SmartPartNet: Part-Informed Person Detection for Body-Worn Smartphones

Heng Yu
Tsinghua University
h-yu14@mails.tsinghua.edu.cn

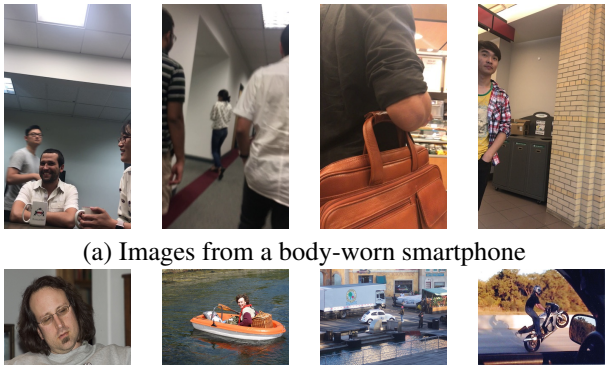
Eshed Ohn-Bar Donghyun Yoo Kris M. Kitani
Carnegie Mellon University
{eohnbar,donghyuy,kkitani}@cs.cmu.edu

Abstract

We are interested in the development of image-based person detection algorithms for wearable computing using commodity smartphones. We focus on the use of smartphones as a wearable device because it is a practical means of augmenting human sensing for applications such as navigation for the blind or assisting social interaction. We identify two unique features of developing a vision-based person detector for body-worn smartphones: (1) the detector must take into account the strong bias in the size of people in the images taken with a wearable device and (2) the detector must consider the low image quality due to dim lighting and rapid ego-motion which leads to motion blur. In order to account for the unique distribution over the visibility of body parts when using a wearable camera, we propose a part-based person detector specialized for chest-mounted smartphones. We perform extensive ablative analysis on the usefulness of part information, providing several insights regarding the design of the optimal person detector across different application domains. To account for the frequent occurrence of motion blur in our target domain, we introduce a data augmentation technique to generate synthetic motion-blurred images during training. In addition to addressing the aforementioned features, the final detector must also run in real-time using only smartphone resources. We leverage recent progress in deep neural networks for mobile devices and show that our proposed person detector, ***SmartPartNet****, obtains performance similar to state-of-the-art pedestrian detection networks, while being* $3\times$ *smaller and* $5\times$ *faster.*

1. Introduction

The widespread use of smartphones provides ample opportunities for scalable deployment of vision-based assistive technologies. On-device computer vision can be used to support situational awareness in people who are impaired or elderly. For instance, a smartphone can be worn and used to provide real-time information about the surrounding scene, such as navigational cues or social cues. In this manner, it can enhance mobility independence and overall quality of life. For such applications, this study focuses on detecting an important and useful component of the environment – people.



(a) Images from a body-worn smartphone

(b) Images from PASCAL dataset [1].

Figure 1. Our paper develops an efficient person detection algorithm that can run on a smartphone while being specifically suited to images captured from a body-worn camera perspective. We propose to leverage the appearance distribution of people body parts in our specific application domain, and contrast it with general person detection settings on the PASCAL dataset.

Detecting people in images is crucial for a variety of application domains in computer vision and robotics. State-of-the-art generic object detection techniques [2, 3] are typically evaluated on various pedestrian detection benchmarks [4]. Many of the existing person detection benchmarks have been developed for generic person detection (e.g. PASCAL [5, 1]), surveillance [6], and driving [7, 8, 9, 10]. Depending on the target task, the appearance distribution of people in any given dataset can be greatly skewed.

For example, general object detection datasets will tend to have a uniform distribution over the size of people in images. In surveillance datasets, the camera angle tends to be oblique and the size of person is typically very small relative to the size of the image. Pedestrians imaged for driving applications tend to be distributed near the horizon line and to the left or right side of the image.

Our work identifies several domain specific issues when the dataset is captured in first-person settings (Fig. 1). In particular, we show that the visibility of people and body parts are relatively consistent in size (roughly half the height of the image height) for images taken with a body-worn camera. We can take advantage of that consistency to learn part detectors specialized for body-worn cameras. Our proposed **SmartPartNet**, takes advantage of the natural body part appearance statistics of egocentric images and uses a specific combination of body-part detectors for optimal detection performance. We also observe high levels of motion blur due to a combination of rapid ego-motion and low brightness imaging conditions in indoor environments. **SmartPartNet** utilizes a data augmentation technique using motion blur to make our person detector more robust to motion blur. In addition to addressing these domain specific issues, we also design our person detection algorithm to run on a mobile device. We leverage recent progress in deep neural networks for mobile devices and show that our proposed person detector, **SmartPartNet**, obtains performance similar to state-of-the-art pedestrian detection networks, while being smaller and faster.

2. Related Work

Object Detection. General object detection is one of the most active research areas in computer vision. A survey of object detection techniques is beyond the scope of this paper, but some notable approaches are R-CNN [11], Fast R-CNN [2], and Faster R-CNN [3], all of which employ a region proposal mechanism as a form of attention before the final classification and localization. Other approaches, such as YOLO [12] and SSD [13], achieve fast run-times by not employing a region proposal mechanism. The most recent version of YOLO [14] combines multiple insights from the aforementioned techniques, including a fully convolutional network and anchor boxes for predicting bounding boxes. The baseline detector in this work is based on tiny-YOLO (YOLO with a smaller network) suitable for real-time processing on a smartphone. Tiny-YOLO is implemented with CoreML on an iPhone.

People Detection. Object detectors are often applied to the task of people detection due to a variety of application domains involving observation of human behavior. Efficient pedestrian detectors have often employed boosted channel features [15, 16, 17, 18]. Several recent methods [19, 20, 21] with good performance apply decision forests on top of convolutional features. In recent studies [22, 23, 24], such techniques are employed as region proposals for deep learning techniques. Scene specific detection schemes have also shown promise [25, 26, 27]. Zhang *et al*. [21] showed that introducing higher resolution feature maps and a bootstrapping strategy to Faster R-CNN result in a state-of-the-art detector. Tian *et al*. [24] partitions pedestrian boxes into sub-areas to handle occlusion. Unlike [24], our part-based detector does not train independent part detection models, but instead learn part complementarity in a holistic multi-task training framework. Furthermore, we employ part annotations, not commonly done in pedestrian detection training schemes.

Mobile Object Detection. Our aim in this work is to study challenges in person detection on a mobile device. Model efficiency and size play a critical role for on-device applications. Recent work has focused on reducing the parameters of networks. Iandola *et al*. [28] introduced SqueezeNet, achieving AlexNet-level accuracy on ImageNet with fewer parameters by using smaller filters. Howard *et al*. [29] proposed MobileNets which employ depth-wise separable convolution rather than using traditional convolution for reducing the size of the network.

Part-based Object Detection. Part-based models have been widely researched [30, 23, 31, 32, 33, 34] to improve object detection performance. Recent part-based techniques close to our work are Faceness-Net [35] and DeepParts [24]. These approaches stand out in robustness to detection under partial visibility. Faceness-Net [35] detects facial parts with a single detector per part, later combining the responses to produce face detection boxes. In contrast, we study a more efficient architecture for part-based detection which requires jointly training a single detection model for all the parts.

3. Cross-Dataset Visibility Statistics

In order to study the challenges specific to people detection from a body-worn smartphone camera, we compare the visibility statistics of two pedestrian detection datasets. In particular, we focus on the visibility of body parts across the PASCAL-Part dataset [1] and a Egocentric People Parts dataset created for this work. Fig. 1 illustrates some of the qualitative differences between the people seen from an egocentric video and a general object/person dataset. Next, we will quantitatively contrast and compare various statistics across the datasets to motivate later design decisions for developing our proposed pedestrian detector for body-worn smartphones.

PASCAL People Parts Dataset. To compute the visibility statistics of people in generic object detection dataset, we use a subset of the PASCAL-Parts Dataset [1] that contains

people. For the *people* category there are 24 part segmentation masks. The dataset provides head, upper body parts (torso, arms, hands), and lower body parts (legs and feet). In addition to combining parts to create upper body and lower body boxes, we join the facial parts (nose, eyes, eyebrows, mouth) in order to create a face box.

Egocentric People Parts (EPP) Dataset. Since existing publicly available datasets for people or pedestrian detection were not captured from a body mounted perspective and also do not contain part annotations, we created our own dataset for comparison and evaluation. The video data was collected in a variety of everyday social setting, including walking in hallways, talking and interacting with others while standing or seated, meeting room discussions, eating with a group, and shopping. The overall data captured comprises of about 2 hours of video data, out of which short and diverse clips were selected for a total of 19,980 frames. Head, face, torso, arms+hands, and lower body parts were annotated for each person. Video was captured at 30 frames per second (fps), and every 30^{th} frame was annotated.

3.1. Comparative Statistics

Here we quantify the difference in the size of people (and their parts) between PASCAL-Parts dataset and EPP dataset. To visualize this difference, we plot histograms over the size of people (and their parts) relative to the image height (Fig. 2) as well as part visibility statistics (Supplementary). Since the EPP dataset is taken with a wearable camera during close social interactions, we would expect the size of people in the video to be larger than people in the PASCAL dataset.

The histogram over people and part sizes for the two datasets are shown in Fig. 2. The x-axis represents the relative height of the part and the y-axis represents the normalized histogram count. The blue and red bars represent the height statistics for the PASCAL part and EPP dataset, respectively. For each body part, we observe that there is a significant difference in the height distribution of parts. Contrary to our expectation, we observe from Fig. 2 *person height* (top left graph) that the size of a person the EPP dataset is concentrated around 60% of the image height, whereas for the PASCAL part dataset, the size of people in images are close to uniformly distributed. For other body parts like the torso and arms, we notice that the size body parts are more peaked at certain heights, whereas the corresponding PASCAL parts has a more diffused distribution. A peaked distribution over the appearance of certain parts may indicate that it may be easier to learn those appearance models, since there is less variance in the appearance that must be modeled. We will validate this hypothesis empirically in later experiments. Furthermore, the differences in distributions leads us to explore cross-data generalization and domain-specific training benefits.

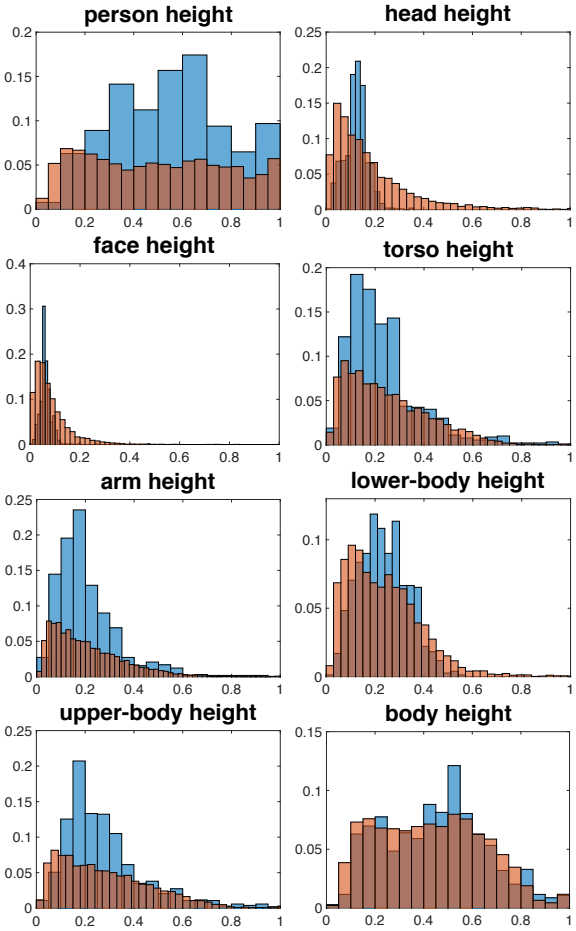


Figure 2. Histogram of the ratio of part heights and image height in the two datasets, PASCAL (red) and our EPP (blue). The x axis is the ratio of part height and image height. The y axis is the normalized value in percentage.

4. Part-Informed Person Detection

The observation that body part appearance statistics tend to have a specific distribution for the wearable camera application domain motivates us to train detectors which can efficiently leverage this phenomenon in order to gain better detection performance. Specifically, we are interested in analyzing whether the distribution differences result in significant impact on cross-dataset training and testing of part-based approaches. We study two techniques for part-informed person detection, one with a multi-task training framework and another with part boxes fusion in test-time. The proposed train and test-time modifications will be shown to significantly improve detection performance without sacrificing run-time speed.

Part-based Training. Common part-based detectors are highly inefficient due to training individual, part-specific end-to-end models [35, 24], which later need to be post-

processed and combined with additional modules. Such approaches are not feasible for on-device deployment, and do not fully utilize complementary feature sharing across the parts.

The efficiency requirement motivated us to consider a different approach, where part supervision is added through modification of the label space. In this framework, instead of single person class the detector output is multi-class, where body parts are treated as different classes (Fig. 3). We note that this modification results in a negligible impact on run-time speed or model size. By providing body part labels, the network can then automatically update the features in order to produce useful shared representations for the person detection task and leverage complmentary information among the part cues. Although similar in nature to multi-task learning, where features can holistically integrate across tasks, it is a different form of part-based detection compared to other state-of-the-art approaches [35, 24] as it does not require a post-processing combination module or significantly changing the architecture. Interestingly, although parts are often claimed to emerge when training a deep network for a supervised high level task, we find that adding parts to the prediction label space result in a significant impact on the detection performance of the original person class. To reiterate, we perform most of the analysis on the person detection task which is our overarching goal, but study the benefits of adding part detection tasks to the network in training time. Related to this approach is the practice of adding auxiliary label targets [36, 37], which is known to help in regularizing training. Unlike the studies [36, 37], our cross-dataset experiments will demonstrate this approach to be highly sensitive to the appearance distribution of parts in the experiments. In the experiments, we train the model with an initial learning rate of $1\mathrm{e}{-3}$, and reduce by a factor of 10 every 30 epochs for a total of 200 epochs.

Test-Time Part Combination Module. YOLO achieves fast run-time performance partly due to an efficient grid-based approach for producing the final detection boxes, as shown in Fig. 4. We find that the task of regressing a full person detection box out of a single cell (YOLO employs a 13×13 grid in the feature space) is difficult, often resulting in poorly localized detection boxes. We therefore propose an efficient part combination module during test time which improves both precision and recall of the person boxes at a negligible computational cost. Furthermore, the combination module is useful in analyzing whether the part-based training approach fully learns to leverage part-based cues for detecting the person class boxes.

Given part detection boxes, we perform a configuration check that results in either refinement of an existing person box or instantiation of a new person box. Formally, the part-informed model generates full person detection boxes, $\{b_i\}_{i=1}^{n}$, and part detection boxes $\{p_i\}_{i=1}^{m}$. A box is composed of the image coordinates, score, and type $[x, y, w, h, s, t]$. As each full person box b_i was regressed from a single cell in the grid, it often struggles to bound the person entirely, especially at body extremities. On the other hand, the part detection boxes occur at *different cells* (Fig. 4), and can therefore aid in localization. We restrict our part boxes by a minimum threshold on the confidence score (0.2) in order to avoid degrading the person boxes with false positives and only processing a small set of part detections for the combination module. Given an existing person box b_i which overlaps (area of Intersection Over Union-IoU) parts $\{p_k\}_{k=1}^{l}$, we determine whether the parts result in an upright person (determined by heuristically checking the relative y-coordinate of the parts, e.g. head is above upper body, upper body above lower body, etc.) and consequently replace b_i box with $\bar{b_i} = \cup_{k=1}^{l} p_k$ and score with $\max_{k=1:l} p_k$. If any of the remaining part boxes produce plausible configurations, determined by comparing upright possibilities against configuration cluster centroids from the training set (produced by k-means) with the Euclidean distance, a new person box is instantiated with a score of $\max_{k=1:l} p_k$. The combination module includes a final non-maximum sup-

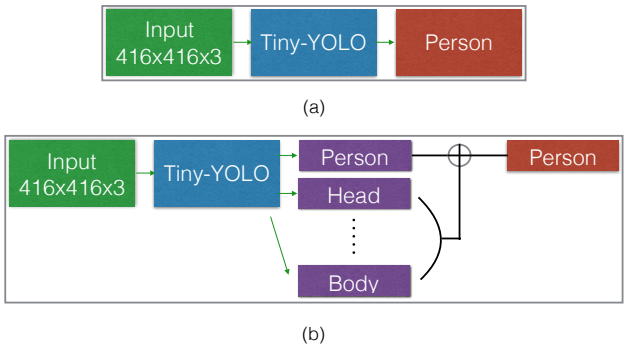


Figure 3. Test-time differences between (a) the baseline person detector and (b) our proposed multi-task part-based approach with part combination module (see Section 4).

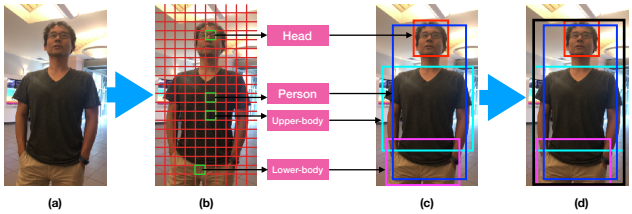


Figure 4. Test-time part combination module. In YOLO, the input image is divided to S x S grid (b). The grid cell containing the center of the target object is responsible to detecting the target object. (c) Result of each part detector. (d) The final person box. Since different parts occur at different cells, combining part boxes often provides better localized boxes.

pression operation. We note that this approach is intended to reflect our application domain while not adding additional computational overhead. The intuitive heuristics allow us to avoid an expensive search over part combinations as done in general detection settings [38, 34] while still significantly improving detection performance.

5. Data Augmentation with Blur Effect

In addition to studying efficient part-based detection approaches for our smartphone settings, we introduce a data augmentation technique to generate synthetic motion-blurred images during training. The augmentation allows for domain-based increase in detection robustness, without increasing the computational run-time cost. We observed object detectors to be highly sensitive to even small levels of motion blur, yet smartphone-captured video often contains blurry images due to the camera motion and long exposure time. To handle this issue, we train the network with synthesized motion-blurred images.

Each training image is pre-processed with motion filter h. The coefficients for the motion filter h are the length l, which defines severeness, and angle θ, which determines the direction of motion blur. A line segment L with the desired length l and angle θ is constructed and centered at the center coefficient of h. For each coefficient location $h(i,j)$, $D_{nearest}(i,j)$ is the nearest distance between that location (i,j) and the line segment L,

$$h(i,j) = \frac{max(1 - D_{nearest}(i,j), 0)}{\sum_{i,j} max(1 - D_{nearest}(i,j), 0)}$$

In training time, we apply the motion filter with a fixed length and a randomly selected angle. We compared three motion blur training sets with three different lengths (100, 150, and 200) and one Gaussian blur training set to the original training set. Fig. 5 visualizes these varying blur levels. We apply motion blur augmentation to our entire training set, regardless of whether an image has originally contained blur. The augmentation will be shown to result in both localization improvement of instances with motion blur and reduction in missed detections due to blur.

6. Experimental Analysis

We begin our analysis by comparing the PASCAL and the EPP dataset in terms of part detection performance, multi-part complementarity, and generalization capability from PASCAL to EPP.

6.1. Part-based Training Results

Impact on PASCAL Person Class Detection. Fig. 15 depicts the results of different part-informed models trained

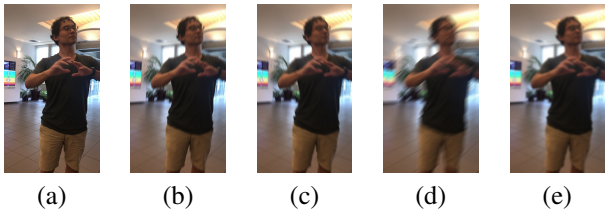


Figure 5. Motion blurring data augmentation significantly impacts detection performance. (a) Original (b) Light motion blur (c) Moderate motion blur (d) Heavy motion blur (e) Gaussian blur.

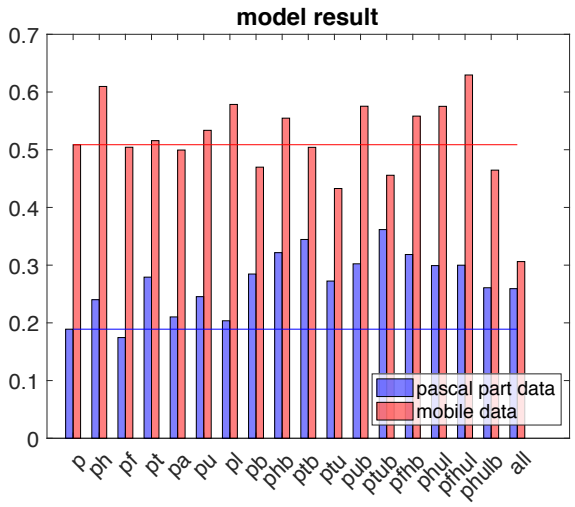


Figure 6. Detection results for the person class of models trained with different part output on the PASCAL training set. Testing is performed on PASCAL and EPP dataset captured in mobile settings. The y-axis shows average precision (AP). Each bar represents the results of a model learned with the following part combinations, p = person, h = head, f = face, t = torso, a = arm, b = body, u = upper body, l = lower body, all = all parts.

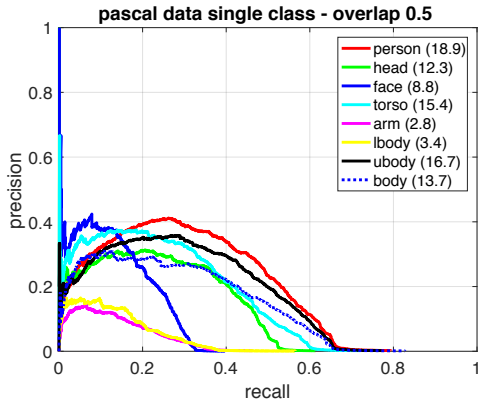
on PASCAL and tested on PASCAL and EPP. The results are compared against the single-class person detector (shown by a line in both plots), and are only shown on the person class. The plots reveal several interesting insights regarding the design of the optimal person detector across different datasets. First we note that the off-the-shelf implementation of tiny-YOLO [14] trained on the person class alone performs poorly on our PASCAL people only validation split. Addition of parts results in significant increase in detection performance, from 18.9 to 36.2 AP when adding torso, upper-body, and body parts. Nonetheless, not all parts are shown to be beneficial, and a combination of all parts shows significant deterioration in detection performance. Simply put, some parts help while others hurt. This phenomenon is not well studied in related literature. In principle, the sharing of features in the multi-task formulation can holistically improve performance across the tasks. Prac-

tically, we find this not to be the case. For instance, holistic object detection approaches may be more sensitive to deterioration in detection performance due to non-useful or poorly recognized parts. Furthermore, they may not generalize well to instances when people are occluded or across datasets with different part visibility distributions. On the other hand, part detectors in related studies [35, 24] often involve independent detection modules, which could better handle cases of non informative part cues. Nonetheless, [35] briefly reports deterioration of performance when over-partitioning the label space into parts and sub-parts (for facial part detection). At the same time, adding auxiliary part tasks is shown to have a large impact on the learned features for the person detection class. While often papers make claims regarding the emergence of parts in deep ConvNets, our study affirms that such strong supervision of most parts is shown to help performance.

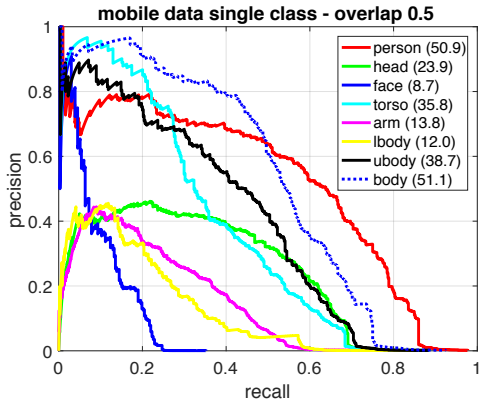
Generalization to EPP. Next, we analyze the role of part-informed training on cross-dataset generalization. The analysis on the EPP dataset (Fig. 15(b)) demonstrates how the difference in part height and visibility distribution results in improvement only due to some selected part combinations. For instance, the head class is shown to highly benefit generalization across the datasets. The face class on the other hand does not, as our dataset contains many instances of people facing away from the camera. The model requires cues from the face, head, upper-body, and lower-body in order to best generalize to the new domain. This experiment also affirms our hypothesis that our domain captured in the EPP dataset has very different characteristics from the general person detection settings in PASCAL.

Correlation with Individual Part Detection Performance. In an attempt to study this interesting phenomenon deeper, we train and test single-class part detectors on both datasets, as shown in Fig. 7. The goal of this experiment is to determine whether the detection quality of independent parts play a key role in their success. Results are shown in Fig. 15 both on PASCAL and EPP. On PASCAL, the most successful parts with two part combinations are the body and torso, followed by upper body and head. Face is the least useful, followed by lower body and arm. Fig. 7 depicts the correlation between individual part detection results and part combination results. We can conclude that the combination of the best performing classes also results in the a high performing part-informed person detector. Hence, performance of part-informed training models is influenced both by the ability of the model to capture part-specific appearance cues, as well as the appearance distribution of parts.

Best Parts Combination for Body-Worn Smartphone Settings. On the EPP dataset, Fig. 15(b) depicts head, lower body, upper body, and torso to be the most successful classes when added to the person class detection tasks



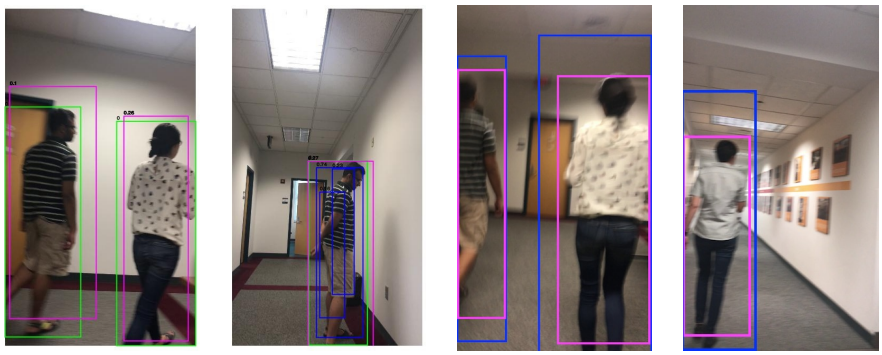
(a)



(b)

Figure 7. Precision-recall curves of single part detection models on (a) the PASCAL dataset and (b) our EPP dataset. Average precision (AP) is shown in the legend. Note that each model is evaluated on their corresponding part class in the test set.

in a two part combination. Yet, when inspecting individual part detection performance in 7(b), the top performing class is actually body (a part composed by combining the upper and lower body, without the head). Furthermore, the previous best combination of person+torso+upper+body does not generalize well at all. Finally, some of the classes which perform poorly are showing to holistically combine with those which perform well, resulting in the best performing combination of person+head+face+upper+lower (63.0 AP compared to the 50.1 AP baseline). Example results of this model are visualized in Fig. 8. A combination of all parts overfits to the training data so badly that it produces the worst performing combination on EPP. We conclude that part-informed person detectors are highly sensitive to the dataset distribution and could result in poor generalization, even when individual part detectors produce reliable results. Furthermore, single part classes over with poor detection performance can holistically combine with other parts to improve generalization across datasets to new application



(a) (b)

Figure 8. (a) Detection results of the single-class person detector baseline (in blue) and the part-informed (pfhul, in pink) detector. The ground truth boxes are shown in green. (b) Localization results improve due to blur augmentation. Detection results with/without the motion blur-based augmentation are shown in blue/purple, respectively.

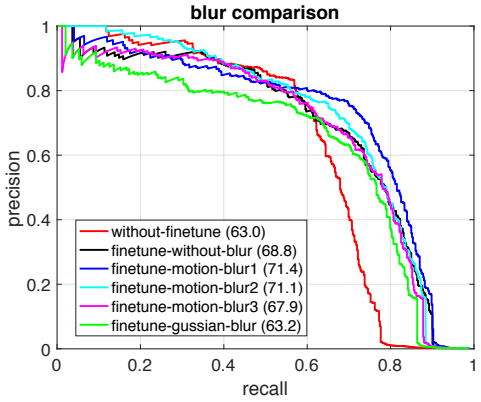


Figure 9. Precision-recall curves with different kinds of blur-based augmentation on the EPP dataset. Motion-blur1, Motion-blur2, and Motion-blur3, refer to increasing levels of motion-blur amount of low, moderate, and high, respectively.

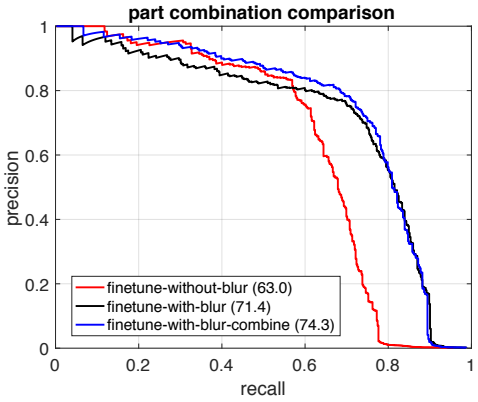


Figure 10. **SmartPartNet** leverages a final step of part detections combination for improving localization and recovering missed pedestrians.

domains. In practice, the results reflect the variations in the types of people (i.e. visibility patterns) in each dataset. The supplementary contains a finer-grained breakdown of detection performance for different visibility patterns in EPP.

6.2. Motion Blur-based Data Augmentation

Person detection deteriorates even under slight motion blur artifacts. We consider this challenge as another opportunity to specialize our detector to the application domain, and analyze whether appropriate data augmentation can alleviate the issue. In this experiment, the model is fine-tuned on the EPP dataset with varying levels of blur. Starting from the PASCAL-trained model, we continue by fine-tuning the model for another 100 epochs. As shown in Fig. 9, fine-tuning benefits precision at both high and low recall. The maximum length of the motion blur is varied to be 100, 150, and 200 pixels, where we observe performance degradation. While fine-tuning alone results in a 5.8 AP improvement, motion-blur data augmentation further improves by 2.7 AP points. Gaussian blur on the other hand is not shown to significantly impact the performance as its charchteristics are quite different from motion blur.

6.3. SmartPartNet

Our final **SmartPartNet** detector is composed of the combined contributions described in Sections 4 and 5. In addition to part-informed training and motion blur handling, we analyze a final part combination module in test time. The goal of this experiment is to find out whether the multi-task part-informed training fully leverages relationships between parts, as well as further improve detection performance with neligble addition in computation. As shown in Fig. 10, this module results in an additional improvement of 3.2 AP points. Therefore, the multi-task formulation, while beneficial, is shown to not be able to fully capture reasoning

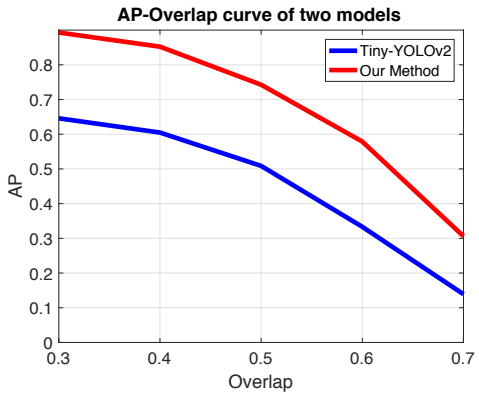


Figure 11. Comparison between the baseline (tiny-YOLO) and **SmartPartNet** on the EPP dataset for different overlap requirements for a true positive detection (0.7 overlap threshold requires better localization).

among parts. This final important step results in our complete **SmartPartNet** performance at 74.3 AP on the EPP dataset. As shown in Fig. 11, the improvement is consistent across varying overlap thresholds (IoU localization requirements) for ground truth boxes. Fig. 12 visualizes some example scenarios in which the test-time part combination module improves detection performance. Table 1 performs an overall summary of the different components of **SmartPartNet**, compared against both the YOLO and tiny-YOLO baselines. Our method performs nearly as well as the full YOLO model, which currently achieves state-of-the-art results on a variety of detection tasks and datasets, including PASCAL and COCO [12]. Run-time is not significantly impacted by the proposed components of **SmartPartNet**, and the part-informed training results in a minor increase in model size due to the addition of the part classes to the last layer. In order to ensure generalization of the proposed contributions, we also employ the MOT [39, 40] challenge dataset, with detection curves shown in the supplementary. Overall, our proposed approach demonstrates consistent improvement in detection performance. For instance, on the ETH-Bahnhof part of the dataset we achieve 62.3 AP with our **SmartPartNet** detector compared to the 48.1 AP with the tiny-YOLO baseline.

7. Conclusion

Real-time person detection on a smartphone can be used for several assistive application domains. This paper deals with challenges specific to person detection from a body-worn, egocentric perspective. A part-informed training procedure resulted in significant detection performance gains in a state-of-the-art detector. The ablative analysis also revealed insights into issues with training generalizable part-based person detectors. A part combination module in test time and a motion blur-based data augmentation provided further gains.

Method	Speed (fps)	Accuracy (AP)
YOLO	40 (GPU)	79.7
tiny-YOLO	>200 (GPU)	50.9
tiny-YOLO + parts	>200 (GPU)	63.0
tiny-YOLO	15 (phone GPU)	50.9
tiny-YOLO + parts	15 (phone GPU)	63.0
tiny-YOLO + parts + ft	15 (phone GPU)	68.8
tiny-YOLO + parts + ft + blur	15 (phone GPU)	71.4
SmartPartNet	15 (phone GPU)	74.3

Table 1. Comparison of performance and run-time on a Titan X and a smartphone GPU with different components of the proposed approach; parts, fine-tuning (ft), motion blur handling, and test-time parts combination which is shown as the final **SmartPartNet** results. Each component is shown to significantly improve performance with negligible impact on the run-time or network size.

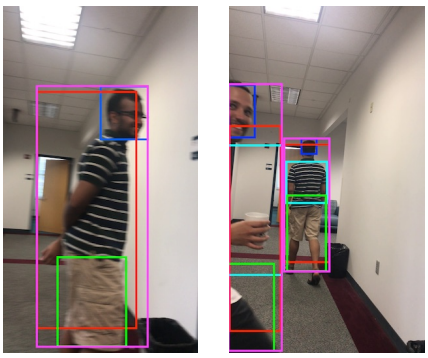


Figure 12. Results with the test-time parts combinations module. The baseline person, head, upper body, and lower body detection boxes are shown in red, blue, cyan, and green, respectively.

In the future, **SmartPartNet** will be improved to handle further domain-specific challenges. Further detection performance gains can be achieved using online on-device training, specializing to certain people and scenarios which may repeat on a daily basis. The parts detector can also be used to perform a fine-grained analysis of the surrounding scene useful for assistive technologies (e.g. people gestures and expressions), although the addition of further tasks in training time needs to be studied carefully. Finally, we would like to study the usefulness of the output provided by **SmartPartNet** in real-world assistive settings, e.g. as a navigational aid to visually impaired people.

8. Acknowledgement

This work was sponsored in part by JST CREST grant (JPMJCR14E1), NSF NRI grant (1637927), and DNDO ER grant (2017-DN-077-ER0001).

References

[1] X. Chen, R. Mottaghi, X. Liu, S. Fidler, R. Urtasun, and A. Yuille, "Detect what you can: Detecting and representing objects using holistic models and body parts," in *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2014.

[2] R. Girshick, "Fast r-cnn," in *The IEEE International Conference on Computer Vision (ICCV)*, December 2015.

[3] S. Ren, K. He, R. Girshick, and J. Sun, "Faster r-cnn: Towards real-time object detection with region proposal networks," in *Advances in Neural Information Processing Systems*, pp. 91–99, 2015.

[4] R. Benenson, M. Omran, J. Hosang, e. L. Schiele, Bernt", M. M. Bronstein, and C. Rother, "Ten years of pedestrian detection, what have we learned?," in *European Conference on Computer Vision*, 2014.

[5] M. Everingham, S. M. A. Eslami, L. Van Gool, C. K. I. Williams, J. Winn, and A. Zisserman, "The pascal visual object classes challenge: A retrospective," *International Journal of Computer Vision*, vol. 111, pp. 98–136, Jan. 2015.

[6] K. Vignesh, G. Yadav, and A. Sethi, "Abnormal event detection on BMTT-PETS 2017 surveillance challenge," in *IEEE Conference on Computer Vision and Pattern Recognition Workshops (CVPRW)*, pp. 2161–2168, July 2017.

[7] A. Geiger, P. Lenz, and R. Urtasun, "Are we ready for autonomous driving? the KITTI vision benchmark suite," in *Conference on Computer Vision and Pattern Recognition (CVPR)*, 2012.

[8] P. Dollar, C. Wojek, B. Schiele, and P. Perona, "Pedestrian detection: An evaluation of the state of the art," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 34, pp. 743–761, April 2012.

[9] M. Cordts, M. Omran, S. Ramos, T. Rehfeld, M. Enzweiler, R. Benenson, U. Franke, S. Roth, and B. Schiele, "The cityscapes dataset for semantic urban scene understanding," in *Proc. of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2016.

[10] S. Zhang, R. Benenson, and B. Schiele, "Citypersons: A diverse dataset for pedestrian detection," *CoRR*, vol. abs/1702.05693, 2017.

[11] R. Girshick, J. Donahue, T. Darrell, and J. Malik, "Rich feature hierarchies for accurate object detection and semantic segmentation," in *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2014.

[12] J. Redmon, S. Divvala, R. Girshick, and A. Farhadi, "You only look once: Unified, real-time object detection," in *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2016.

[13] W. Liu, D. Anguelov, D. Erhan, C. Szegedy, S. Reed, C.-Y. Fu, and A. C. Berg, "Ssd: Single shot multibox detector," in *European Conference on Computer Vision*, pp. 21–37, Springer, 2016.

[14] J. Redmon and A. Farhadi, "YOLO9000: better, faster, stronger," in *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, July 2017.

[15] J. W. Davis and M. A. Keck, "A two-stage template approach to person detection in thermal imagery," in *WACV*, 2005.

[16] P. Dollar, Z. Tu, and S. Belongie, "Integral channel features," in *In British Machine Vision Conference (BMVC)*, 2009.

[17] P. Dollar, R. Appel, and P. Perona, "Fast feature pyra- mids for object detection," in *IEEE Transactions on Pattern Analysis and Machine Intelligence (TPAMI)*, 2014.

[18] W. Nam, P. Dollar, and J. H. Han, "Local decorrelation for improved pedestrian detection," in *Neural Information Processing Systems (NIPS)*, 2014.

[19] Z. Cai, M. Saberian, and N. Vasconcelos, "Learning complexity-aware cascades for deep pedestrian detection," in *Proceedings of the IEEE International Conference on Computer Vision*, pp. 3361–3369, 2015.

[20] B. Yang, J. Yan, Z. Lei, and S. Z. Li, "Convolutional channel features," in *Proceedings of the IEEE international conference on computer vision*, pp. 82–90, 2015.

[21] L. Zhang, L. Lin, X. Liang, and K. He, "Is faster r-cnn doing well for pedestrian detection?," in *European Conference on Computer Vision*, pp. 443–457, Springer, 2016.

[22] J. Hosang, M. Omran, R. Benenson, and B. Schiele, "Taking a deeper look at pedestrians," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 4073–4082, 2015.

[23] Y. Tian, P. Luo, X. Wang, and X. Tang, "Pedestrian detection aided by deep learning semantic tasks," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 5079–5087, 2015.

[24] Y. Tian, P. Luo, X. Wang, and X. Tang, "Deep learning strong parts for pedestrian detection," in *IEEE International Conference on Computer Vision (ICCV)*, pp. 1904–1912, Dec 2015.

[25] Y. Mao and Z. Yin, "Training a scene-specific pedestrian detector using tracklets," in *WACV*, 2015.

[26] A. Setia and A. Mittal, "Co-operative pedestrians group tracking in crowded scenes using an mst approach," in *WACV*, 2015.

[27] X. Zeng, W. Ouyang, M. Wang, and X. Wang, "Deep learning of scene-specific classifier for pedestrian detection," in *European Conference on Computer Vision*, pp. 472–487, Springer, 2014.

[28] F. N. Iandola, M. W. Moskewicz, K. Ashraf, S. Han, W. J. Dally, and K. Keutzer, "Squeezenet: Alexnet-level accuracy with 50x fewer parameters and <1mb model size," *CoRR*, vol. abs/1602.07360, 2016.

[29] A. G. Howard, M. Zhu, B. Chen, D. Kalenichenko, W. Wang, T. Weyand, M. Andreetto, and H. Adam, "MobileNets: Efficient convolutional neural networks for mobile vision applications," *CoRR*, vol. abs/1704.04861, 2017.

[30] P. F. Felzenszwalb, R. B. Girshick, D. McAllester, and D. Ramanan, "Object detection with discriminatively trained part-based models," *IEEE transactions on pattern analysis and machine intelligence*, vol. 32, no. 9, pp. 1627–1645, 2010.

[31] A. Mohan, C. Papageorgiou, and T. Poggio, "Example-based object detection in images by components," *IEEE transactions on pattern analysis and machine intelligence*, vol. 23, no. 4, pp. 349–361, 2001.

[32] K. Mikolajczyk, C. Schmid, and A. Zisserman, "Human detection based on a probabilistic assembly of robust part detectors," *European Conference on Computer Vision*, pp. 69–82, 2004.

[33] M. Enzweiler, A. Eigenstetter, B. Schiele, and D. M. Gavrila, "Multi-cue pedestrian classification with partial occlusion handling," in *Proceedings of the IEEE conference on Computer Vision and Pattern Recognition*, pp. 990–997, IEEE, 2010.

[34] R. Girshick, F. Iandola, T. Darrell, and J. Malik, "Deformable part models are convolutional neural networks," in *Proceedings of the IEEE conference on Computer Vision and Pattern Recognition*, pp. 437–446, 2015.

[35] S. Yang, P. Luo, C.-C. Loy, and X. Tang, "From facial parts responses to face detection: A deep learning approach," in *Proceedings of the IEEE International Conference on Computer Vision*, pp. 3676–3684, 2015.

[36] Z. C. Lipton, D. C. Kale, C. Elkan, and R. Wetzell, "Learning to diagnose with LSTM recurrent neural networks," *ICLR*, 2016.

[37] R. Caruana, S. Baluja, and T. Mitchell, "Using the future to "sort out" the present: Rankprop and multitask learning for medical risk evaluation," in *Advances in neural information processing systems*, 1996.

[38] P. Felzenszwalb, D. McAllester, and D. Ramanan, "A discriminatively trained, multiscale, deformable part model," in *Computer Vision and Pattern Recognition*, 2008.

[39] A. Ess, B. Leibe, and L. V. Gool, "Depth and appearance for mobile scene analysis," in *International Conference on Computer Vision (ICCV)*, 2007.

[40] L. Leal-Taixé, A. Milan, I. Reid, S. Roth, and K. Schindler, "Motchallenge 2015: Towards a benchmark for multi-target tracking," *arXiv preprint arXiv:1504.01942*, 2015.

[41] A. Milan, L. Leal-Taixe, I. Reid, S. Roth, and K. Schindler, "Mot16: A benchmark for multi-object tracking. arxiv 2016," *arXiv preprint arXiv:1603.00831*.

9. Supplementary

9.1. Validation of SmartPartNet's Performance on Two Additional Datasets

The main paper studied the PASCAL and EPP datasets in detail, but mentioned consistent improvement on two additional datasets from the MOT challenge [40, 41]. Figure 13 shows the full performance curves on the ETH-Bahnhof [39] dataset, and Figure 14 on the MOT16-11 dataset, which also provides a Fast R-CNN baseline [2]. The technical report in [41] contains all the details regarding the dataset and the training and testing parameters of the baseline Fast R-CNN. Overall, our proposed approach demonstrates consistent improvement in detection performance over the original tiny-YOLO baseline for pedestrian detection.

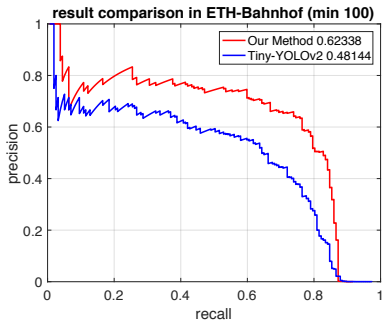


Figure 13. Person detection results of tiny-yolo baseline and our **SmartPartNet** method on the ETH-Bahnhof dataset.

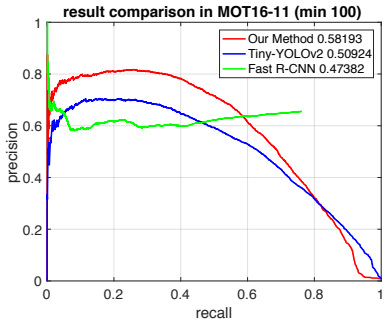


Figure 14. Person detection results of tiny-yolo baseline, Fast RCNN baseline and our **SmartPartNet** method on the MOT16-11 dataset.

9.2. Supplementary to Figure 6

The idea behind the initial experiments in Section 6 of the paper is to study the benefits of the multi-task part-based training on detection performance on PASCAL, as well as study its role on cross-data generalization, specifically to the EPP dataset captured in mobile settings. Figure 15 shows the detection precision recall curves for different part combinations, visualized more clearly in a bar plot in Figure 6 of the main paper.

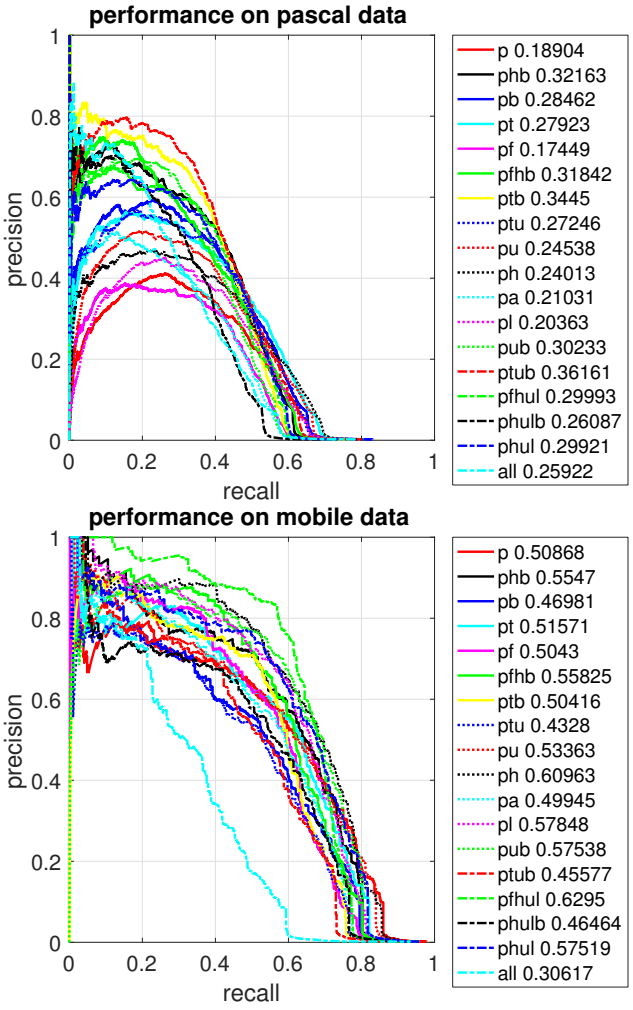


Figure 15. Precision-recall curves of models with different parts output on pascal part dataset and our mobile dataset. AP = average precision. p = person. h = head. f = face. t = torso. a = arm. b = body. u = upper body. l = lower body. all = all parts. The values in the legend are average precisions.

9.3. Detection in Different Part Visibility Settings - Supplementary to Section 6

In order to quantify the detection performance improvement on different types of visibility patterns of parts, Figure 16 compares the single-class person detector baseline with the best part-informed detector on the EPP dataset. The figure isolates the improvement in performance due to detecting people with upper body and head visible as well as full visibility. These instances of people allow the combination model to apply the learned relationships between the parts. No improvement is shown for people with no head visible occurring at truncated instances (e.g. social interactions).

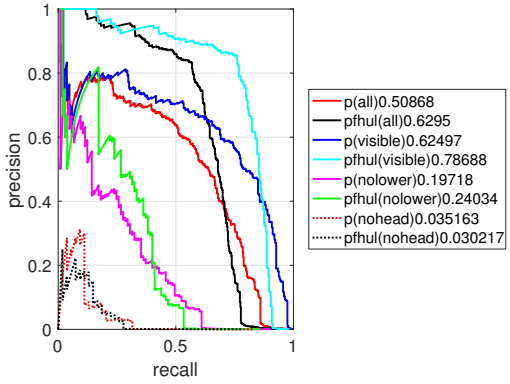


Figure 16. Precision-recall curves on subsets of people in EPP with different part visibility patterns. We compare the baseline single-class person detector, p, and the best part-informed detector pfhul (abbreviated part notation; person, face, head, upper body, and lower body). In parenthesis is the type of part visibility pattern. (all) - entire dataset, (visible) - only people with all parts visible, (nolower) - people without lower body visible, and (nohead) - people without head visible. We can observe how performance due to part-based training does not occur on instances where the head is not visible, such as during close social interaction settings.

9.4. Cross-Dataset Visibility Statistics Comparison - Supplementary to Section 3

The visibility distribution of parts is another factor contributing to the cross-dataset performance changes. Our dataset is shown to contain more pedestrians without a head and one arm (due to close social settings), as shown in Figure 17.

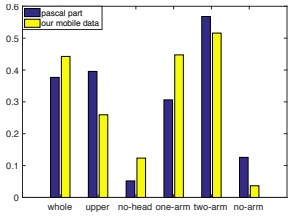


Figure 17. Comparing part visibility distribution of PASCAL and the EPP dataset. Whole are instances with all parts visible, upper are instances without lower body visible, no-head are without head visible, and one-arm are cases with only one arm visible. Two-arms and no-arms cases are similarly defined.