Detector-in-Detector: Multi-Level Analysis for Human-Parts

Xiaojie Li[1][0000−0001−6449−2727], Lu Yang[2][0000−0003−3857−3982], Qing Song[2][0000000346162200], and Fuqiang Zhou[1][0000−0001−9341−9342]

[1] Beihang University, Beijing 100191, China
[2] Beijing University of Posts and Telecommunications Beijing 100876, China
{xiaojieli,zfq}@buaa.edu.cn
{soeaver,songqing512}@bupt.edu.cn

Abstract. Vision-based person, hand or face detection approaches have achieved incredible success in recent years with the development of deep convolutional neural network (CNN). In this paper, we take the inherent correlation between the body and body parts into account and propose a new framework to boost up the detection performance of the multi-level objects. In particular, we adopt region-based object detection structure with two carefully designed detectors to separately pay attention to the human body and body parts in a coarse-to-fine manner, which we call Detector-in-Detector network (DID-Net). The first detector is designed to detect human body, hand and face. The second detector, based on the body detection results of the first detector, mainly focus on detection of small hand and face inside each body. The framework is trained in an end-to-end way by optimizing a multi-task loss. Due to the lack of human body, face and hand detection dataset, we have collected and labeled a new large dataset named *Human-Parts* with 14,962 images and 106,879 annotations. Experiments show that our method can achieve excellent performance on *Human-Parts*.

Keywords: Convolutional neural network · Detector in Detector · Human parts.

1 Introduction

Robust detection of human body, face and hand in the wild are canonical subproblems of general object detection. They are prerequisite for various person-based tasks such as pedestrian detection [1], person re-identification [2,3], facial landmarking [4] and driver behavior monitoring [5]. The problems of human parts detection in the wild have been intensely studied for decades and significant progress have been made in recent detection algorithms due to the advancement of deep Convolutional Neural Networks (CNN) such as [1,6] in person detection, [7,8] in face detection and [9,10,11] in hand detection.

Human parts are multi-level objects [12], where face and hand are sub-objects of body. There are many other multi-level objects in our daily life such as laptop and keyboard, lung and lung-nodule or bus and wheel, which are shown in Fig. 1. However, most detection frameworks ignore the inherent correlation between multi-level objects and coarsely treat these **sub-objects** and **objects** as normal objects when they solve this multi-level objects detection problem. In this paper, we perform the person, face and hand detection tasks together to explore the more efficient detection methods for the multi-level objects.

When doing this multi-level objects task using general detection algorithm, detection performance for large objects, such as the human body, is relatively straightforward. The crucial challenges in real-world applications mainly come from training detectors for small objects such as face and hand due to large pose variations and serious occlusions, which make them still far from achieving the same detection capabilities as a human. Due to the large scale variance between the body (objects) and small body parts (sub-objects), the whole image is mainly occupied by the big objects like human body. Small hand and face usually occupy a relatively smaller area in practice. Thus, there are more background information than the small objects during training, which results in a serious disturbance when doing small objects detection. To cope with the problems, inspired by the top-down pose estimation approaches that first locate and crop all persons from image and then solve the single person pose estimation problem in the cropped person patches [3,2], we proposed a region-based convolutional neural network, named Detector-in-Detector Network (**DID-Net**), which allows the network to see the body in the pictures first and then look inside these bodies to find the tiny faces and hands. Our network contains two region-based detectors: *BodyDetector* and *PartsDetector*. The *BodyDetector* adopt the traditional Faster R-CNN implementation to predict a set of human body bounding boxes. Then the feature maps of the detected bodies are wrapped and sent to the second region-based *PartsDetector* to predict the bounding boxes of hand and face. In *PartsDetector*, many background regions that are useless for training are cut down. Thus there is less disturbance inside the cropped body features, which is beneficial to the detection of small parts. The whole network is trained in an end-to-end way.

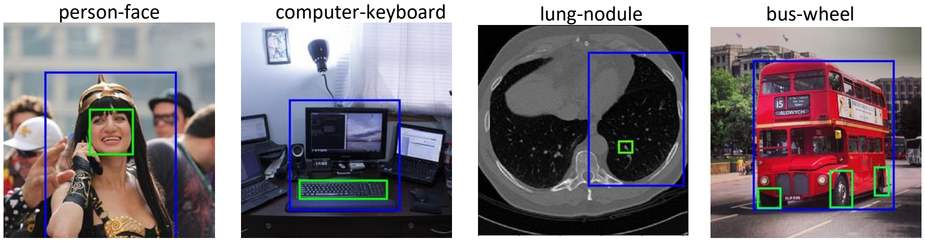

Fig. 1. Examples of multi-level objects. Boxes in green are sub-objects of boxes in blue.

In order to demonstrate the proposed method in more practical scenes, we construct a new *Human-Parts* dataset, which contains 14,962 well-labeled images for human body, face and hand in realistic unconstrained conditions. To the best of our knowledge, it is the first detection dataset that combines three important parts of human. The dataset is available at `https://github.com/xiaojie1017/Human-Parts`.

2 Related Work

2.1 Object Detection methods

CNN-based object detection algorithms fall into two categories. The first category is built upon a two-stage pipeline [14,20,40], where the first stage proposes candidate object bounding boxes and the second stage extracts features using RoIPool from each candidate box and performs classification and bounding-box regression. The second category divides the image into a grid, then simultaneously makes prediction for each square or rectangle in the grid, and finally figures out the bounding boxes of targeting objects based on the predictions of the squares or rectangle, such as [15,16,17]. That design will achieve relatively faster computational speed. However, the one-stage methods can not achieve comparable accuracy with two-stage approaches. In this regard, most of the existing human parts detection methods employ the two-stage pipeline such as [8,18,19].

2.2 Human-Parts detection methods

Faster R-CNN [20] trains the object proposal and classifier at the same time on the same base network. Region Proposal Networks (RPN) in the Faster R-CNN [20] are successful in eliminating the need for a precomputed object proposals. [18,19,8] as well as our work are all the frameworks extended from Faster R-CNN framework. Other object-proposal-free detectors for human parts such as [21,22], perform detection in a fully convolutional manner as RPN does for proposing bounding boxes. Cascaded stages design algorithms are widely used in human parts detection tasks recently such as [23,24]. The advantage of these cascaded stages lies in that they can handle unbalanced distribution of negative and positive samples. In the low-resolution stages, weak classifiers can reject most false positives. In the high-resolution stages, stronger classifiers can save computation with fewer proposals. [23] employs a cascade of classifiers to efficiently reject backgrounds. In [24] three stages of carefully designed CNNs are used to predict face and landmark in a coarse-to-fine manner. However, most of detection methods perform multi-level objects detection together without taking the inherent correlation between them into account.

2.3 Top-down pose estimation methods

Top-down human pose estimation algorithms perform pose estimation task using two separated networks: one person detection network and one single person pose estimation network. Persons are first detected in an image with the detector.

Then the detected person will be cropped from the image and the single person pose estimation network is used to predict the keypoints of each person. A number of top-down algorithms achieve excellent performance on COCO [25] keypoint benchmark such as [2,3]. By sharing features efficiently, Mask-RCNN [26] can do this task in an end-to-end approach. It first generates proposals of person. Then it predict K masks for each proposal, one for each of K keypoint types, which is a Pose-estimator-in-Detector structure.

As a consequence, instead of conducting traditional cascaded stages detection methods that refine predictions step by step, we deal with the multi-level objects detection task using a Detector-in-Detector network. That approach perform parts detection inside each person proposal inspired by the top-down pose estimation methods. Details will be discussed in Section 3.

3 Detector-in-Detector Network

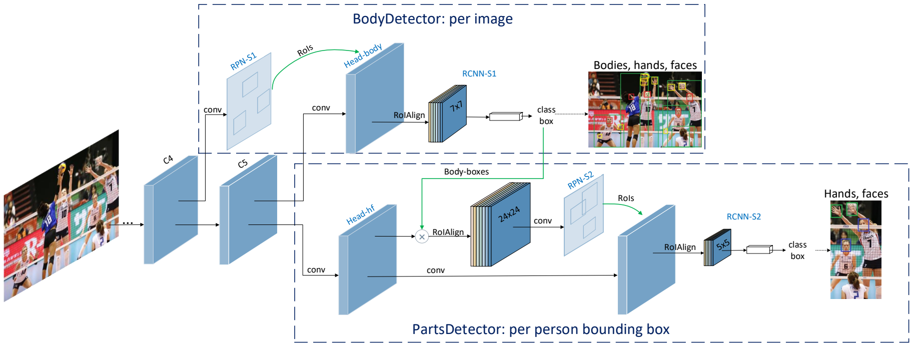

Fig. 2. The pipeline of the proposed DID-Net framework. (1)*BodyDetector*, which contains RPN-S1 and RCNN-S1, is used to detect body, hand and face. (2)*BodyDetector*, which contains RPN-S2 and RCNN-S2, is used to predict hand and face inside the body bounding boxes detected by *BodyDetector*.

The architecture of the proposed Detector-in-Detector network (DID-Net) is illustrated in Fig. 2. We extended Faster RCNN structure by appending another light-weight parts detector after it, which is a Detertor-in-Detector structure. The framework contains two detectors: (1) *BodyDetector* is designed for human body, hand and face detection. (2) *PartsDetector* is appended for hand and face detection inside the selected body bounding boxes, which are detected by *BodyDetector*.

3.1 BodyDetector

BodyDetector uses the same structure with Faster RCNN. Our backbone network is Resnet-50 [13], which is pre-trainied on ImageNet [27]. First, the backbone network generates a series of convolutional feature maps at several scales. We denote these feature maps as (C2, C3, C4, C5) which is extracted from last layer of each scale stage. To increase the resolution of the C5 feature maps, we reduce the effective stride from 32 pixels to 16 pixels as done in [28]. Consequently, their corresponding strides of each layer are (4, 8, 16, 16) pixels with respect to the input image. C4 features are used to predict a set of proposals using the region proposal network, which we named it RPN-S1. Then features of each proposal are cropped from Head-body and pooled into a fixed-size feature vector using RoI Align (We use 7×7 size as [20] does). The features are then fed into RCNN-S1 to do per-proposal classification operation and box refinement of body, hand and face. For model compactness, Head-body is generated after a 1×1 convolutional layer with 256 kernels over the top features of C5 to reduce the dimension. The RoI Align layer is added on the top of the feature maps of Head-body. Two new fully connected (fc) layers of dimension 1024 are applied on the RoI Align features, followed by the bounding box regression and classification branches.

3.2 PartsDetector

PartsDetector shares the same convolutional features of the first stage by simply adding the **Head-hf** to C5 in parallel with **Head-body**, which contains 256 channels features as the same as **Head-body**. The predicted body bounding boxes of the first stage with high classification scores will be chosen. Non-maximum suppression (NMS) [29] is performed on the chosen proposals to eliminate highly overlapped detection bounding boxes. After box regression, the features of person proposals will be wrapped from Head-hf with RoI Align and pooled into size 24×24. RPN-S2 is applied on the wrapped body features to generate RoIs of hand and face. Then the features of RoIs will be extracted from another set of features, which are 256 channels features outputted by adding two 3×3 convolutional layers over Head-hf. Two fully connected layers are appended after the extracted RoI features for further classification and box regression using RCNN-S2. In this stage, we use size 24×24 for the RoI Align size of body and use size 5×5 for the cropped features of hand or face due to the scale variance between them.

During training, we select the top 16 body detections within each image considering the dataset we used, which usually contains a maximum number of 11 persons in one image. While the positions of body bounding boxes are generated during training and are uncertain before. To handle this, we generate the ground truth annotations inside the predicted human body in an online way. When the original face and hand ground truth lie inside or partly inside the detected body bounding boxes, They will be regarded as the ground truth annotations of that body. After transferring the coordinate of these hand and face boxes from the original image to the body bounding boxes, a new batch of training data for hand and face inside each body can be sent to the *PartsDetector* for parts detection. In this detector, we treat every detected human body as one input image in the first detector when writing the implementation code. NMS is applied to eliminate highly overlapped detection bounding boxes during inference and training.

3.3 Loss function

Our network is trained to minimize a multi-task loss, which is composed of classification and bounding-box regression losses of RPN and RCNN modules from the two detectors. We use softmax cross-entropy loss for classification and smooth *L1* loss [20] for bounding-box regression. For the RPN stage in two detectors, the classification loss is softmax cross-entropy loss over background/foreground classes. The loss can be formulated as Equation 1. Here, $\mathcal{L}^{s1}_{rpn}$ and $\mathcal{L}^{s1}_{cls}$ denote the losses of RPN-S1 and RCNN-S1 in *BodyDetector*, and $\mathcal{L}^{s2}_{rpn}$ and $\mathcal{L}^{s2}_{cls}$ denote losses of RPN-S1 and RCNN-S2 in *PartsDetector*.

$$\mathcal{L} = \mathcal{L}^{s1}_{rpn} + \mathcal{L}^{s1}_{cls} + \mathcal{L}^{s2}_{rpn} + \mathcal{L}^{s2}_{cls} \tag{1}$$

4 Experiments

4.1 Human-Parts Datasets

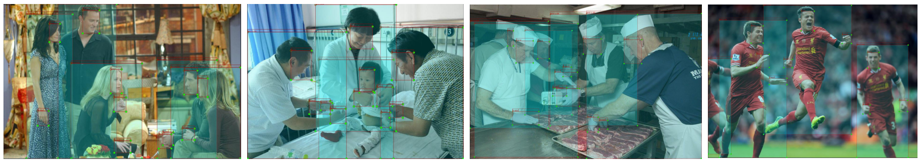

Fig. 3. Samples of annotated images in *Human-Parts* dataset.

In this paper, we collected and labeled a detection dataset named *Human-Parts* which contains annotations of three categories, including person, hand and face. The proposed dataset contains high-resolution images which are randomly selected from AI-challenger [35] dataset. Person category has already been labeled in this dataset. However, the small human whose body parts are hard to distinguish or the vague ones whose body contours are hard to recognize are missed-labeled in this dataset. We added the missed person body annotations and labeled hand and face additionally in each image. The number of persons in each image range from 1 to 11. In total, our dataset consists of 14,962 images (we use 12,000 for train, 2,962 for testing) with 10,6879 annotations (35,306 persons, 27,821 faces and 43,752 hands). Our dataset followed the same standard annotation principles as Wider Face [34] and VGG Hand dataset [11]. We have labeled every visible person, hand or face with xmin, ymin, xmax and ymax coordinates and ensured that annotations cover the entire objects including the blocked parts but without extra background. The annotation format follows

PASCAL VOC [36]. The representative images and annotations are shown in Fig 3. More details about this dataset and comparisons with other human parts datasets can be seen in Table 1.

Table 1. Comparison of different human parts detection datasets

DataSet	Images	Person	Hand	Face	Total Instance
Caltech [30]	42,782	✓	-	-	13,674
CityPersons [31]	2,975	✓	-	-	19,238
VGG Hand [11]	4,800	-	✓	-	15,053
EgoHands [32]	11,194	-	✓	-	13,050
FDDB [33]	2,854	-	-	✓	5,171
Wider Face [34]	32,203	-	-	✓	393,703
Human Parts	14,962	✓	✓	✓	106,879

4.2 Implementation Details

We perform all experiments on *Human-Parts* dataset. Training and evaluation are performed on the 12,000 images in the *train* set and the 2,962 images on the *test* set. For evaluation, we use the standard average precision(AP) and mean average precision(mAP). We report AP and mAP scores using the intersection over union (IoU) [36] threshold at 0.5. All networks are fine-tuned from a pre-trained ImageNet classification network ResNet-50. Our system is implemented in Pytorch and source code will be made publicly available.

For anchor generation, we use scales (64^2, 128^2, 256^2, 512^2) in the *BodyDetector* stage and (32^2, 64^2, 128^2, 256^2) in the *PartsDetector* stage considering the scale variance of body and body parts. All anchors have the aspect ratio of (0.5, 1, 2) due to the large variations of hand and face in the wild. During training, the mini-batch size of 512 is employed for the RCNN-S1 stage of *BodyDetector*, and 32 is employed for each person RoI in the RCNN-S2 stage of *PartsDetector*. Multi-scale training strategy is adopted to be robust to different scale objects. In our method, the shorter side is resized to (416, 480, 576, 688, 864, 1024) pixels. We only use horizontal image flipping augmentation. In the testing phase, the shorter side of each image is resized to 800 pixels and tested independently. For other settings, we follows the work [20].

During training, we adopt an end-to-end learning procedure using stochastic gradient descent (SGD). The whole network is trained for 30,000 iterations with initial learning rate of 0.01. We decay the learning rate by 0.1 at 20,000 iterations. A momentum of 0.9 and a weight decay of 0.0005 are used. During inference, *BodyDetector* module outputs 300 best scoring anchors as detections. While in the *PartsDetector* module, we only select 30 best scoring anchors on account of limited parts inside each body region. The results of hand and face in

BodyDetector and *PartsDetector* are fused together and NMS with a threshold of 0.45 is performed on the outputs due to the existence of large overlap objects in the wild such as Fig. 3.

4.3 Main results

In Table 2, we perform studies of traditional Faster RCNN training with different data and our DID-Net. *Separate Network* adopts only one category detection task in one ResNet-50 Faster R-CNN network. *Union Network* performs three categories detection together in the same Network as *Single Network*. From Table 2 we can see that:

Table 2. Basic results of Faster RCNN and our DID-Net. **P** denotes person, **H** denotes hand and **F** denote face. The bold values are the best performance in each column

	Train Data	Detector	Person AP	Face AP	Hand AP	mAP
Separate Network	P	ResNet-50-Faster	88.1	-	-	90.2
	F		-	95.9	-	
	H		-	-	86.7	
Union Network	P + H + F	ResNet-50 -Faster	89.3	93.0	82.3	88.2
Our Network	P + H + F	ResNet-50-DID	**89.6**	**96.1**	**87.5**	**91.1**

- **Person AP** The AP performance of person is increased from 88.1 to 89.6 after conducting multi-objects detection task together in a single network, from which we can infer that the saliency of hand and face will contribute to the detection performance of the person. Works of [37,38] also show that the facial contributes based supervision can effectively enhance the capability of a face detection network.
- **Small Parts AP** The AP values of small parts in *Union Nework* decrease compared with the *Separate Network*. The reasons mainly come from the scale variance between human body and body parts. In practice, the whole image is mainly occupied by the big objects like human body. Small hand and face usually occupy a relatively smaller area. Given a fixed anchor scales and anchor ratios, there will be less small anchors generated by RPN, which lead to the poor performance of small parts detection.

Compared with *Single Network* and *Union Network*, DID-Net conducts another RPN inside each body, where less disturbance from other objects or background exist. The network can learn more spacial region relationship between body parts. In addition, when extracting features of the person proposals, we pooled these wrapped features to size 24 × 24, which will be larger than the original person feature maps mostly. That operation will zoom out the features of the small parts and result in a high resolution of feature maps for further detection. Results show that our DID-Net outperforms the *Separate Network* and *Union Network*, which demonstrate the efficiency of our framework.

4.4 Ablation study

In Table 3, we evaluate how different components affect the detection performance of *PartsDetector*. For fair comparison, we fixed the parameters of the backbone network and *BodyDetector*. Thus the person AP in Table 3 is constant.

Table 3. Experiment results about how structure design and the number of training bodies affect the detection performance of *PartsDetector*

PartsDetector	Person RoI	Person AP	Face AP	Hand AP	mAP
RPN-S2 + RCNN-S2	16	89.6	**96.1**	**87.5**	**91.1**
RPN-S2	16	89.6	93.5	83.3	88.8
RPN-S2 + RCNN-S2	8	89.6	95.4	86.0	90.3

- **Architecture.** There are two designs of *PartsDetector*: (1) *RPN-S2 + RCNN-S2* design: It follows the structure we described in Section 3. Two RoI Align operations are adopted as shown in Fig. 2. (2) *RPN-S2* is changed to output the classification and box regression of each RoI directly as SSD did. There is only one RoI Align operation. Table 3 shows that two-stage design of *PartsDetector* can achieve a higher accuracy on small parts detection.
- **Person RoI.** Person RoI represents that how many person bodies after NMS opetation are selected from the *BodyDetector* during training. We use a different number of 8 and 16 for this number and conduct experiments. Larger number is not used considering the computation efficiency. Results show that, if the selected bodies are less than the bodies the image really contains during training, the features of Head-hf will not contain enough response to unselected bodies. And that incomplete training will lead to a performance drop of the *PartsDetector*.

4.5 Comparisons with the state-of-arts

We compare the proposed DID-Net to the state-of-the-art methods performed on *Human-Parts* dataset in Table 4, which includes SSD (training image size are 512 × 512), Faster R-CNN, RFCN [28] with online hard example mining(OHEM) [39] and FPN [40]. SSD is conducted using original settings of paper. For those region-based detectors, multi-scale training (described in 4.2), anchor scales of (64^2, 128^2, 256^2, 512^2), anchor ratios of (0.5, 1, 2) and RoI Align are

Table 4. Table of Average Precision on validation set of Human parts. Our DID-Net outperforms SSD [15], Faster R-CNN [20], RFCN [28] with OHEM [39] and FPN [40] on person and face detection. Comparable results are achieved on face detection ability compared with FPN

	backbone	Multi scale	RoI Align	AP_{person}	AP_{face}	AP_{hand}	mAP
SSD	VGG16	-	-	84.3	90.4	77.4	84.0
Faster R-CNN	ResNet-50	✓	✓	89.2	93.0	82.3	88.1
RFCN + ohem	ResNet-50	✓	✓	88.9	93.2	84.5	88.9
FPN	ResNet-50	✓	✓	87.9	**96.5**	85.4	89.9
DID-Net	ResNet-50	✓	✓	**89.6**	96.1	**87.5**	**91.1**

adopted in all models during training. All the models are trained for 30,000 iterations with a initial learning rate of 0.01, a weight decay of 0.0005 and a momentum of 0.9.

Results show that our DID-Net achieves the highest accuracy on person and hand category. Region-based two-stage detection models (Faster R-CNN, RFCN, FPN) outperform single-stage detectors (SSD), which demonstrates the ability of two-stage detectors. FPN achieves better performance on hand and face detection performance than Faster R-CNN owing to the efficient fusion of features with different scales. While the performance of person dropped slightly than Faster R-CNN due to the scale variance between human body and body parts. After adding the *PartsDetector* to the basic *BodyDetector*, the detection performance of face and hand in our architecture have a large promotion.

4.6 Qualitative Results

We show some qualitative human parts detection results on sample images in Fig. 4. From the Faster RCNN results in the first row, we observe that there still exists some small hands or faces (in yellow boxes), which can not be detected. While in the results of DID-Net, those hard parts can be found. Results can be seen in the second row of Fig. 4.

5 Conclusion

In this paper, we propose a Detector-in-Detector network (DID-Net) for the multi-level objects by simply appending a light-weight *PartsDetector* after Faster RCNN structure to perform parts detection inside each body. We implement our methods on *Human-Parts* dataset and experiments show that our methods can achieve an excellent performance, especially for small hands and faces. The novel framework we constructed can also be performed on other multi-level objects. In addition, we have also build a dataset named *Human-Parts*, which aims to the human body, hand, and face in many realistic environments. Others can also use our dataset for related tasks and applications.

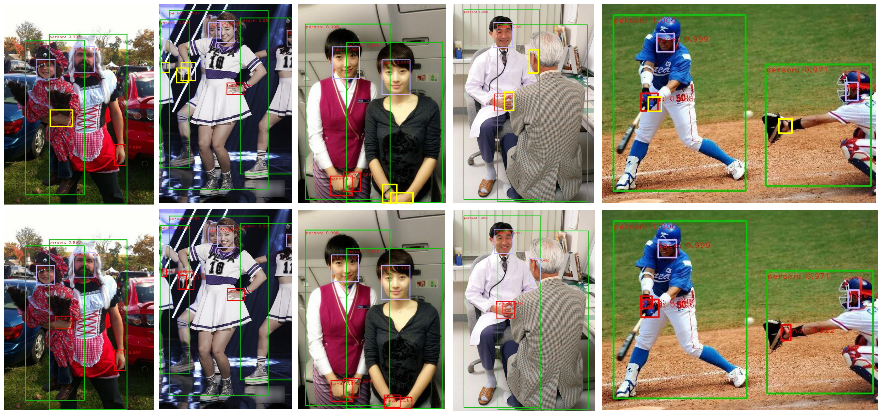

Fig. 4. Some example results of Faster RCNN (Top) and the proposed DID-Net (Bottom). Green boxes are the detected persons. Red boxes are the detected hands. Purple boxes are the detected faces. Yellow boxes in the first row are the missed objects of Faster-RCNN.

References

1. Ribeiro, D., Mateus, A., Nascimento, J.C., Miraldo, P.: A real-time pedestrian detector using deep learning for human-aware navigation. (2016)
2. Chen, Y., Wang, Z., Peng, Y., Zhang, Z., Yu, G., Sun, J.: Cascaded pyramid network for multi-person pose estimation. (2017)
3. Papandreou, G., Zhu, T., Kanazawa, N., Toshev, A., Tompson, J., Bregler, C., Murphy, K.: Towards accurate multi-person pose estimation in the wild. (2017) 3711–3719
4. Xiao, S., Feng, J., Liu, L., Nie, X., Wang, W., Yan, S., Kassim, A.: Recurrent 3d-2d dual learning for large-pose facial landmark detection. In: IEEE International Conference on Computer Vision. (2017) 1642–1651
5. Dhawan, A., Honrao, V.: Implementation of hand detection based techniques for human computer interaction. Computer Science **72** (2013) 6–13
6. Ghorban, F., Marn, J., Yu, S., Colombo, A., Kummert, A.: Aggregated channels network for real-time pedestrian detection. (2018)
7. Samangouei, P., Najibi, M., Davis, L., Chellappa, R.: Face-magnet: Magnifying feature maps to detect small faces. (2018)
8. Zhu, C., Zheng, Y., Luu, K., Savvides, M.: Cms-rcnn: Contextual multi-scale region-based cnn for unconstrained face detection. (2017)
9. Deng, X., Yuan, Y., Zhang, Y., Tan, P., Chang, L., Yang, S., Wang, H.: Joint hand detection and rotation estimation by using cnn. IEEE Transactions on Image Processing **27** (2016)
10. Le, T.H.N., Quach, K.G., Zhu, C., Chi, N.D., Luu, K., Savvides, M.: Robust hand detection and classification in vehicles and in the wild. In: Computer Vision and Pattern Recognition Workshops. (2017) 1203–1210
11. Mittal, A., Zisserman, A., Torr, P.: Hand detection using multiple proposals. In: British Machine Vision Conference. (2011) 75.1–75.11

12. Zhao, K., Zhang, W., Jiang, Y.: Semantic interactions in multi-level objects segmentation. In: International Conference on Computational and Information Sciences. (2010) 665–668
13. He, K., Zhang, X., Ren, S., Sun, J.: Deep residual learning for image recognition. In: IEEE Conference on Computer Vision and Pattern Recognition. (2016) 770–778
14. Girshick, R.: Fast r-cnn. Computer Science (2015)
15. Liu, W., Anguelov, D., Erhan, D., Szegedy, C., Reed, S., Fu, C.Y., Berg, A.C.: Ssd: Single shot multibox detector. (2015) 21–37
16. Li, Z., Zhou, F.: Fssd: Feature fusion single shot multibox detector. (2017)
17. Redmon, J., Divvala, S., Girshick, R., Farhadi, A.: You only look once: Unified, real-time object detection. (2015) 779–788
18. Jiang, H., Learnedmiller, E.: Face detection with the faster r-cnn. (2016) 650–657
19. He, K., Fu, Y., Xue, X.: A jointly learned deep architecture for facial attribute analysis and face detection in the wild. (2017)
20. Ren, S., He, K., Girshick, R., Sun, J.: Faster r-cnn: towards real-time object detection with region proposal networks. In: International Conference on Neural Information Processing Systems. (2015) 91–99
21. Hu, P., Ramanan, D.: Finding tiny faces. (2016)
22. Najibi, M., Samangouei, P., Chellappa, R., Davis, L.S.: Ssh: Single stage headless face detector. (2017) 4885–4894
23. Li, H., Lin, Z., Shen, X., Brandt, J., Hua, G.: A convolutional neural network cascade for face detection. In: Computer Vision and Pattern Recognition. (2015) 5325–5334
24. Zhang, K., Zhang, Z., Li, Z., Qiao, Y.: Joint face detection and alignment using multitask cascaded convolutional networks. IEEE Signal Processing Letters **23** (2016) 1499–1503
25. Lin, T.Y., Maire, M., Belongie, S., Hays, J., Perona, P., Ramanan, D., Dollr, P., Zitnick, C.L.: Microsoft coco: Common objects in context. **8693** (2014) 740–755
26. He, K., Gkioxari, G., Dollr, P., Girshick, R.: Mask r-cnn. (2017)
27. Deng, J., Dong, W., Socher, R., Li, L.J., Li, K., Li, F.F.: Imagenet: A large-scale hierarchical image database. In: Computer Vision and Pattern Recognition, 2009. CVPR 2009. IEEE Conference on. (2009) 248–255
28. Dai, J., Li, Y., He, K., Sun, J.: R-fcn: Object detection via region-based fully convolutional networks. (2016)
29. Girshick, R., Iandola, F., Darrell, T., Malik, J.: Deformable part models are convolutional neural networks. (2014) 437–446
30. Dollar, P., Wojek, C., Schiele, B., Perona, P.: Pedestrian detection : A benchmark. Proc.conf.on Computer Vision Pattern Recognition (2009) 304–311
31. Zhang, S., Benenson, R., Schiele, B.: Citypersons: A diverse dataset for pedestrian detection. (2017)
32. Huang, S., Wang, W., He, S., Lau, R.W.H.: Egocentric hand detection via dynamic region growing. Acm Transactions on Multimedia Computing Communications Applications **14** (2017)
33. Jain, V., Learned-Miller, E.: FDDB: A Benchmark for Face Detection in Unconstrained Settings. (2010)
34. Yang, S., Luo, P., Chen, C.L., Tang, X.: Wider face: A face detection benchmark. In: IEEE Conference on Computer Vision and Pattern Recognition. (2016) 5525–5533
35. Wu, J., Zheng, H., Zhao, B., Li, Y., Yan, B., Liang, R., Wang, W., Zhou, S., Lin, G., Fu, Y.: Ai challenger : A large-scale dataset for going deeper in image understanding. (2017)

36. Everingham, M., Gool, L., Williams, C.K., Winn, J., Zisserman, A.: The pascal visual object classes (voc) challenge. International Journal of Computer Vision **88** (2010) 303–338
37. Qin, H., Yan, J., Li, X., Hu, X.: Joint training of cascaded cnn for face detection. In: IEEE Conference on Computer Vision and Pattern Recognition. (2016) 3456–3465
38. Yang, S., Luo, P., Loy, C.C., Tang, X.: Faceness-net: Face detection through deep facial part responses. IEEE Trans Pattern Anal Mach Intell **PP** (2017) 1–1
39. Shrivastava, A., Gupta, A., Girshick, R.: Training region-based object detectors with online hard example mining. (2016) 761–769
40. Lin, T.Y., Dollar, P., Girshick, R., He, K., Hariharan, B., Belongie, S.: Feature pyramid networks for object detection. (2016) 936–944