand, and foot keypoints present a high false positive rate, producing a “ghosting” effect on their respective confidence map and PAF channels. Visually, this means that these channels are outputting a non-zero value in areas of the image that do not contain people. To mitigate this problem, their binary mask $W_i(p)$ is re-enabled in the COCO dataset for the parts of the image with no people. We also add an additional dataset consisting of the COCO images with no people.

Further refinement: Face and hand datasets do not necessarily annotate all the people that appear on each image. We apply Mask R-CNN [50] to mask out the regions of the image with non-labeled people. In addition, the pixel localization precision of the face and hand keypoint detectors remains low. To moderately improve it, we reduce the radius of the Gaussian distribution used to generate the groundtruth of their confidence map channels.

Shallow whole-body detector: At this point, we can build a working whole-body pose detector. The inference runtime of this refined detector matches that of running body-foot OpenPose. However, it continues to suffer from two main issues. On the one hand, the body and foot accuracy considerably decreases compared to its standalone analog (i.e., the OpenPose body-foot detector). The complexity of the network output has increased from predicting 25 to 135 keypoints (and their corresponding PAFs). The network has to compress about 5 times more information with the same number of parameters, reducing the accuracy of each individual part. On the other hand, face and hand detection accuracy appear relatively similar to that of our original multi-stage approach (OpenPose) in the benchmarks, but the qualitative results show that their pixel localization precision remains low. We reuse the same network as that used in body-only pose estimation, which presents low input resolution. Face and hand detection requires a network with higher resolution to provide results with high pixel localization precision. This initial detector is defined as “Shallow whole-body” in Sec. 4.

Improved network architecture: To match the accuracy of the body-only detector and solve the resolution issue of face and hand, the whole-body network architecture must diverge from that of our original model (OpenPose). It must still maintain a large receptive field for accurate body detection but also offer high-resolution for precise face and hand keypoint detection. Additionally, its inference runtime should remain similar to or improve upon that of its analogous multi-stage whole-body detector. Our final model architecture, refined for whole-body estimation and shown in Fig. 3, differs from the original one in the following details:

- The network input resolution is increased to considerably improve face and hand precision. Unfortunately, this implicitly reduces the effective receptive field (further reducing body accuracy).
- The number of convolutional blocks on each PAF stage is increased to recover the effective receptive field that was previously reduced.

- The width of each convolutional layer in the last PAF stage is increased to improve the overall accuracy, enabling our model to match the body accuracy of the standalone body detector.

- The previous solutions considerably increase the overall accuracy of our approach but also harm the training and testing speed. The number of PAF stages is reduced to partially overcome this issue, which only results in a moderate reduction in overall accuracy.

This improved model highly outperforms multi-stage OpenPose in speed, being approximately $p\times$ faster for an image with p people in it. Additionally, it also slightly improves its global accuracy (Sec 4.3, 4.4 and 4.5). This network is denoted as "Deep whole-body" in Sec. 4.

4 Evaluation

4.1 Experimental Setup

Datasets: We train and evaluate our method on different benchmarks for each set of keypoints: (1) COCO keypoint dataset [39] for multi-person body estimation; (2) OpenPose foot dataset [1], which is a subset of 15K annotations out of the COCO keypoint dataset; (3) OpenPose hand dataset [13], which combines a subset of 1k hand instances manually annotated from MPII [40] as well as a set of 15k samples automatically annotated on the Dome or Panoptic Studio [75]; (4) our custom face dataset, consisting of a combination of the CMU Multi-PIE Face [76], Face Recognition Grand Challenge (FRGC) [77], and i-bug [78] datasets; (5) the Monocular Total Capture dataset [21], the only available 2D whole-body dataset which has been recorded in the same Panoptic Studio used for the hand dataset. Following the standard COCO multi-person metrics, we report mean Average Precision (AP) and mean Average Recall (AR) for all sets of keypoints.

Training: All models are trained using 4-GPU servers, with a batch size of 10 images, Adam optimization, and an initial learning rate of 5e-5. We also decrease the learning rate by a factor of 2 after 200k, 300k, and every additional 60k iterations. We apply random cropping, rotation ($\pm 45^o$), flipping (50%), and scale (in the range $[1/3, 1.5]$) augmentation. The scale is modified to $[2/3, 4.5]$ and $[0.5, 4.0]$ for Dome and MPII hand datasets, respectively. The input resolution of the network is set to 480×480 pixels. Similarly to our original work [1], we maintain VGG-19 as the backbone. The probability of picking an image from each dataset is 76.5% for COCO, 5% for foot and MPII, 0.33% for each face dataset, 0.5% for Dome hand, 5% for MPII hand, 5% for whole-body data, and 2% for picking an image with no people in it.

Evaluation: We report both single-scale (image resized to a height of 480 pixels while maintaining the aspect ratio) and multi-scale results (results averaged from images resized to a height of 960, 720, 480, and 240 pixels).

4.2 Ablation Experiments

Increasing the network resolution is crucial to enable accurate hand and facial detection. Nevertheless, it directly results in slower training and testing speeds. We aim to maximize the accuracy while preserving a reasonable runtime performance. Thus, we explore multiple models tuned to maintain the same inference runtime. The final model is selected as the one maximizing the body AP. Table 1 show the results on the COCO [39] validation set. The most efficient configuration is achieved when increasing the number of convolutional blocks and their width, while reducing the number of stages in order to preserve the speed.

4.3 Body and Foot Keypoint Detection Accuracy

Once the optimal model has been selected, it is trained for whole-body estimation. Table 2 show the accuracy results on the COCO validation set for our 4 different models,

Model	AP	AR	APs	ARs
PAF (1s, 10b, 256w), CM (1s, 10b, 256w)	65.8	70.3	56.1	61.1
PAF (2s, 8b, 128-288w), CM (1s, 8b, 256w)	66.1	70.5	56.7	61.9
PAF (2s, 10b, 128-256w), CM (1s, 10b, 256w)	66.1	70.7	**57.0**	**62.0**
PAF (3s, 8b, 96-256w), CM (1s, 8b, 192w)	**66.4**	**70.9**	56.9	61.9
PAF (4s, 8b, 96-256w), CM (1s, 8b, 224w)	65.7	70.2	56.3	61.4
PAF (5s, 8b, 64-256w), CM (1s, 5b, 256w)	65.5	70.1	56.7	61.8

Table 1: Self-comparison on the body COCO validation set. All models have been tuned to have about the same inference runtime. "APs" and "ARs" refer to the single-scale results. "PAF" represents the Part Affinity Field network configuration and "CM" the confidence map configuration. "s" refers to the number of stages of refinement, "b" to the number of convolutional blocks per stage, "w" to the number of output channels (or width) of each convolutional layer. All other settings follow Sec. 4.1.

including our original work in [1]. Our deeper architecture slightly increases the accuracy of the original approach when trained for whole-body estimation. It can also be applied to body-foot estimation, achieving a 1.1% improvement over accuracy compared to that of the original OpenPose. Interestingly, adding face and hand keypoints to the same model results in a considerable decrease of the body detection accuracy of about 5% for the shallow model when compared to that of the original OpenPose. Intuitively, we are trying to fit nearly six times as many keypoints into the same network. The original model might not be deep enough to handle the additional complexity introduced for the new keypoints. However, this gap is smaller than 1% for the improved architecture (deep body-foot vs. deep whole-body). The additional depth helps the network generalize to a higher number of output keypoints.

Method	AP body	AP foot
Body-foot OpenPose (multi-scale) [1]	65.3	**77.9**
Shallow whole-body (multi-scale)	60.9	70.2
Deep body-foot (multi-scale)	**66.4**	76.8
Deep whole-body (multi-scale)	65.6	76.2

Table 2: Accuracy results on the COCO validation set. "Shallow" refers to the network architecture with the same depth and input resolution as that of our original OpenPose, while "Deep" refers to our improved architecture. "Body-foot" refers to the network that simply predicts body and foot keypoints, following the default OpenPose output, while "Whole-body" refers to our novel single-network model.

4.4 Face Keypoint Detection Accuracy

In order to evaluate the accuracy of face alignment, traditional approaches have used the Probability of Correct Keypoint (PCK) metric, which checks the probability that a predicted keypoint is within a distance threshold of its true location. However, it does not generalize to a multi-person setting. In order to evaluate our work, we reuse the mean Average Precision (AP) and Recall (AR), following the COCO multi-person metric. We train our whole-body algorithm with the same facial datasets that we used for the multi-stage OpenPose model: Multi-PIE [76], FRGC [77], and i-bug [78]. We create a custom validation set by selecting a small subset of images from each dataset. We show the results in Table 3. We can see that both our new method and original OpenPose greatly over-fit the Multi-PIE and FRGC datasets. These datasets consist of images annotated in controlled lab environments, and all faces appear frontal and with no occlusion, similar to the last image in Fig. 2. However, their accuracy is considerably lower in the in-the-wild i-bug dataset, where our approach is about 2% more accurate.

Method	AP frgc	AP MPie	AR ibug
OpenPose (single-scale) [1]	98.3	**96.3**	52.4
Shallow whole-body (single-scale)	**98.4**	90.6	50.6
Deep whole-body (single-scale)	**98.4**	93.2	**54.5**

Table 3: Accuracy results on our custom CMU Multi-PIE and FRGC validation sets. All the people in each image are not necessarily labeled on i-bug. Thus, those samples might be considered erroneous "false positives" and affect the AP results. However, AR is only affected by the annotated samples, so it is used as the main metric for i-bug.

4.5 Hand Keypoint Detection Accuracy

Analog to face evaluation, we randomly select a small subset of images from each hand dataset for validation. We denote "Hand Dome" for the subset of [13] recorded in the Panoptic Studio [75], and "Hand MPII" for the subset manually annotated from MPII [79] images. We show the results in Table 4. We can see that both our new method and original OpenPose greatly over-fit the Dome dataset, where usually only 1 person appears in each frame, similar to the first image in Fig. 6. However, the manually annotated images from MPII seem more challenging for both approaches, as it represents a truly in-the-wild dataset. In those images, we can see the clear benefits of our deeper architecture with respect to the original OpenPose and shallow models, outperforming them by about 5.5% on the Hand MPII dataset.

Method	AR Hand Dome	AR Hand MPII
OpenPose (single-scale) [1]	97.0	82.7
Shallow whole-body (single-scale)	94.6	82.4
Deep whole-body (single-scale)	**97.8**	**88.1**

Table 4: Accuracy results on our custom Hand Dome and Hand MPII validation sets. Analog to i-bug, these datasets might contain unlabeled people, so AR is used as the sole evaluation metric.

4.6 Runtime Comparison

In Fig. 7, we compare the inference runtime between the default whole-body OpenPose and our current single-network approach. Our new method is only 10% faster than original OpenPose for images with a single person. However, the inference time

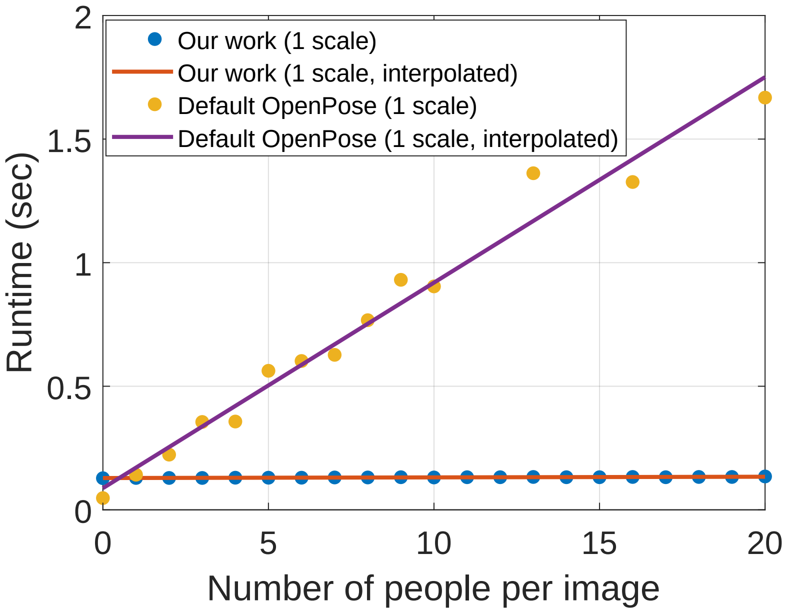


Figure 7: Inference time comparison between our original (default whole-body OpenPose) and current work (single-network approach). While the single-network inference time is invariant, multi-stage OpenPose runtime grows linearly with the number of people. The multi-stage runtime presents some oscillations because it does not run face and hand detectors if the nose or wrist keypoints (provided by the body network) of a person are not found. This is a common case in images with many people or crowded images. This analysis was performed on a system with a Nvidia 1080 Ti.

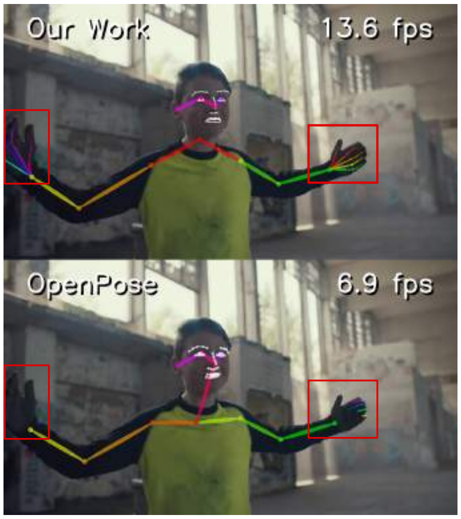
of our single-network approach remains constant, while multi-stage OpenPose's time is proportional to the number of people detected. To be more precise, it is proportional to the number of face and hand proposals. This leads to a massive speedup of our approach when the number of people increases. For images with p people, our new approach is approximately p times faster than the original OpenPose. For crowded images, many hands and faces are occluded, slightly reducing this speedup. For instance, our new approach is about 7 times faster than multi-stage OpenPose for typical images with 10 people in them.

5 Conclusion

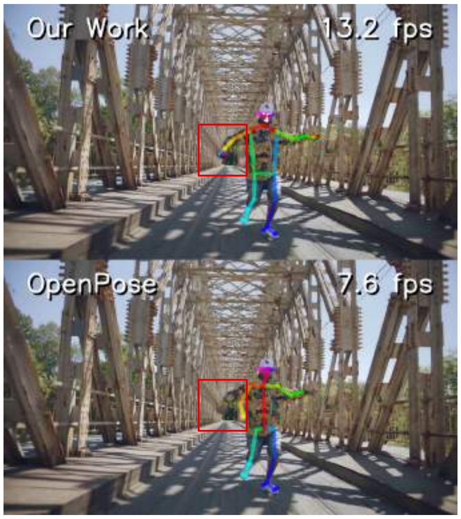
In this paper, we resort to multi-task learning, combined with an improved model architecture, to train the first single-network approach for 2D whole-body estimation. Analogous to what Fast R-CNN did for object detection, our work brings together multiple and, currently, independent keypoint detection tasks into a unified framework. We evaluate our method on multiple keypoint detection benchmarks and compare it to the state-of-the-art (our previous work, OpenPose), considerably outperforming it in both training and testing speed as well as slightly improving its accuracy. We qualitatively show in Fig. 8a that our face and hand detectors generalize better to in-the-wild images, benefiting from their indirect exposure to the immense body datasets. Nevertheless, there are still some limitations with our method. First, we observe global failure cases when a significant part of the target person is occluded or outside of the image boundaries. Secondly, the accuracy of the face and especially hand keypoint detectors is still limited, recursively failing in the case of severe motion blur, small people, and extreme gestures. Third, we qualitatively observe that our multi-stage model (original OpenPose) outperforms our new approach for face and hand detection when their poses are simple and no occlusion occurs. Original OpenPose crops the bounding box proposal of those bounding box candidates, resizes them up, and feeds them into its dedicated networks. This higher input resolution leads to an increased pixel localization precision if the keypoint detection is successful. We will publicly release the code, trained models, and validation benchmark as a baseline for future work in whole-body pose estimation.

6 Appendix: Qualitative Comparison

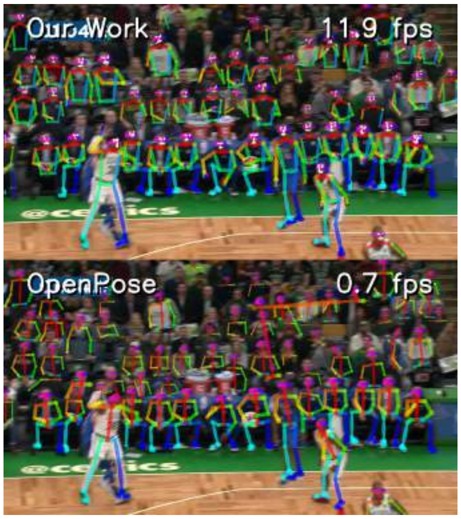
On this appendix, we provide a qualitative comparison between our method and OpenPose [1], performed on a system with 2 Nvidia 1080 Ti. Figures (a) through (j) show improved results and (k) through (o) recurrent failing cases.



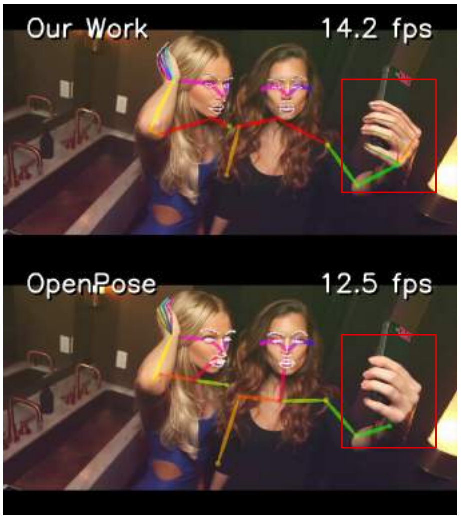
(a) The body information helps our hand detector properly predict hands when they are cropped or wearing gloves.



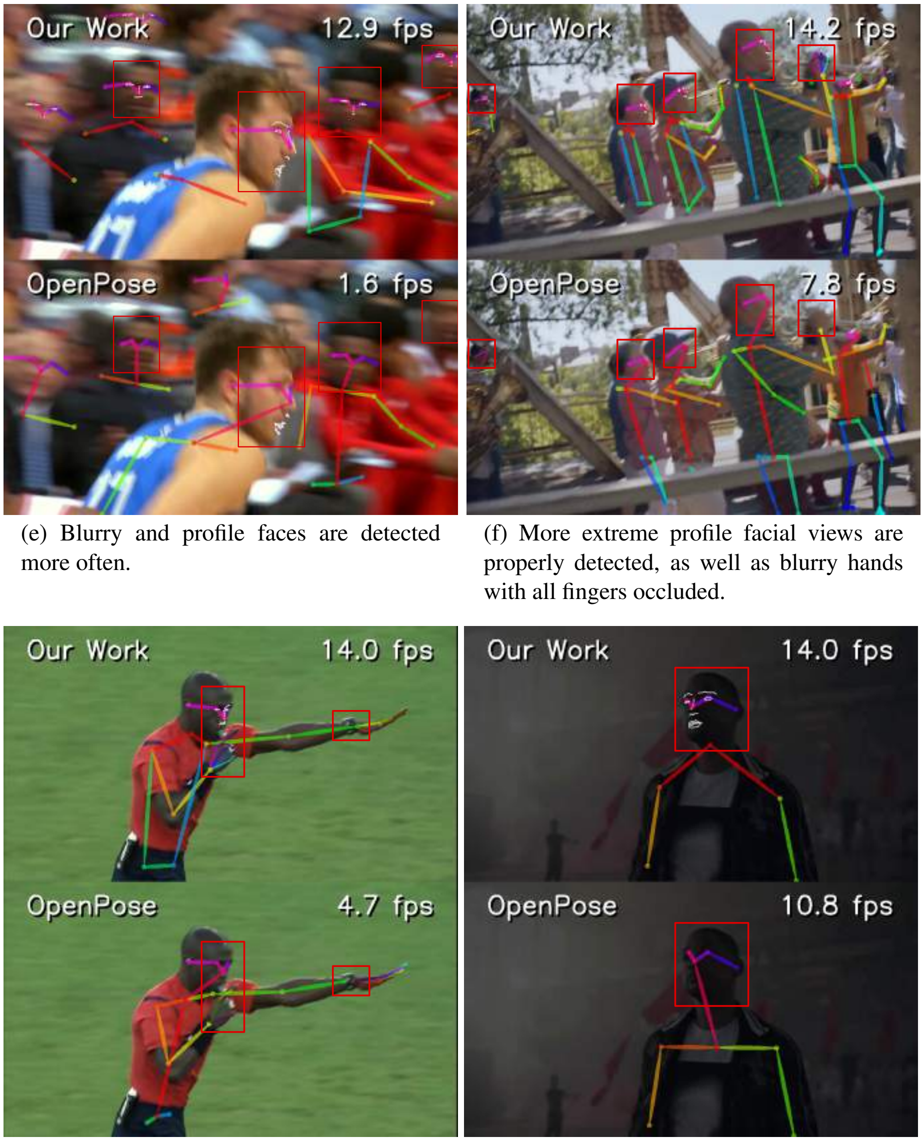
(b) The finger information implicitly helps wrist and elbow detection.



(c) Much smaller faces and hands are detected more often.



(d) More extreme hand poses are detected.

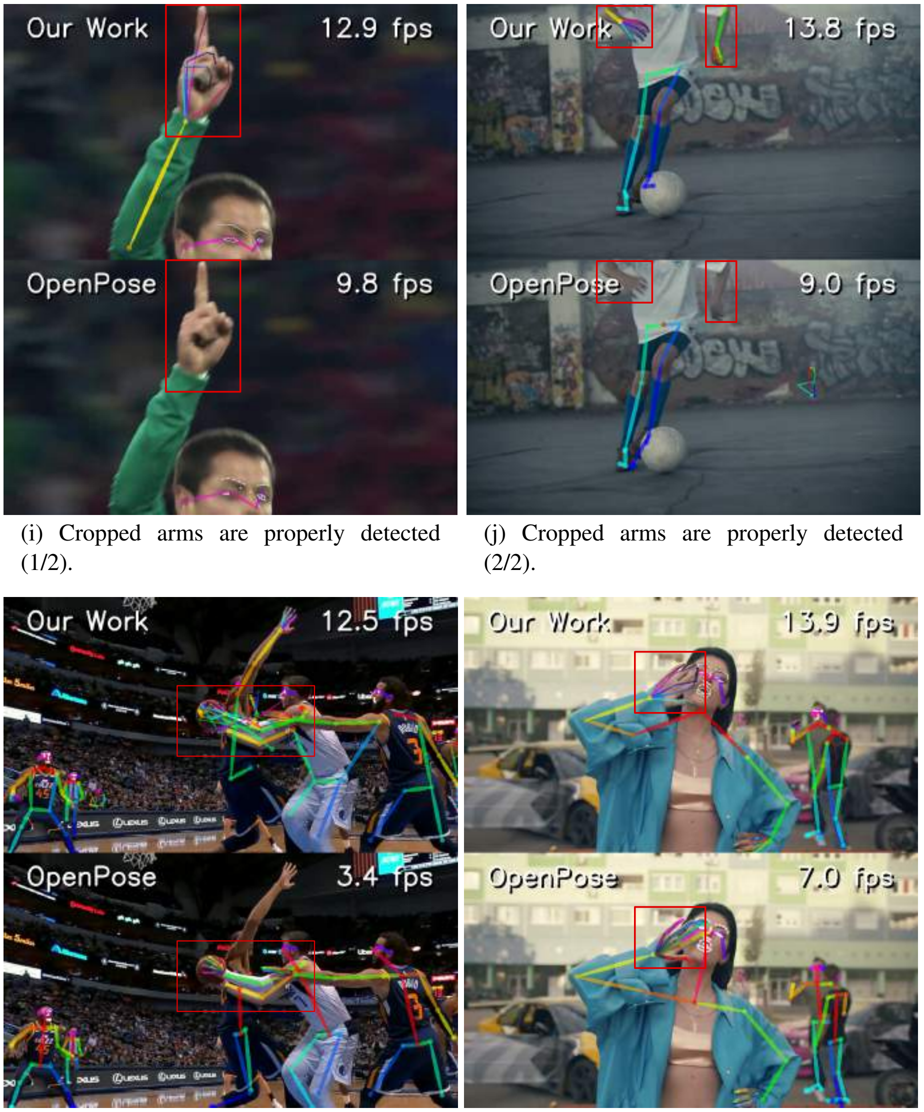


(e) Blurry and profile faces are detected more often.

(f) More extreme profile facial views are properly detected, as well as blurry hands with all fingers occluded.

(g) Hands where most fingers are occluded are detected more often. In addition, the finger information helps wrist detection when it is occluded by an object (e.g., a watch).

(h) Facial datasets mostly contain faces from Caucasian people. Indirectly exposing our face detector to the more general COCO dataset results in higher facial accuracy for people with darker skin tones and in low-brightness images.

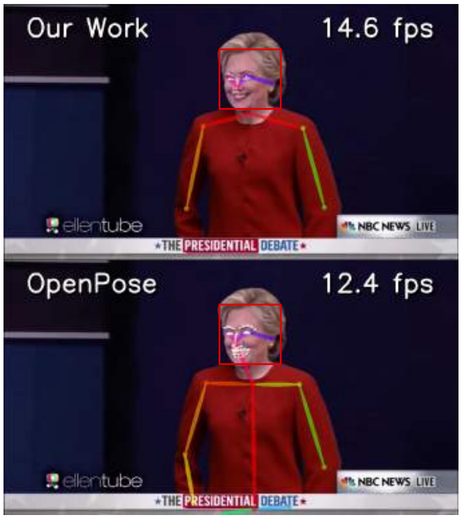


(i) Cropped arms are properly detected (1/2).

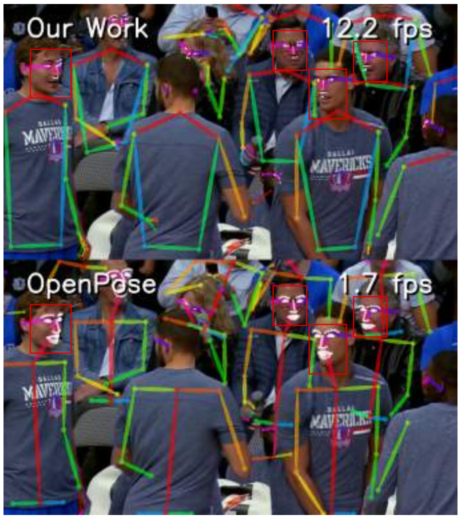
(j) Cropped arms are properly detected (2/2).

(k) Failure: Multiple hands are improperly fused together more often.

(l) Failure: Relative simple hand poses fail more often.



(m) Failure: Relative simple frontal face detection fails more often.



(n) Failure: The algorithm seems to recursively fail to detect the mouth keypoints more often.

References

[1] Z. Cao, G. Hidalgo, T. Simon, S.-E. Wei, and Y. Sheikh, “OpenPose: realtime multi-person 2D pose estimation using Part Affinity Fields,” in *arXiv preprint arXiv:1812.08008*, 2018.

[2] A. Asthana, S. Zafeiriou, S. Cheng, and M. Pantic, “Incremental face alignment in the wild,” in *CVPR*, pp. 1859–1866, 2014.

[3] R. Ranjan, V. M. Patel, and R. Chellappa, “Hyperface: A deep multi-task learning framework for face detection, landmark localization, pose estimation, and gender recognition,” *TPAMI*, vol. 41, no. 1, pp. 121–135, 2019.

[4] J. M. Saragih, S. Lucey, and J. F. Cohn, “Face alignment through subspace constrained mean-shifts,” in *ICCV*, pp. 1034–1041, IEEE, 2009.

[5] K. Zhang, Z. Zhang, Z. Li, and Y. Qiao, “Joint face detection and alignment using multitask cascaded convolutional networks,” *IEEE Signal Processing Letters*, vol. 23, no. 10, pp. 1499–1503, 2016.

[6] X. Xiong and F. De la Torre, “Supervised descent method and its applications to face alignment,” in *CVPR*, pp. 532–539, 2013.

[7] Z. Zhang, P. Luo, C. C. Loy, and X. Tang, “Learning deep representation for face alignment with auxiliary attributes,” *TPAMI*, vol. 38, no. 5, pp. 918–930, 2016.

[8] M. Andriluka, S. Roth, and B. Schiele, “Monocular 3D pose estimation and tracking by detection,” in *CVPR*, 2010.

[9] V. Belagiannis and A. Zisserman, “Recurrent human pose estimation,” in *IEEE FG*, 2017.

[10] Z. Cao, T. Simon, S.-E. Wei, and Y. Sheikh, “Realtime multi-person 2d pose estimation using part affinity fields,” in *CVPR*, 2017.

[11] Y. Chen, Z. Wang, Y. Peng, Z. Zhang, G. Yu, and J. Sun, “Cascaded pyramid network for multi-person pose estimation,” in *CVPR*, 2018.

[12] M. Dantone, J. Gall, C. Leistner, and L. Van Gool, “Human pose estimation using body parts dependent joint regressors,” in *CVPR*, 2013.

[13] T. Simon, H. Joo, I. Matthews, and Y. Sheikh, “Hand keypoint detection in single images using multiview bootstrapping,” in *CVPR*, 2017.

[14] S. Sridhar, A. Oulasvirta, and C. Theobalt, “Interactive markerless articulated hand motion tracking using rgb and depth data,” in *ICCV*, pp. 2456–2463, 2013.

[15] C. Zimmermann and T. Brox, “Learning to estimate 3d hand pose from single rgb images,” in *ICCV*, pp. 4903–4911, 2017.

[16] R. Alp Güler, N. Neverova, and I. Kokkinos, “Densepose: Dense human pose estimation in the wild,” in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 7297–7306, 2018.

[17] A. Bansal, S. Ma, D. Ramanan, and Y. Sheikh, “Recycle-gan: Unsupervised video retargeting,” in *ECCV*, 2018.

[18] C. Chan, S. Ginosar, T. Zhou, and A. A. Efros, “Everybody dance now,” in *ECCV Workshop*, 2018.

[19] H. Joo, T. Simon, and Y. Sheikh, “Total capture: A 3d deformation model for tracking faces, hands, and bodies,” in *CVPR*, 2018.

[20] P. Panteleris, I. Oikonomidis, and A. Argyros, “Using a single rgb frame for real time 3d hand pose estimation in the wild,” in *IEEE WACV*, pp. 436–445, IEEE, 2018.

[21] D. Xiang, H. Joo, and Y. Sheikh, “Monocular total capture: Posing face, body, and hands in the wild,” *arXiv preprint arXiv:1812.01598*, 2018.

[22] L. Gui, K. Zhang, Y.-X. Wang, X. Liang, J. M. Moura, and M. M. Veloso, “Teaching robots to predict human motion,” in *IROS*, 2018.

[23] 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,” in *ACM TOG*, 2017.

[24] Y. Raaj, H. Idrees, G. Hidalgo, and Y. Sheikh, “Efficient online multi-person 2d pose tracking with recurrent spatio-temporal affinity fields,” *arXiv preprint arXiv:1811.11975*, 2018.

[25] X. Qian, Y. Fu, T. Xiang, W. Wang, J. Qiu, Y. Wu, Y.-G. Jiang, and X. Xue, “Pose-normalized image generation for person re-identification,” in *ECCV*, 2018.

[26] R. Girshick, “Fast r-cnn,” in *Proceedings of the IEEE international conference on computer vision*, pp. 1440–1448, 2015.

[27] I. Misra, A. Shrivastava, A. Gupta, and M. Hebert, “Cross-stitch networks for multi-task learning,” in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 3994–4003, 2016.

[28] Z. Zhang, P. Luo, C. C. Loy, and X. Tang, “Facial landmark detection by deep multi-task learning,” in *European conference on computer vision*, pp. 94–108, Springer, 2014.

[29] V. Kazemi and J. Sullivan, “One millisecond face alignment with an ensemble of regression trees,” in *CVPR*, pp. 1867–1874, 2014.

[30] S. Zhu, C. Li, C. Change Loy, and X. Tang, “Face alignment by coarse-to-fine shape searching,” in *CVPR*, pp. 4998–5006, 2015.

[31] P. F. Felzenszwalb and D. P. Huttenlocher, “Pictorial structures for object recognition,” in *IJCV*, 2005.

[32] L. Pishchulin, M. Andriluka, P. Gehler, and B. Schiele, “Poselet conditioned pictorial structures,” in *CVPR*, 2013.

[33] D. Ramanan, D. A. Forsyth, and A. Zisserman, “Strike a Pose: Tracking people by finding stylized poses,” in *CVPR*, 2005.

[34] Y. Yang and D. Ramanan, “Articulated human detection with flexible mixtures of parts,” in *TPAMI*, 2013.

[35] L. Karlinsky and S. Ullman, “Using linking features in learning non-parametric part models,” in *ECCV*, 2012.

[36] X. Lan and D. P. Huttenlocher, “Beyond trees: Common-factor models for 2d human pose recovery,” in *ICCV*, 2005.

[37] L. Sigal and M. J. Black, “Measure locally, reason globally: Occlusion-sensitive articulated pose estimation,” in *CVPR*, 2006.

[38] Y. Wang and G. Mori, “Multiple tree models for occlusion and spatial constraints in human pose estimation,” in *ECCV*, 2008.

[39] 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 *ECCV*, 2014.

[40] M. Andriluka, L. Pishchulin, P. Gehler, and B. Schiele, “2D human pose estimation: new benchmark and state of the art analysis,” in *CVPR*, 2014.

[41] Y. Chen, C. Shen, X.-S. Wei, L. Liu, and J. Yang, “Adversarial posenet: A structure-aware convolutional network for human pose estimation,” in *ICCV*, 2017.

[42] X. Chu, W. Yang, W. Ouyang, C. Ma, A. L. Yuille, and X. Wang, “Multi-context attention for human pose estimation,” in *CVPR*, 2017.

[43] L. Ke, M.-C. Chang, H. Qi, and S. Lyu, “Multi-scale structure-aware network for human pose estimation,” in *ECCV*, 2018.

[44] A. Newell, K. Yang, and J. Deng, “Stacked hourglass networks for human pose estimation,” in *ECCV*, 2016.

[45] W. Tang, P. Yu, and Y. Wu, “Deeply learned compositional models for human pose estimation,” in *ECCV*, 2018.

[46] S.-E. Wei, V. Ramakrishna, T. Kanade, and Y. Sheikh, “Convolutional Pose Machines,” in *CVPR*, 2016.

[47] W. Yang, S. Li, W. Ouyang, H. Li, and X. Wang, “Learning feature pyramids for human pose estimation,” in *ICCV*, 2017.

[48] H.-S. Fang, S. Xie, Y.-W. Tai, and C. Lu, “RMPE: Regional multi-person pose estimation,” in *ICCV*, 2017.

[49] G. Gkioxari, B. Hariharan, R. Girshick, and J. Malik, “Using k-poselets for detecting people and localizing their keypoints,” in *CVPR*, 2014.

[50] K. He, G. Gkioxari, P. Dollár, and R. Girshick, “Mask r-cnn,” in *ICCV*, 2017.

[51] U. Iqbal and J. Gall, “Multi-person pose estimation with local joint-to-person associations,” in *ECCV Workshop*, 2016.

[52] G. Papandreou, T. Zhu, N. Kanazawa, A. Toshev, J. Tompson, C. Bregler, and K. Murphy, “Towards accurate multi-person pose estimation in the wild,” in *CVPR*, 2017.

[53] L. Pishchulin, A. Jain, M. Andriluka, T. Thormählen, and B. Schiele, “Articulated people detection and pose estimation: Reshaping the future,” in *CVPR*, 2012.

[54] B. Xiao, H. Wu, and Y. Wei, “Simple baselines for human pose estimation and tracking,” in *ECCV*, 2018.

[55] A. Newell, Z. Huang, and J. Deng, “Associative embedding: End-to-end learning for joint detection and grouping,” in *NIPS*, 2017.

[56] X. Nie, J. Feng, J. Xing, and S. Yan, “Pose partition networks for multi-person pose estimation,” in *ECCV*, 2018.

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

[58] L. Pishchulin, E. Insafutdinov, S. Tang, B. Andres, M. Andriluka, P. Gehler, and B. Schiele, “Deepcut: Joint subset partition and labeling for multi person pose estimation,” in *CVPR*, 2016.

[59] I. Oikonomidis, N. Kyriazis, and A. A. Argyros, “Tracking the articulated motion of two strongly interacting hands,” in *CVPR*, pp. 1862–1869, IEEE, 2012.

[60] T. Sharp, C. Keskin, D. Robertson, J. Taylor, J. Shotton, D. Kim, C. Rhemann, I. Leichter, A. Vinnikov, Y. Wei, *et al.*, “Accurate, robust, and flexible real-time hand tracking,” in *ACM CHI*, pp. 3633–3642, ACM, 2015.

[61] S. Sridhar, F. Mueller, A. Oulasvirta, and C. Theobalt, “Fast and robust hand tracking using detection-guided optimization,” in *CVPR*, pp. 3213–3221, 2015.

[62] Y. Cai, L. Ge, J. Cai, and J. Yuan, “Weakly-supervised 3d hand pose estimation from monocular rgb images,” in *ECCV*, pp. 666–682, 2018.

[63] U. Iqbal, P. Molchanov, T. Breuel Juergen Gall, and J. Kautz, “Hand pose estimation via latent 2.5 d heatmap regression,” in *ECCV*, pp. 118–134, 2018.

[64] F. Mueller, F. Bernard, O. Sotnychenko, D. Mehta, S. Sridhar, D. Casas, and C. Theobalt, “Ganerated hands for real-time 3d hand tracking from monocular rgb,” in *CVPR*, pp. 49–59, 2018.

[65] H. Joo, H. Liu, L. Tan, L. Gui, B. Nabbe, I. Matthews, T. Kanade, S. Nobuhara, and Y. Sheikh, “Panoptic studio: A massively multiview system for social motion capture,” in *ICCV*, 2015.

[66] A. Kanazawa, M. J. Black, D. W. Jacobs, and J. Malik, “End-to-end recovery of human shape and pose,” in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 7122–7131, 2018.

[67] L. Duong, T. Cohn, S. Bird, and P. Cook, “Low resource dependency parsing: Cross-lingual parameter sharing in a neural network parser,” in *Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics and the 7th International Joint Conference on Natural Language Processing (Volume 2: Short Papers)*, vol. 2, pp. 845–850, 2015.

[68] Y. Yang and T. M. Hospedales, “Trace norm regularised deep multi-task learning,” *arXiv preprint arXiv:1606.04038*, 2016.

[69] S. Ruder, “An overview of multi-task learning in deep neural networks,” *arXiv preprint arXiv:1706.05098*, 2017.

[70] K. Simonyan and A. Zisserman, “Very deep convolutional networks for large-scale image recognition,” *arXiv preprint arXiv:1409.1556*, 2014.

[71] D. B. West *et al.*, *Introduction to graph theory*, vol. 2. Prentice hall Upper Saddle River, NJ, 1996.

[72] D. W. Marquardt, “An algorithm for least-squares estimation of nonlinear parameters,” *Journal of the society for Industrial and Applied Mathematics*, vol. 11, no. 2, pp. 431–441, 1963.

[73] G. Bradski, “The OpenCV Library,” *Dr. Dobb’s Journal of Software Tools*, 2000.

[74] G. Huang, Z. Liu, K. Q. Weinberger, and L. van der Maaten, “Densely connected convolutional networks,” in *CVPR*, 2017.

[75] H. Joo, T. Simon, X. Li, H. Liu, L. Tan, L. Gui, S. Banerjee, T. S. Godisart, B. Nabbe, I. Matthews, T. Kanade, S. Nobuhara, and Y. Sheikh, “Panoptic studio: A massively multiview system for social interaction capture,” *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2017.

[76] R. Gross, I. Matthews, J. Cohn, T. Kanade, and S. Baker, “Multi-pie,” *Image and Vision Computing*, vol. 28, no. 5, pp. 807–813, 2010.

[77] “Face Recognition Grand Challenge (FRGC).” https://www.nist.gov/programs-projects/face-recognition-grand-challenge-frgc.

[78] C. Sagonas, E. Antonakos, G. Tzimiropoulos, S. Zafeiriou, and M. Pantic, “300 faces in-the-wild challenge: Database and results,” *Image and vision computing*, vol. 47, pp. 3–18, 2016.

[79] M. Andriluka, L. Pishchulin, P. Gehler, and B. Schiele, “2d human pose estimation: New benchmark and state of the art analysis,” in *CVPR*, June 2014.