

PifPaf: Composite Fields for Human Pose Estimation

Sven Kreiss, Lorenzo Bertoni, Alexandre Alahi
 EPFL VITA lab
 CH-1015 Lausanne
 sven.kreiss@epfl.ch

Abstract

We propose a new bottom-up method for multi-person 2D human pose estimation that is particularly well suited for urban mobility such as self-driving cars and delivery robots. The new method, PifPaf, uses a Part Intensity Field (PIF) to localize body parts and a Part Association Field (PAF) to associate body parts with each other to form full human poses. Our method outperforms previous methods at low resolution and in crowded, cluttered and occluded scenes thanks to (i) our new composite field PAF encoding fine-grained information and (ii) the choice of Laplace loss for regressions which incorporates a notion of uncertainty. Our architecture is based on a fully convolutional, single-shot, box-free design. We perform on par with the existing state-of-the-art bottom-up method on the standard COCO keypoint task and produce state-of-the-art results on a modified COCO keypoint task for the transportation domain.

1. Introduction

Tremendous progress has been made in estimating human poses “in the wild” driven by popular data collection campaigns [1, 27]. Yet, when it comes to the “transportation domain” such as for self-driving cars or social robots, we are still far from matching an acceptable level of accuracy. While a pose estimate is not the final goal, it is an effective low dimensional and interpretable representation of humans to detect critical actions early enough for autonomous navigation systems (e.g., detecting pedestrians who intend to cross the street). Consequently, the further away a human pose can be detected, the safer an autonomous system will be. This directly relates to pushing the limits on the minimum resolution needed to perceive human poses.

In this work, we tackle the well established multi-person 2D human pose estimation problem given a sin-



Figure 1: We want to estimate human 2D poses in the transportation domain where autonomous navigation systems operate in crowded scenes. Humans occupy small portion of the images and could partially occlude each other. We show the output of our PifPaf method with colored segments.

gle input image. We specifically address challenges that arise in autonomous navigation settings as illustrated in Figure 1: (i) wide viewing angle with limited resolution on humans, *i.e.*, a height of 30-90 pixels, and (ii) high density crowds where pedestrians occlude each other. Naturally, we aim for high recall and precision.

Although pose estimation has been studied before the deep learning era, a significant cornerstone is the work of OpenPose [3], followed by Mask R-CNN [18]. The former is a bottom-up approach (detecting joints without a person detector), and the latter is a top-down one (using a person detector first and outputting joints within the detected bounding boxes). While the performance of these methods is stunning on high enough resolution images, they perform poorly in the limited resolution regime, as well as in dense crowds where humans partially occlude each other.

In this paper, we propose to extend the notion of fields in pose estimation [3] to go beyond scalar and vector fields to *composite* fields. We introduce a new

neural network architecture with two head networks. For each body part or joint, one head network predicts the confidence score, the precise location and the size of this joint, which we call a Part Intensity Field (PIF) and which is similar to the fused part confidence map in [34]. The other head network predicts associations between parts, called the Part Association Field (PAF), which is of a new composite structure. Our encoding scheme has the capacity to store fine-grained information on low resolution activation maps. The precise regression to joint locations is critical, and we use a Laplace-based L_1 loss [23] instead of the vanilla L_1 loss [18]. Our experiments show that we outperform both bottom-up and established top-down methods on low resolution images while performing on par on higher resolutions. The software is open source and available online¹.

2. Related Work

Over the past years, state-of-the-art methods for pose estimation are based on Convolutional Neural Networks [18, 3, 31, 34]. They outperform traditional methods based on pictorial structures [12, 8, 9] and deformable part models [11]. The deep learning tsunami started with DeepPose [39] that uses a cascade of convolutional networks for full-body pose estimation. Then, instead of predicting absolute human joint locations, some works refine pose estimates by predicting error feedback (*i.e.*, corrections) at each iteration [4, 17] or using a human pose refinement network to exploit dependencies between input and output spaces [13]. There is now an arms race towards proposing alternative neural network architectures: from convolutional pose machines [42], stacked hourglass networks [32, 28], to recurrent networks [2], and voting schemes such as [26].

All these approaches for human pose estimation can be grouped into bottom-up and top-down methods. The former one estimates each body joint first and then groups them to form a unique pose. The latter one runs a person detector first and estimates body joints within the detected bounding boxes.

Top-down methods. Examples of top-down methods are PoseNet [35], RMPE [10], CFN [20], Mask R-CNN [18, 15] and more recently CPN [6] and MSRA [44]. These methods profit from advances in person detectors and vast amounts of labeled bounding boxes for people. The ability to leverage that data turns the requirement of a person detector into an advantage. Notably, Mask R-CNN treats keypoint detec-

tions as an instance segmentation task. During training, for every independent keypoint, the target is transformed to a binary mask containing a single foreground pixel. In general, top-down methods are effective but struggle when person bounding boxes overlap.

Bottom-up methods. Bottom-up methods include the pioneering work by Pishchulin with DeepCut [37] and Insafutdinov with DeeperCut [21]. They solve the part association with an integer linear program which results in processing times for a single image of the order of hours. Later works accelerate the prediction time [5] and broaden the applications to track animal behavior [30]. Other methods drastically reduce prediction time by using greedy decoders in combination with additional tools as in Part Affinity Fields [3], Associative Embedding [31] and PersonLab [34]. Recently, MultiPoseNet [24] develops a multi-task learning architecture combining detection, segmentation and pose estimation for people.

Other intermediate representations have been build on top of 2D pose estimates in the image plane including 3D pose estimates [29], human pose estimation in videos [36] and dense pose estimation [16] that would all profit from improved 2D pose estimates.

3. Method

The goal of our method is to estimate human poses in crowded images. We address challenges related to low-resolution and partially occluded pedestrians. Top-down methods particularly struggle when pedestrians are occluded by other pedestrians where bounding boxes clash. Previous bottom-up methods are bounding box free but still contain a coarse feature map for localization. Our method is free of any grid-based constraint on the spatial localization of the joints and has the capacity to estimate multiple poses occluding each other.

Figure 2 presents our overall model. It is a shared ResNet [19] base network with two head networks: one head network predicts a confidence, precise location and size of a joint, which we call a Part Intensity Field (PIF), and the other head network predicts associations between parts, called the Part Association Field (PAF). We refer to our method as *PifPaf*.

Before describing each head network in detail, we briefly define our field notation.

3.1. Field Notation

Fields are a useful tool to reason about structure on top of images. The notion of composite fields directly motivates our proposed Part Association Fields.

¹<https://github.com/vita-epfl/openpifpaf>

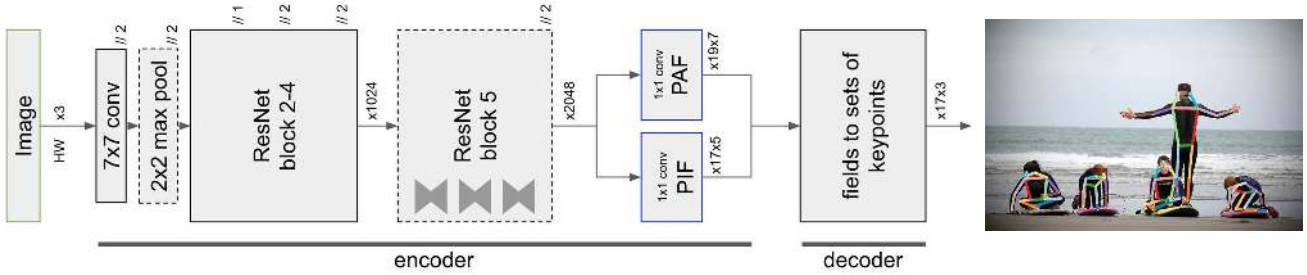


Figure 2: Model architecture. The input is an image of size (H, W) with three color channels, indicated by “x3”. The neural network based encoder produces PIF and PAF fields with 17×5 and 19×7 channels. An operation with stride two is indicated by “//2”. The decoder is a program that converts PIF and PAF fields into pose estimates containing 17 joints each. Each joint is represented by an x and y coordinate and a confidence score.

We will use i, j to enumerate spatially the output locations of the neural network and x, y for real-valued coordinates. A field is denoted with \mathbf{f}^{ij} over the domain $(i, j) \in \mathbb{Z}_+^2$ and can have as codomain (the values of the field) scalars, vectors or composites. For example, the composite of a scalar field \mathbf{s}^{ij} and a vector field \mathbf{v}^{ij} can be represented as $\{s, v_x, v_y\}$ which is equivalent to “overlying” a confidence map with a vector field.

3.2. Part Intensity Fields

The Part Intensity Fields (PIF) detect and precisely localize body parts. The fusion of a confidence map with a regression for keypoint detection was introduced in [35]. Here, we recap this technique in the language of composite fields and add a scale σ as a new component to form our PIF field.

PIF have composite structure. They are composed of a scalar component for confidence, a vector component that points to the closest body part of the particular type and another scalar component for the size of the joint. More formally, at every output location (i, j) , a PIF predicts a confidence c , a vector (x, y) with spread b (details in Section 3.4) and a scale σ and can be written as $\mathbf{p}^{ij} = \{p_c^{ij}, p_x^{ij}, p_y^{ij}, p_b^{ij}, p_\sigma^{ij}\}$.

The confidence map of a PIF is very coarse. Figure 3a shows a confidence map for the left shoulders for an example image. To improve the localization of this confidence map, we fuse it with the vectorial part of the PIF shown in Figure 3b into a high resolution confidence map. We create this high resolution part confidence map $f(x, y)$ with a convolution of an unnormalized Gaussian kernel \mathcal{N} with width p_σ over the regressed targets from the Part Intensity Field weighted by its confidence p_c :

$$f(x, y) = \sum_{ij} p_c^{ij} \mathcal{N}(x, y | p_x^{ij}, p_y^{ij}, p_\sigma^{ij}) . \quad (1)$$

This equation emphasizes the grid-free nature of the

localization. The spatial extent σ of a joint is learned as part of the field. An example is shown in Figure 3c. The resulting map of highly localized joints is used to seed the pose generation and to score the location of newly proposed joints.

3.3. Part Association Fields

Associating joints into multiple poses is challenging in crowded scenes where people partially occlude each other. Especially two step processes – top-down methods – struggle in this situation: first they detect person bounding boxes and then they attempt to find one joint-type for each bounding box. Bottom-up methods are bounding box free and therefore do not suffer from the clashing bounding box problem.

We propose bottom-up Part Association Fields (PAF) to connect joint locations together into poses. An illustration of the PAF scheme is shown in Figure 4. At every output location, PAFs predict a confidence, two vectors to the two parts this association is connecting and two widths b (details in Section 3.4) for the spatial precisions of the regressions. PAFs are represented with $\mathbf{a}^{ij} = \{a_c^{ij}, a_{x1}^{ij}, a_{y1}^{ij}, a_{b1}^{ij}, a_{x2}^{ij}, a_{y2}^{ij}, a_{b2}^{ij}\}$. Visualizations of the associations between left shoulders and left hips are shown in Figure 5.

Both endpoints are localized with regressions that do not suffer from discretizations as they occur in grid-based methods. This helps to resolve joint locations of close-by persons precisely and to resolve them into distinct annotations.

There are 19 connections for the person class in the COCO dataset each connecting two types of joints; *e.g.*, there is a right-knee-to-right-ankle association. The algorithm to construct the PAF components at a particular feature map location consists of two steps. First, find the closest joint of either of the two types which determines one of the vector components. Sec-

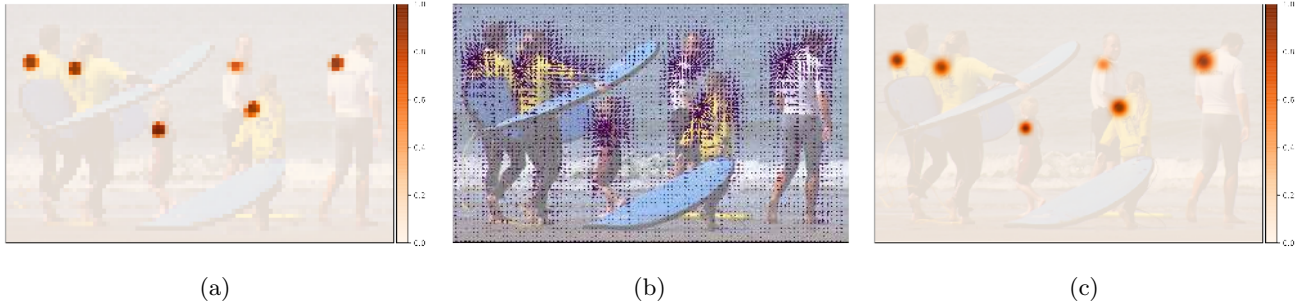


Figure 3: Visualizing the components of the PIF for the left shoulder. This is one of the 17 composite PIF. The confidence map is shown in (a) and the vector field is shown in (b). The fused confidence, vector and scale components are shown in (c).

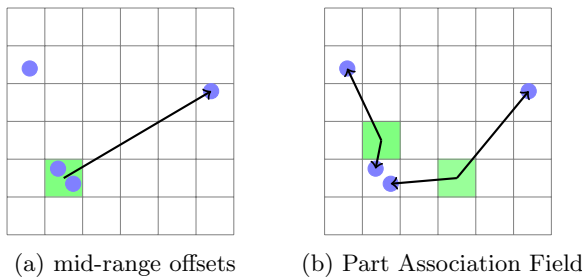


Figure 4: Illustrating the difference between PersonLab’s mid-range offsets (a) and Part Association Fields (b) on a feature map grid. Blue circles represent joints and confidences are marked in green. Mid-range offsets (a) have their origins at the center of feature map cells. Part Association Fields (b) have floating point precision of their origins.

ond, the ground truth pose determines the other vector component to represent the association. The second joint is not necessarily the closest one and can be far away.

During training, the components of the field have to point to the parts that should be associated. Similar to how an x component of a vector field always has to point to the same target as the y component, the components of the PAF field have to point to the same association of parts.

3.4. Adaptive Regression Loss

Human pose estimation algorithms tend to struggle with the diversity of scales that a human pose can have in an image. While a localization error for the joint of a large person can be minor, that same absolute error might be a major mistake for a small person. We use an L_1 -type loss to train regressive outputs. We improve the localization ability of the network by injecting a scale dependence into that regression loss with

the SmoothL1 [14] or Laplace loss [23].

The SmoothL1 loss allows to tune the radius r^{smooth} around the origin where it produces softer gradients. For a person instance bounding box area of A_i and keypoint size of σ_k , $r_{i,k}^{\text{smooth}}$ can be set proportionally to $\sqrt{A_i}\sigma_k$ which we study in Table 3.

The Laplace loss is another L_1 -type loss that is attenuated via the predicted spread b :

$$L = |x - \mu|/b + \log(2b) \quad (2)$$

It is independent of any estimates of A_i and σ_k and we use it for all vectorial components.

3.5. Greedy Decoding

Decoding is the process of converting the output feature maps of a neural network into sets of 17 coordinates that make human pose estimates. Our process is similar to the fast greedy decoding used in [34].

A new pose is seeded by PIF vectors with the highest values in the high resolution confidence map $f(x, y)$ defined in equation 1. Starting from a seed, connections to other joints are added with the help of PAF fields. The algorithm is fast and greedy. Once a connection to a new joint has been made, this decision is final.

Multiple PAF associations can form connections between the current and the next joint. Given the location of a starting joint \vec{x} , the scores s of PAF associations \mathbf{a} are calculated with

$$s(\mathbf{a}, \vec{x}) = a_c \exp\left(-\frac{\|\vec{x} - \vec{a}_1\|_2}{b_1}\right) f_2(a_{x2}, a_{y2}) \quad (3)$$

which takes into account the confidence in this connection a_c , the distance to the first vector’s location calibrated with the two-tailed Laplace distribution probability and the high resolution part confidence at the second vector’s target location f_2 . To confirm the proposed position of the new joint, we run reverse match-

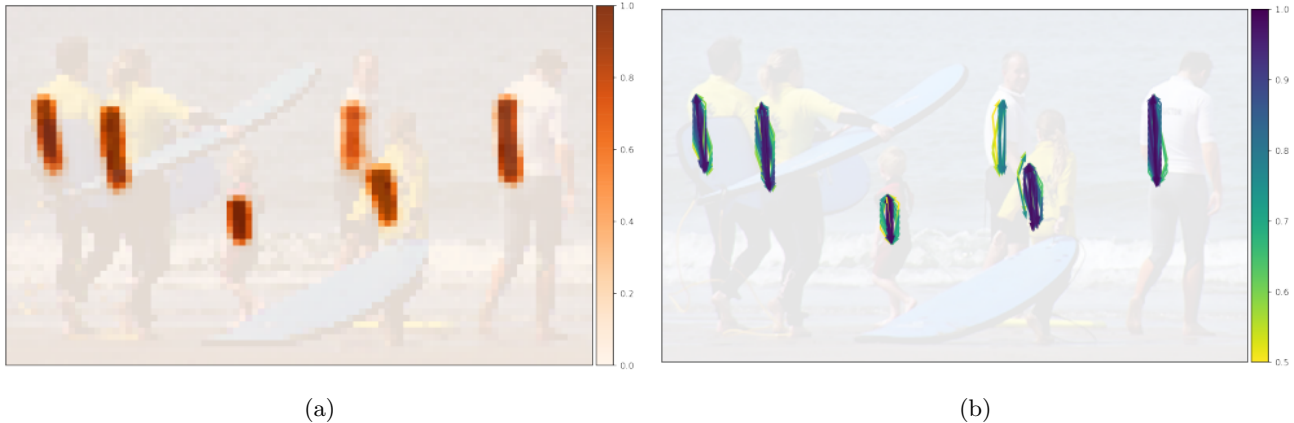


Figure 5: Visualizing the components of the PAF that associates left shoulder with left hip. This is one of the 19 PAF. Every location of the feature map is the origin of two vectors which point to the shoulders and hips to associate. The confidence of associations \mathbf{a}_c is shown at their origin in (a) and the vector components for $\mathbf{a}_c > 0.5$ are shown in (b).

ing. This process is repeated until a full pose is obtained. We apply non-maximum suppression at the keypoint level as in [34]. The suppression radius is dynamic and based on the predicted scale component of the PIF field. We do not refine any fields neither during training nor test time.

4. Experiments

Cameras in self-driving cars have a wide field of view and have to resolve small instances of pedestrians within that field of view. We want to emulate that small pixel-height distribution of pedestrians with a publicly available dataset and evaluation protocol for human pose estimation.

In addition, and to demonstrate the broad applicability of our method, we also investigate pose estimation in the context of the person re-identification task (Re-Id) – that is, given an image of a person, identify that person in other images. Some prior work has used part-based or region-based models [45, 7, 43] that would profit from quality pose estimates.

Datasets. We quantitatively evaluate our proposed method, PifPaf, on the COCO keypoint task [27] for people in low resolution images. Starting from the original COCO dataset, we constrain the maximum image side length to 321 pixels to emulate a crop of a 4k camera. We obtain person bounding boxes that are 66 ± 65 px high. The COCO metrics contain a breakdown for medium-sized humans under AP^M and AR^M that have bounding box area in the original image between between $(32 \text{ px})^2$ and $(96 \text{ px})^2$. After resizing for

low resolution, this corresponds to bounding boxes of height 44 ± 19 px.

We qualitatively study the performance of our method on images captured by self-driving cars as well as random crowded scenarios. We use the recently released *nuScenes* dataset [33]. Since labels and evaluation protocols are not yet available we qualitatively study the results.

In the context of Re-Id, we investigate the popular and publicly available Market-1501 dataset [46]. It consists of 64×128 pixel crops of pedestrians. We apply the same model that we trained on COCO data. Figure 8 qualitatively compares extracted poses from Mask R-CNN [18] with our proposed method. The comparison shows a clear improvement of the poses extracted with our PifPaf method.

Performance on higher resolution images is not the focus of this paper, however other methods are optimized for full resolution COCO images and therefore we also show our results and comparisons for high resolution COCO poses.

Evaluation. The COCO keypoint detection task is evaluated like an object detection task, with the core metrics being variants of average precision (AP) and average recall (AR) thresholded at an object keypoint similarity (OKS) [27]. COCO assumes a fixed ratio of keypoint size to bounding box area per keypoint type to define OKS. For each image, pose estimators have to provide the 17 keypoint locations per pose and a score for each pose. Only the top 20 scoring poses are considered for evaluation.

	AP	AP ^{0.50}	AP ^{0.75}	AP ^M	AP ^L	AR	AR ^{0.50}	AR ^{0.75}	AR ^M	AR ^L
Mask R-CNN* [18]	41.6	68.1	42.5	28.2	59.8	49.0	76.0	50.0	35.6	67.5
OpenPose [3]	37.6	62.5	37.2	25.0	55.3	43.9	65.3	44.9	26.7	67.5
PifPaf (ours)	50.0	73.5	52.9	35.9	69.7	55.0	76.0	57.9	39.4	76.4

Table 1: Applying pose estimation to low resolution images with the long side equal to 321 px for top-down (top part) and bottom-up (bottom part) methods. For the Mask R-CNN and OpenPose reference values, we ran the implementations by [40, 41] modified to enforce the maximum image side length. *Mask R-CNN was retrained for low resolution. The PifPaf result is based on a ResNet50 backbone.

Implementation details. All our models are based on Imagenet pretrained base networks followed by custom, multiple head sub-networks. Specifically, we use the 64115 images in the 2017 COCO training set that have a person annotation for training. Our validation is done on the 2017 COCO validation set of 5000 images. The base networks are modified ResNet50/101/152 networks. The head networks are single-layer 1x1 sub-pixel convolutions [38] that double the spatial resolution. The confidence component of a field is normalized with a sigmoid non-linearity.

The base network has various modification options. The strides of the input convolution and the input max-pooling operation can be changed. It is also possible to remove the max-pooling operation in the input block and the entire last block. The default modification used here is to remove the max-pool layer from the input block.

We apply only few and weak data augmentations. To create uniform batches, we crop images to squares where the side of the square is between 95% and 100% of the short edge of the image and the location is chosen randomly. These are large crops to keep as much of the training data as possible. Half of the time the entire image is used un-cropped and bars are added to make it square. The subsequent resizing uses bicubic interpolation. Training images and annotations are randomly horizontally flipped.

The components of the fields that form confidence maps are trained with independent binary cross entropy losses. We use L_1 losses for the scale components of the PIF fields and use Laplace losses for all vectorial components.

During training, we fix the running statistics of the Batch Normalization operations [22] to their pretrained values [34]. We use the SGD optimizer with a learning rate of 10^{-3} , momentum of 0.95, batch size of 8 and no weight decay. We employ model averaging to extract stable models for validation. At each optimization step, we update an exponentially weighted version of the model parameters. Our decay constant is 10^{-3} . The training time for 75 epochs of ResNet101 on two

GTX1080Ti is approximately 95 hours.

Baselines. We compare our proposed PifPaf method against the reproducible state-of-the-art bottom-up OpenPose [3] and top-down Mask R-CNN [18] methods. While our goal is to outperform bottom-up approaches, we still report results of a top-down approach to evaluate the strength of our method. Since this is an emulation of small humans within a much larger image, we modified existing methods to prevent upscaling of small images.

Results. Table 1 presents our quantitative results on the COCO dataset. We outperform the bottom-up OpenPose and even the top-down Mask R-CNN approach on all metrics. These numbers are overall lower than their higher resolution counterparts. The two conceptually very different baseline methods show similar performance while our method is clearly ahead by over 18% in AP.

Our quantitative results emulate the person distribution in urban street scenes using a public, annotated dataset. Figure 6 shows qualitative results of the kind of street scenes we want to address. Not only do we have less false positives, we detect pedestrians who partially occlude each other. It is interesting to see that a critical gesture such as “waving” towards a car is only detected with our method. Both Mask-RCNN and OpenPose have not accurately estimated the arm gesture in the first row of Figure 6. Such level of difference can be fundamental in developing safe self-driving cars.

We further show qualitative results on more crowded images in Figure 7. For perspectives like the one in the second row, we observe that bounding boxes of close-by pedestrians occlude further away pedestrians. This is a difficult scenario for top-down methods. Bottom-up methods perform here better which we can also observe for our PifPaf method.

To quantify the performance on the Market-1501 dataset, we created a simplified accuracy metric. The accuracy is 43% for Mask R-CNN and 96% for PifPaf. The evaluation is based on the number of images with



Figure 6: Illustration of our PifPaf method (right hand-side) against OpenPose [3] (first column) and Mask R-CNN [40] (second column) on the *nuScenes* dataset. We highlight with bounding boxes all humans that other methods did not detect, and with circles all false positives. Note that our method correctly estimates the waving pose (first row, first bounding box) of a person whereas the others fail to do so.

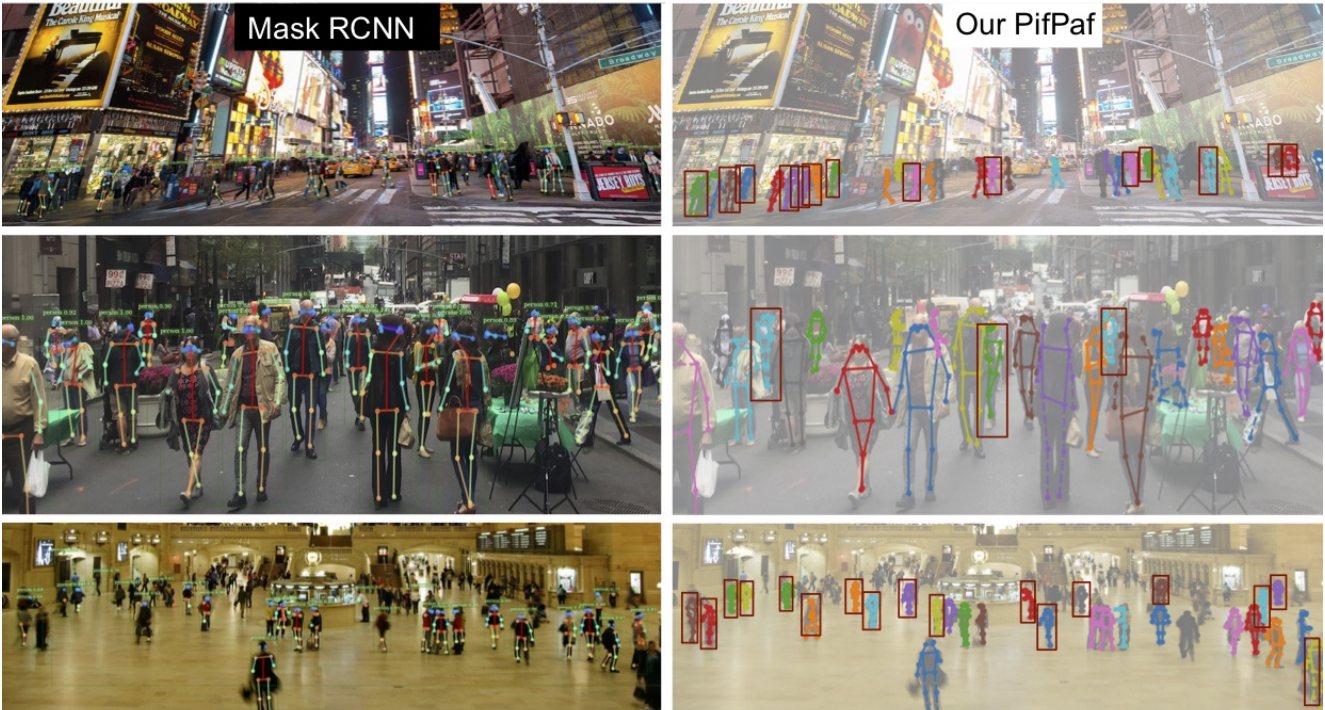


Figure 7: Illustration of our PifPaf method (right hand-side) against Mask R-CNN [40] (left hand-side). We highlight with bounding boxes all humans where Mask R-CNN misses their poses with respect to our method. Our method estimates all poses that Mask R-CNN estimates as well as the ones highlighted with bounding boxes.

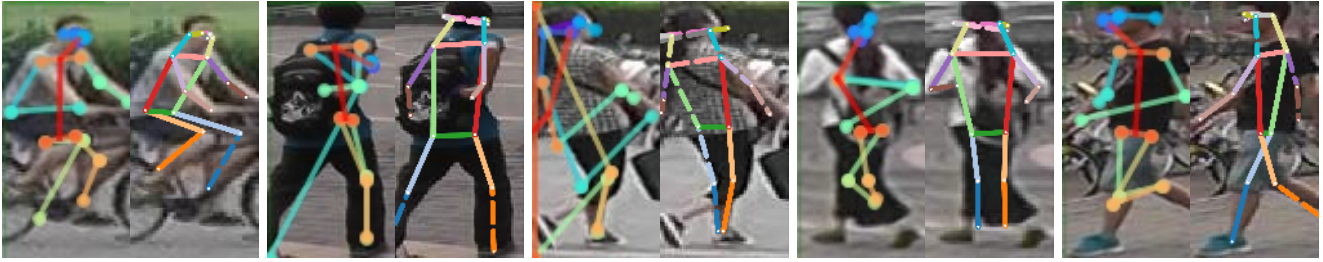


Figure 8: A selection of images from the Market-1501 [46] dataset. The left image is the output from Mask R-CNN. To improve the Mask R-CNN result, we forced it to predict exactly one pose in a bounding box that spans the entire image. The right image is the output of our PifPaf method that was not constrained to one person and could have chosen to output none or multiple poses, which is a harder task.

	AP	AP ^M	AP ^L
Mask R-CNN [18]	63.1	58.0	70.4
OpenPose [3]	61.8	57.1	68.2
PersonLab [34] – single-scale	66.5	62.4	72.3
PifPaf – single-scale (ours)	66.7	62.4	72.9

Table 2: Metrics in percent evaluated on the COCO 2017 test-dev set at optimal resolutions for top-down (top part) and bottom-up (bottom part) methods.

	AP	AP ^M	AP ^L
vanilla L_1	41.7	26.5	62.5
SmoothL1, $r = 0.2\sqrt{A_i}\sigma_k$	42.0	26.9	62.6
SmoothL1, $r = 0.5\sqrt{A_i}\sigma_k$	41.9	27.0	62.5
SmoothL1, $r = 1.0\sqrt{A_i}\sigma_k$	41.6	26.5	62.3
Laplace	45.1	31.4	64.0
Laplace (using b in decoder)	45.5	31.4	64.9

Table 3: Study of the dependence on the type of L_1 loss. Metrics are reported in percent. All models have a ResNet50 backbone and were trained for 20 epochs.

a correct pose out of 202 random images from the train set. A correct pose has up to three joints misplaced.

Other methods are optimized for higher resolution images. For a fair comparison, we show a quantitative comparison on the high resolution COCO 2017 test-dev set in Table 2. We perform on par with the best existing bottom-up method.

Ablation Studies. We studied the effects of various design decisions that are summarized in Table 3. We found that we can tune the performance towards smaller or larger objects by modifying the overall scale of r^{smooth} and so we studied its impact. However, the real improvement is obtained with the Laplace-based loss. The added scale component σ to

	AP [%]	t [ms]	t^{dec} [ms]
ResNet50	62.6	222	178
ResNet101	65.7 (60.0)	240 (355)	175
ResNet152	67.4	263	173

Table 4: Interplay between precision and single-image prediction time t on a GTX1080Ti with different ResNet backbones for the COCO val set. Last column is the decoding time t^{dec} . PersonLab [34] timing numbers (which include decoding instance masks) are given in parenthesis where available at image width 801px.

the PIF field improved AP of our ResNet101 model from 64.5% to 65.7%.

Runtime. Metrics for varying ResNet backbones are in Table 4. For the same backbone, we outperform PersonLab by 9.5% in AP with a simultaneous 32% speed up.

5. Conclusions

We have developed a new bottom-up method for multi-person 2D human pose estimation that addresses failure modes that are particularly prevalent in the transportation domain, *i.e.*, in self-driving cars and social robots. We demonstrated that our method outperforms previous state-of-the-art methods in the low resolution regime and performs on par at high resolution.

The proposed PAF fields can be applied to other tasks as well. Within the image domain, predicting structured image concepts [25] is an exciting next step.

Acknowledgements. We would like to thank EPFL SCITAS for their support with compute infrastructure.

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. 1
- [2] V. Belagiannis and A. Zisserman. Recurrent human pose estimation. In *Automatic Face & Gesture Recognition (FG 2017), 2017 12th IEEE International Conference on*, pages 468–475. IEEE, 2017. 2
- [3] Z. Cao, T. Simon, S.-E. Wei, and Y. Sheikh. Real-time multi-person 2d pose estimation using part affinity fields. In *CVPR*, volume 1, page 7, 2017. 1, 2, 6, 7, 8
- [4] J. Carreira, P. Agrawal, K. Fragkiadaki, and J. Malik. Human pose estimation with iterative error feedback. In *CVPR*, 2016. 2
- [5] L.-C. Chen, G. Papandreou, I. Kokkinos, K. Murphy, and A. L. Yuille. Deeplab: Semantic image segmentation with deep convolutional nets, atrous convolution, and fully connected crfs. *IEEE transactions on pattern analysis and machine intelligence*, 40(4):834–848, 2018. 2
- [6] Y. Chen, Z. Wang, Y. Peng, Z. Zhang, G. Yu, and J. Sun. Cascaded pyramid network for multi-person pose estimation. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 7103–7112, 2018. 2
- [7] D. Cheng, Y. Gong, S. Zhou, J. Wang, and N. Zheng. Person re-identification by multi-channel parts-based cnn with improved triplet loss function. In *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2016. 5
- [8] M. Dantone, J. Gall, C. Leistner, and L. Gool. Human pose estimation using body parts dependent joint regressors. In *CVPR*, 2013. 2
- [9] M. Eichner, M. Marin-Jimenez, A. Zisserman, and V. Ferrari. 2d articulated human pose estimation and retrieval in (almost) unconstrained still images. In *IJCV*, 2012. 2
- [10] H. Fang, S. Xie, and C. Lu. Rmpe: Regional multi-person pose estimation. *2017 IEEE International Conference on Computer Vision (ICCV)*, pages 2353–2362, 2017. 2
- [11] P. F. Felzenszwalb, R. B. Girshick, D. McAllester, and D. Ramanan. Object detection with discriminatively trained part-based models. In *PAMI*, 2010. 2
- [12] P. F. Felzenszwalb and D. P. Huttenlocher. Pictorial structures for object recognition. In *IJCV*. Springer, 2005. 2
- [13] M. Fieraru, A. Khoreva, L. Pishchulin, and B. Schiele. Learning to refine human pose estimation. *CoRR*, abs/1804.07909, 2018. 2
- [14] R. Girshick. Fast r-cnn. In *Proceedings of the IEEE international conference on computer vision*, pages 1440–1448, 2015. 4
- [15] R. Girshick, I. Radosavovic, G. Gkioxari, P. Dollár, and K. He. Detectron. <https://github.com/facebookresearch/detectron>, 2018. 2
- [16] R. A. Güler, N. Neverova, and I. Kokkinos. Densepose: Dense human pose estimation in the wild. *arXiv preprint arXiv:1802.00434*, 2018. 2
- [17] A. Haque, B. Peng, Z. Luo, A. Alahi, S. Yeung, and L. Fei-Fei. Towards viewpoint invariant 3d human pose estimation. In *European Conference on Computer Vision*, pages 160–177. Springer, 2016. 2
- [18] 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. 1, 2, 5, 6, 8
- [19] K. He, X. Zhang, S. Ren, and J. Sun. Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 770–778, 2016. 2
- [20] S. Huang, M. Gong, and D. Tao. A coarse-fine network for keypoint localization. *2017 IEEE International Conference on Computer Vision (ICCV)*, pages 3047–3056, 2017. 2
- [21] E. Insafutdinov, L. Pishchulin, B. Andres, M. Andriluka, and B. Schiele. Deeppercut: A deeper, stronger, and faster multi-person pose estimation model. In *European Conference on Computer Vision*, pages 34–50. Springer, 2016. 2
- [22] S. Ioffe and C. Szegedy. Batch normalization: Accelerating deep network training by reducing internal covariate shift. *arXiv preprint arXiv:1502.03167*, 2015. 6
- [23] A. Kendall and Y. Gal. What uncertainties do we need in bayesian deep learning for computer vision? In *Advances in neural information processing systems*, pages 5574–5584, 2017. 2, 4
- [24] M. Kocabas, S. Karagoz, and E. Akbas. Multiposenet: Fast multi-person pose estimation using pose residual network. *CoRR*, abs/1807.04067, 2018. 2
- [25] R. Krishna, Y. Zhu, O. Groth, J. Johnson, K. Hata, J. Kravitz, S. Chen, Y. Kalantidis, L.-J. Li, D. A. Shamma, M. Bernstein, and L. Fei-Fei. Visual genome: Connecting language and vision using crowdsourced dense image annotations. 2016. 8
- [26] I. Lifshitz, E. Fetaya, and S. Ullman. Human pose estimation using deep consensus voting. In *European Conference on Computer Vision*, pages 246–260. Springer, 2016. 2
- [27] 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. 1, 5
- [28] Y. Luo, Z. Xu, P. Liu, Y. Du, and J.-M. Guo. Multi-person pose estimation via multi-layer fractal network and joints kinship pattern. *IEEE Transactions on Image Processing*, 28:142–155, 2019. 2

- [29] J. Martinez, R. Hossain, J. Romero, and J. J. Little. A simple yet effective baseline for 3d human pose estimation. In *International Conference on Computer Vision*, volume 1, page 5, 2017. 2
- [30] A. Mathis, P. Mamidanna, K. M. Cury, T. Abe, V. N. Murthy, M. W. Mathis, and M. Bethge. Deepplabcut: markerless pose estimation of user-defined body parts with deep learning. Technical report, Nature Publishing Group, 2018. 2
- [31] 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 2277–2287, 2017. 2
- [32] 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. 2
- [33] NuTonomy. NuScenes data set. <https://www.nuscenes.org/>, 2018. 5
- [34] G. Papandreou, T. Zhu, L. Chen, S. Gidaris, J. Tompson, and K. Murphy. Personlab: Person pose estimation and instance segmentation with a bottom-up, part-based, geometric embedding model. *CoRR*, abs/1803.08225, 2018. 2, 4, 5, 6, 8
- [35] 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*, volume 3, page 6, 2017. 2, 3
- [36] T. Pfister, J. Charles, and A. Zisserman. Flowing convnets for human pose estimation in videos. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 1913–1921, 2015. 2
- [37] 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. 2
- [38] W. Shi, J. Caballero, F. Huszár, J. Totz, A. P. Aitken, R. Bishop, D. Rueckert, and Z. Wang. Real-time single image and video super-resolution using an efficient sub-pixel convolutional neural network. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 1874–1883, 2016. 6
- [39] A. Toshev and C. Szegedy. Deeppose: Human pose estimation via deep neural networks. In *CVPR*, 2014. 2
- [40] R. Tseng. Detectron.pytorch. <https://github.com/roytseng-tw/Detectron.pytorch>, 2018. 6, 7
- [41] H. Wang, W. P. An, X. Wang, L. Fang, and J. Yuan. Magnify-net for multi-person 2d pose estimation. In *2018 IEEE International Conference on Multimedia and Expo (ICME)*, pages 1–6, July 2018. 6
- [42] S.-E. Wei, V. Ramakrishna, T. Kanade, and Y. Sheikh. Convolutional pose machines. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 4724–4732, 2016. 2
- [43] Z. Wu, Y. Li, and R. J. Radke. Viewpoint invariant human re-identification in camera networks using pose priors and subject-discriminative features. *IEEE transactions on pattern analysis and machine intelligence*, 37(5):1095–1108, 2015. 5
- [44] B. Xiao, H. Wu, and Y. Wei. Simple baselines for human pose estimation and tracking. *arXiv preprint arXiv:1804.06208*, 2018. 2
- [45] H. Zhao, M. Tian, S. Sun, J. Shao, J. Yan, S. Yi, X. Wang, and X. Tang. Spindle net: Person re-identification with human body region guided feature decomposition and fusion. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 1077–1085, 2017. 5
- [46] L. Zheng, L. Shen, L. Tian, S. Wang, J. Wang, and Q. Tian. Scalable person re-identification: A benchmark. In *Computer Vision, IEEE International Conference on*, 2015. 5, 8