High-Resolution Representations for Labeling Pixels and Regions

Ke Sun[1,2*] Yang Zhao[3*] Borui Jiang[2,4*] Tianheng Cheng[2,5*] Bin Xiao[2]
Dong Liu[1] Yadong Mu[4] Xinggang Wang[5] Wenyu Liu[5] Jingdong Wang[2†]
[1]University of Science and Technology of China   [2]Microsoft Research Asia
[3]The University of Adelaide [4]Peking University [5]Huazhong University of Science and Technology
sunk@mail.ustc.edu.cn, yang.zhao4@griffithuni.edu.au, jbr@pku.edu.cn, muyadong@gmail.com
{vic,liuwy,xgwang}@hust.edu.cn, dongleiu@ustc.edu.cn, {Bin.Xiao,jingdw}@microsoft.com

Abstract

High-resolution representation learning plays an essential role in many vision problems, e.g., pose estimation and semantic segmentation. The high-resolution network (HRNet) [91], recently developed for human pose estimation, maintains high-resolution representations through the whole process by connecting high-to-low resolution convolutions in parallel *and produces strong high-resolution representations by repeatedly conducting fusions across parallel convolutions.*

In this paper, we conduct a further study on high-resolution representations by introducing a simple yet effective modification and apply it to a wide range of vision tasks. We augment the high-resolution representation by aggregating the (upsampled) representations from all the parallel convolutions rather than only the representation from the high-resolution convolution as done in [91]. This simple modification leads to stronger representations, evidenced by superior results. We show top results in semantic segmentation on Cityscapes, LIP, and PASCAL Context, and facial landmark detection on AFLW, COFW, 300W, and WFLW. In addition, we build a multi-level representation from the high-resolution representation and apply it to the Faster R-CNN object detection framework and the extended frameworks. The proposed approach achieves superior results to existing single-model networks on COCO object detection. The code and models have been publicly available at https://github.com/HRNet.

1. Introduction

Deeply-learned representations have been demonstrated to be strong and achieved state-of-the-art results in many vision tasks. There are two main kinds of representations:

low-resolution representations that are mainly for image classification, and high-resolution representations that are essential for many other vision problems, e.g., semantic segmentation, object detection, human pose estimation, etc. The latter one, the interest of this paper, remains unsolved and is attracting a lot of attention.

There are two main lines for computing high-resolution representations. One is to recover high-resolution representations from low-resolution representations outputted by a network (e.g., ResNet) and optionally intermediate medium-resolution representations, e.g., Hourglass [72], SegNet [2], DeconvNet [74], U-Net [83], and encoder-decoder [77]. The other one is to maintain high-resolution representations through high-resolution convolutions and strengthen the representations with parallel low-resolution convolutions [91, 30, 132, 86]. In addition, dilated convolutions are used to replace some strided convolutions and associated regular convolutions in classification networks to compute medium-resolution representations [13, 126].

We go along the research line of maintaining high-resolution representations and further study the high-resolution network (HRNet), which is initially developed for human pose estimation [91], for a broad range of vision tasks. An HRNet maintains high-resolution representations by connecting high-to-low resolution convolutions in parallel and repeatedly conducting multi-scale fusions across parallel convolutions. The resulting high-resolution representations are not only strong but also spatially precise.

We make a simple modification by exploring the representations from all the high-to-low resolution parallel convolutions other than only the high-resolution representations in the original HRNet [91]. This modification adds a small overhead and leads to stronger high-resolution representations. The resulting network is named as HRNetV2. We empirically show the superiority to the original HRNet.

We apply our proposed network to semantic segmentation/facial landmark detection through estimating segmentation maps/facial landmark heatmaps from the output high-

---

*Equal contribution.
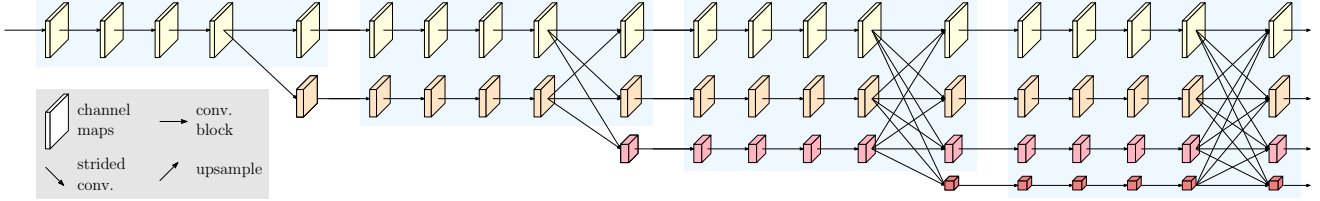†Corresponding author, welleast@outlook.com

Figure 1. A simple example of a high-resolution network. There are four stages. The 1st stage consists of high-resolution convolutions. The 2nd (3rd, 4th) stage repeats two-resolution (three-resolution, four-resolution) blocks. The detail is given in Section 3.

resolution representations. In semantic segmentation, the proposed approach achieves state-of-the-art results on PASCAL Context, Cityscapes, and LIP with similar model sizes and lower computation complexity. In facial landmark detection, our approach achieves overall best results on four standard datasets: AFLW, COFW, 300W, and WFLW.

In addition, we construct a multi-level representation from the high-resolution representation, and apply it to the Faster R-CNN object detection framework and its extended frameworks, Mask R-CNN [38] and Cascade R-CNN [9]. The results show that our method gets great detection performance improvement and in particular dramatic improvement for small objects. With single-scale training and testing, the proposed approach achieves better COCO object detection results than existing single-model methods.

2. Related Work

Strong high-resolution representations play an essential role in pixel and region labeling problems, e.g., semantic segmentation, human pose estimation, facial landmark detection, and object detection. We review representation learning techniques developed mainly in the semantic segmentation, facial landmark detection [92, 50, 69, 104, 123, 94, 119] and object detection areas[1], from low-resolution representation learning, high-resolution representation recovering, to high-resolution representation maintaining.

Learning low-resolution representations. The fully-convolutional network (FCN) approaches [67, 87] compute low-resolution representations by removing the fully-connected layers in a classification network, and estimate from their coarse segmentation confidence maps. The estimated segmentation maps are improved by combining the fine segmentation score maps estimated from intermediate low-level medium-resolution representations [67], or iterating the processes [50]. Similar techniques have also been applied to edge detection, e.g., holistic edge detection [106].

The fully convolutional network is extended, by replacing a few (typically two) strided convolutions and the associated convolutions with dilated convolutions, to the dilation version, leading to medium-resolution representations [126, 13, 115, 12, 57]. The representations are further

augmented to multi-scale contextual representations [126, 13, 15] through feature pyramids for segmenting objects at multiple scales.

Recovering high-resolution representations. An upsample subnetwork, like a decoder, is adopted to gradually recover the high-resolution representations from the low-resolution representations outputted by the downsample process. The upsample subnetwork could be a symmetric version of the downsample subnetwork, with skipping connection over some mirrored layers to transform the pooling indices, e.g., SegNet [2] and DeconvNet [74], or copying the feature maps, e.g., U-Net [83] and Hourglass [72, 111, 7, 22, 6], encoder-decoder [77], FPN [62], and so on. The full-resolution residual network [78] introduces an extra full-resolution stream that carries information at the full image resolution, to replace the skip connections, and each unit in the downsample and upsample subnetworks receives information from and sends information to the full-resolution stream.

The asymmetric upsample process is also widely studied. RefineNet [60] improves the combination of upsampled representations and the representations of the same resolution copied from the downsample process. Other works include: light upsample process [5]; light downsample and heavy upsample processes [97], recombinator networks [40]; improving skip connections with more or complicated convolutional units [76, 125, 42], as well as sending information from low-resolution skip connections to high-resolution skip connections [133] or exchanging information between them [36]; studying the details the upsample process [100]; combining multi-scale pyramid representations [16, 105]; stacking multiple DeconvNets/U-Nets/Hourglass [31, 101] with dense connections [93].

Maintaining high-resolution representations. High-resolution representations are maintained through the whole process, typically by a network that is formed by connecting multi-resolution (from high-resolution to low-resolution) parallel convolutions with repeated information exchange across parallel convolutions. Representative works include GridNet [30], convolutional neural fabrics [86], interlinked CNNs [132], and the recently-developed high-resolution networks (HRNet) [91] that is our interest.

The two early works, convolutional neural fabrics [86]

---

[1]The techniques developed for human pose estimation are reviewed in [91].
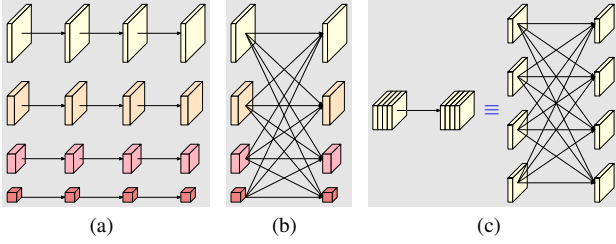
Figure 2. Multi-resolution block: (a) multi-resolution group convolution and (b) multi-resolution convolution. (c) A normal convolution (left) is equivalent to fully-connected multi-branch convolutions (right).

and interlinked CNNs [132], lack careful design on when to start low-resolution parallel streams and how and when to exchange information across parallel streams, and do not use batch normalization and residual connections, thus not showing satisfactory performance.

GridNet [30] is like a combination of multiple U-Nets and includes two symmetric information exchange stages: the first stage only passes information from high-resolution to low-resolution, and the second stage only passes information from low-resolution to high-resolution. This limits its segmentation quality.

3. Learning High-Resolution Representations

The high-resolution network [91], which we named HRNetV1 for convenience, maintains high-resolution representations by connecting high-to-low resolution convolutions in parallel, where there are repeated multi-scale fusions across parallel convolutions.

Architecture. The architecture is illustrated in Figure 1. There are four stages, and the 2nd, 3rd and 4th stages are formed by repeating modularized multi-resolution blocks. A multi-resolution block consists of a multi-resolution group convolution and a multi-resolution convolution which is illustrated in Figure 2 (a) and (b). The multi-resolution group convolution is a simple extension of the group convolution, which divides the input channels into several subsets of channels and performs a regular convolution over each subset over different spatial resolutions separately.

The multi-resolution convolution is depicted in Figure 2 (b). It resembles the multi-branch full-connection manner of the regular convolution, illustrated in in Figure 2 (c). A regular convolution can be divided as multiple small convolutions as explained in [122]. The input channels are divided into several subsets, and the output channels are also divided into several subsets. The input and output subsets are connected in a fully-connected fashion, and each connection is a regular convolution. Each subset of output channels is a summation of the outputs of the convolutions over each subset of input channels.

The differences lie in two-fold. (i) In a multi-resolution

convolution each subset of channels is over a different resolution. (ii) The connection between input channels and output channels needs to handle The resolution decrease is implemented in [91] by using several 2-strided 3×3 convolutions. The resolution increase is simply implemented in [91] by bilinear (nearest neighbor) upsampling.

Modification. In the original approach HRNetV1, only the representation (feature maps) from the high-resolution convolutions in [91] are outputted, which is illustrated in Figure 3 (a). This means that only a subset of output channels from the high-resolution convolutions is exploited and other subsets from low-resolution convolutions are lost.

We make a simple yet effective modification by exploiting other subsets of channels outputted from low-resolution convolutions. The benefit is that the capacity of the multi-resolution convolution is fully explored. This modification only adds a small parameter and computation overhead.

We rescale the low-resolution representations through bilinear upsampling to the high resolution, and concatenate the subsets of representations, illustrated in Figure 3 (b), resulting in the high-resolution representation, which we adopt for estimating segmentation maps/facial landmark heatmaps. In application to object detection, we construct a multi-level representation by downsampling the high-resolution representation with average pooling to multiple levels, which is depicted in Figure 3 (c). We name the two modifications as HRNetV2 and HRNetV2p, respectively, and empirically compare them in Section 4.4.

Instantiation We instantiate the network using a similar manner as HRNetV1 [91][2]. The network starts from a stem that consists of two strided 3×3 convolutions decreasing the resolution to $1/4$. The 1st stage contains 4 residual units where each unit is formed by a bottleneck with the width 64, and is followed by one 3×3 convolution reducing the width of feature maps to C. The 2nd, 3rd, 4th stages contain 1, 4, 3 multi-resolution blocks, respectively. The widths (number of channels) of the convolutions of the four resolutions are C, $2C$, $4C$, and $8C$, respectively. Each branch in the multi-resolution group convolution contains 4 residual units and each unit contains two 3×3 convolutions in each resolution.

In applications to semantic segmentation and facial landmark detection, we mix the output representations (Figure 3 (b)), from all the four resolutions through a 1×1 convolution, and produce a $15C$-dimensional representation. Then, we pass the mixed representation at each position to a linear classifier/regressor with the softmax/MSE loss to predict the segmentation maps/facial landmark heatmaps. For semantic segmentation, the segmentation maps are upsampled (4 times) to the input size by bilinear upsampling for both training and testing. In application to object detection,

---

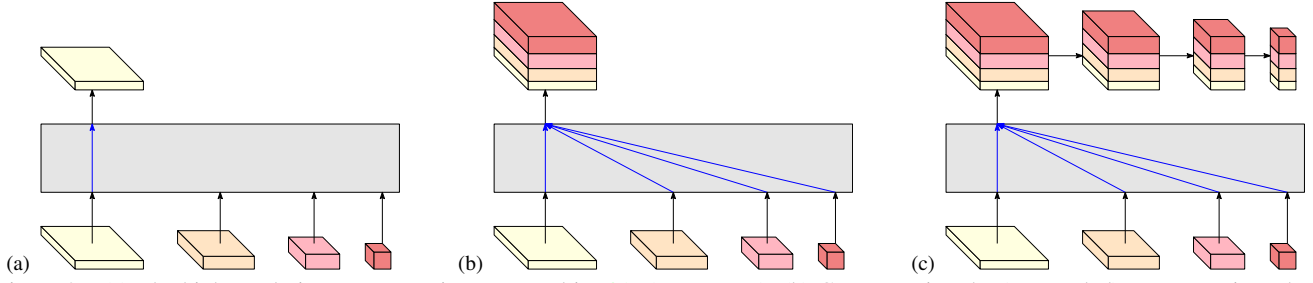[2]https://github.com/leoxiaobin/deep-high-resolution-net.pytorch

Figure 3. (a) The high-resolution representation proposed in [91] (HRNetV1); (b) Concatenating the (upsampled) representations that are from all the resolutions for semantic segmentation and facial landmark detection (HRNetV2); (c) A feature pyramid formed over (b) for object detection (HRNetV2p). The four-resolution representations at the bottom in each sub-figure are outputted from the network in Figure 1, and the gray box indicates how the output representation is obtained from the input four-resolution representations.

Table 1. Segmentation results on Cityscapes val (single scale and no flipping). The GFLOPs is calculated on the input size 1024×2048.

	backbone	#param.	GFLOPs	mIoU
UNet++ [133]	ResNet-101	59.5M	748.5	75.5
DeepLabv3 [14]	Dilated-ResNet-101	58.0M	1778.7	78.5
DeepLabv3+ [16]	Dilated-Xception-71	43.5M	1444.6	79.6
PSPNet [126]	Dilated-ResNet-101	65.9M	2017.6	79.7
Our approach	HRNetV2-W40	45.2M	493.2	80.2
Our approach	HRNetV2-W48	65.9M	747.3	**81.1**

we reduce the dimension of the high-resolution representation to 256, similar to FPN [62], through a 1×1 convolution before forming the feature pyramid in Figure 3 (c).

4. Experiments

4.1. Semantic Segmentation

Semantic segmentation is a problem of assigning a class label to each pixel. We report the results over two scene parsing datasets, PASCAL Context [71] and Cityscapes [19], and a human parsing dataset, LIP [34]. The mean of class-wise intersection over union (mIoU) is adopted as the evaluation metric.

Cityscapes. The Cityscapes dataset [19] contains $5,000$ high quality pixel-level finely annotated scene images. The finely-annotated images are divided into $2,975/500/1,525$ images for training, validation and testing. There are 30 classes, and 19 classes among them are used for evaluation. In addition to the mean of class-wise intersection over union (mIoU), we report other three scores on the test set: IoU category (cat.), iIoU class (cla.) and iIoU category (cat.).

We follow the same training protocol [126, 127]. The data are augmented by random cropping (from 1024×2048 to 512×1024), random scaling in the range of $[0.5, 2]$, and random horizontal flipping. We use the SGD optimizer with the base learning rate of 0.01, the momentum of 0.9 and the weight decay of 0.0005. The poly learning rate policy with the power of 0.9 is used for dropping the learning rate. All the models are trained for $120K$ iterations with the batch size of 12 on 4 GPUs and syncBN.

Table 1 provides the comparison with several representative methods on the Cityscapes validation set in

Table 2. Semantic segmentation results on Cityscapes test.

	backbone	mIoU	iIoU cla.	IoU cat.	iIoU cat.
Model learned on the train *set*					
PSPNet [126]	Dilated-ResNet-101	78.4	56.7	90.6	78.6
PSANet [127]	Dilated-ResNet-101	78.6	-	-	-
PAN [54]	Dilated-ResNet-101	78.6	-	-	-
AAF [45]	Dilated-ResNet-101	79.1	-	-	-
Our approach	HRNetV2-W48	**80.4**	**59.2**	**91.5**	**80.8**
Model learned on the train+valid *set*					
GridNet [30]	-	69.5	44.1	87.9	71.1
LRR-4x [33]	-	69.7	48.0	88.2	74.7
DeepLab [13]	Dilated-ResNet-101	70.4	42.6	86.4	67.7
LC [55]	-	71.1	-	-	-
Piecewise [61]	VGG-16	71.6	51.7	87.3	74.1
FRRN [78]	-	71.8	45.5	88.9	75.1
RefineNet [60]	ResNet-101	73.6	47.2	87.9	70.6
PEARL [43]	Dilated-ResNet-101	75.4	51.6	89.2	75.1
DSSPN [59]	Dilated-ResNet-101	76.6	56.2	89.6	77.8
LKM [76]	ResNet-152	76.9	-	-	-
DUC-HDC [99]	-	77.6	53.6	90.1	75.2
SAC [120]	Dilated-ResNet-101	78.1	-	-	-
DepthSeg [47]	Dilated-ResNet-101	78.2	-	-	-
ResNet38 [103]	WResNet-38	78.4	59.1	90.9	78.1
BiSeNet [113]	ResNet-101	78.9	-	-	-
DFN [114]	ResNet-101	79.3	-	-	-
PSANet [127]	Dilated-ResNet-101	80.1	-	-	-
PADNet [108]	Dilated-ResNet-101	80.3	58.8	90.8	78.5
DenseASPP [126]	WDenseNet-161	80.6	59.1	90.9	78.1
Our approach	HRNetV2-W48	**81.6**	**61.8**	**92.1**	**82.2**

terms of parameter and computation complexity and mIoU class. (i) HRNetV2-W40 (40 indicates the width of the high-resolution convolution), with similar model size to DeepLabv3+ and much lower computation complexity, gets better performance: 4.7 points gain over UNet++, 1.7 points gain over DeepLabv3 and about 0.5 points gain over PSPNet, DeepLabv3+. (ii) HRNetV2-W48, with similar model size to PSPNet and much lower computation complexity, achieves much significant improvement: 5.6 points gain over UNet++, 2.6 points gain over DeepLabv3 and about 1.4 points gain over PSPNet, DeepLabv3+. In the following comparisons, we adopt HRNetV2-W48 that is pretrained

Table 3. Semantic segmentation results on PASCAL-context. The methods are evaluated on 59 classes and 60 classes.

	backbone	mIoU (59 classes)	mIoU (60 classes)
FCN-8s [88]	VGG-16	-	35.1
BoxSup [20]	-	-	40.5
HO_CRF [1]	-	-	41.3
Piecewise [61]	VGG-16	-	43.3
DeepLab-v2 [13]	Dilated-ResNet-101	-	45.7
RefineNet [60]	ResNet-152	-	47.3
UNet++ [133]	ResNet-101	47.7	-
PSPNet [126]	Dilated-ResNet-101	47.8	-
Ding et al. [23]	ResNet-101	51.6	-
EncNet [117]	Dilated-ResNet-101	52.6	-
Our approach	HRNetV2-W48	**54.0**	**48.3**

Table 4. Semantic segmentation results on LIP. Our method doesn't exploit any extra information, e.g., pose or edge.

	backbone	extra.	pixel acc.	avg. acc.	mIoU
Attention+SSL [34]	VGG16	Pose	84.36	54.94	44.73
DeepLabV3+ [16]	Dilated-ResNet-101	-	84.09	55.62	44.80
MMAN [68]	Dilated-ResNet-101	-	-	-	46.81
SS-NAN [128]	ResNet-101	Pose	87.59	56.03	47.92
MuLA [73]	Hourglass	Pose	88.50	60.50	49.30
JPPNet [58]	Dilated-ResNet-101	Pose	86.39	62.32	51.37
CE2P [66]	Dilated-ResNet-101	Edge	87.37	63.20	53.10
Our approach	HRNetV2-W48	N	**88.21**	**67.43**	**55.90**

on ImageNet [3] and has similar model size as most Dilated-ResNet-101 based methods.

Table 2 provides the comparison of our method with state-of-the-art methods on the Cityscapes test set. All the results are with six scales and flipping. Two cases w/o using coarse data are evaluated: One is about the model learned on the `train` set, and the other is about the model learned on the `train+valid` set. In both cases, HRNetV2-W48 achieves the best performance and outperforms the previous state-of-the-art by 1 point.

PASCAL context. The PASCAL context dataset [71] includes $4,998$ scene images for training and $5,105$ images for testing with 59 semantic labels and 1 background label.

The data augmentation and learning rate policy are the same as Cityscapes. Following the widely-used training strategy [117, 23], we resize the images to 480×480 and set the initial learning rate to 0.004 and weight decay to 0.0001. The batch size is 16 and the number of iterations is $60K$.

We follow the standard testing procedure [117, 23]. The image is resized to 480×480 and then fed into our network. The resulting 480×480 label maps are then resized to the original image size. We evaluate the performance of our approach and other approaches using six scales and flipping.

Table 3 provides the comparison of our method with state-of-the-art methods. There are two kinds of evaluation schemes: mIoU over 59 classes and 60 classes (59 classes + background). In both cases, HRNetV2-W48 performs su-

Table 5. GFLOPs and #parameters of Faster R-CNN for COCO object detection. The numbers are obtained with the input size 800×1200 and 512 proposals fed into R-CNN. ResNet-x-FPN (R-x), X-101-64×4d (X-101), HRNetV2p-Wx (H-x).

	R-50	H-18	R-101	H-32	R-152	H-40	X-101	H-48
#param. (M)	39.8	26.2	57.8	45.0	72.7	60.5	94.9	79.4
GFLOPs	172.3	159.1	239.4	245.3	306.4	314.9	381.8	399.1

Table 6. Object detection results evaluated on COCO `val` in the Faster R-CNN framework. LS = learning schedule.

backbone	LS	AP	AP_{50}	AP_{75}	AP_S	AP_M	AP_L
ResNet-50-FPN	1×	36.7	58.3	39.9	20.9	39.8	47.9
HRNetV2p-W18	1×	36.2	57.3	39.3	20.7	39.0	46.8
ResNet-50-FPN	2×	37.6	58.7	41.3	21.4	**40.8**	**49.7**
HRNetV2p-W18	2×	38.0	58.9	41.5	22.6	**40.8**	49.6
ResNet-101-FPN	1×	39.2	61.1	43.0	22.3	42.9	50.9
HRNetV2p-W32	1×	39.6	61.0	43.3	23.7	42.5	50.5
ResNet-101-FPN	2×	39.8	61.4	43.4	22.9	43.6	52.4
HRNetV2p-W32	2×	40.9	61.8	44.8	24.4	43.7	53.3
ResNet-152-FPN	1×	39.5	61.2	43.0	22.1	43.3	51.8
HRNetV2p-W40	1×	40.4	61.8	44.1	23.8	43.8	52.3
ResNet-152-FPN	2×	40.6	61.9	44.5	22.8	44.0	53.1
HRNetV2p-W40	2×	41.6	62.5	45.6	23.8	44.9	53.8
X-101-64×4d-FPN	1×	41.3	**63.4**	45.2	24.5	**45.8**	53.3
HRNetV2p-W48	1×	41.3	62.8	45.1	**25.1**	44.5	52.9
X-101-64×4d-FPN	2×	40.8	62.1	44.6	23.2	44.5	53.7
HRNetV2p-W48	2×	**41.8**	62.8	**45.9**	25.0	44.7	**54.6**

perior to previous state-of-the-arts.

LIP. The LIP dataset [34] contains $50,462$ elaborately annotated human images, which are divided into $30,462$ training images, and $10,000$ validation images. The methods are evaluated on 20 categories (19 human part labels and 1 background label). Following the standard training and testing settings [66], the images are resized to 473×473 and the performance is evaluated on the average of the segmentation maps of the original and flipped images.

The data augmentation and learning rate policy are the same as Cityscapes. The training strategy follows the recent setting [66]. We set the initial learning rate to 0.007 and the momentum to 0.9 and the weight decay to 0.0005. The batch size is 40 and the number of iterations is $110K$.

Table 4 provides the comparison of our method with state-of-the-art methods. The overall performance of HRNetV2-W48 performs the best with fewer parameters and lighter computation cost. We also would like to mention that our networks do not use extra information such as pose or edge.

4.2. COCO Object Detection

We apply our multi-level representations (HRNetV2p)[4], shown in Figure 3 (c), in the Faster R-CNN [82] and Mask R-CNN [38] frameworks. We perform the evaluation on the MS-COCO 2017 detection dataset, which contains $\sim 118k$

---

[3]The description about ImageNet pretraining is given in the Appendix.

[4]Same as FPN [63], we also use 5 levels.

Table 7. Object detection results evaluated on COCO `val` in the Mask R-CNN framework. LS = learning schedule.

backbone	LS	mask				bbox			
		AP	AP_S	AP_M	AP_L	AP	AP_S	AP_M	AP_L
ResNet-50-FPN	1×	34.2	15.7	36.8	50.2	37.8	22.1	40.9	49.3
HRNetV2p-W18	1×	33.8	15.6	35.6	49.8	37.1	21.9	39.5	47.9
ResNet-50-FPN	2×	35.0	16.0	**37.5**	**52.0**	38.6	21.7	41.6	50.9
HRNetV2p-W18	2×	**35.3**	**16.9**	37.5	51.8	**39.2**	**23.7**	**41.7**	**51.0**
ResNet-101-FPN	1×	36.1	16.2	39.0	53.0	40.0	22.6	43.4	52.3
HRNetV2p-W32	1×	36.7	17.3	39.0	53.0	40.9	24.5	43.9	52.2
ResNet-101-FPN	2×	36.7	17.0	39.5	54.8	41.0	23.4	44.4	53.9
HRNetV2p-W32	2×	**37.6**	**17.8**	**40.0**	**55.0**	**42.3**	**25.0**	**45.4**	**54.9**

images for training, 5k for validation (`val`) and $\sim$ 20k testing without provided annotations (`test-dev`). The standard COCO-style evaluation is adopted.

We train the models for both our HRNetV2p and the ResNet on the public mmdetection platform [11] with the provided training setup, except that we use the learning rate schedule suggested in [37] for 2×. The data is augmented by standard horizontal flipping. The input images are resized such that the shorter edge is 800 pixels [62]. Inference is performed on a single image scale.

Table 5 summarizes #parameters and GFLOPs. Table 6 and Table 7 report the detection results on COCO `val`. There are several observations. (i) The model size and computation complexity of HRNetV2p-W18 (HRNetV2p-W32) are smaller than ResNet-50-FPN (ResNet-101-FPN). (ii) With 1×, HRNetV2p-W32 performs better than ResNet-101-FPN. HRNetV2p-W18 performs worse than ResNet-50-FPN, which might come from insufficient optimization iterations. (iii) With 2×, HRNetV2p-W18 and HRNetV2p-W32 perform better than ResNet-50-FPN and ResNet-101-FPN, respectively.

Table 8 reports the comparison of our network to state-of-the-art single-model object detectors on COCO `test-dev` without using multi-scale training and multi-scale testing that are done in [65, 79, 56, 90, 89, 75]. In the Faster R-CNN framework, our networks perform better than ResNets with similar parameter and computation complexity: HRNetV2p-W32 vs. ResNet-101-FPN, HRNetV2p-W40 vs. ResNet-152-FPN, HRNetV2p-W48 vs. X-101-64×4d-FPN. In the Cascade R-CNN framework, our HRNetV2p-W32 performs better.

4.3. Facial Landmark Detection

Facial landmark detection a.k.a. face alignment is a problem of detecting the keypoints from a face image. We perform the evaluation over four standard datasets: WFLW [101], AFLW [49], COFW [8], and 300W [85]. We mainly use the normalized mean error (NME) for evaluation. We use the inter-ocular distance as normalization for WFLW, COFW, and 300W, and the face bounding box as normalization for AFLW. We also report area-under-the-curve scores (AUC) and failure rates.

We follow the standard scheme [101] for training. All the faces are cropped by the provided boxes according to the center location and resized to 256×256. We augment the data by ± 30 degrees in-plane rotation, $0.75 - 1.25$ scaling, and randomly flipping. The base learning rate is 0.0001 and is dropped to 0.00001 and 0.000001 at the 30th and 50th epochs. The models are trained for 60 epochs with the batch size of 16 on one GPU. Different from semantic segmentation, the heatmaps are not upsampled from $1/4$ to the input size, and the loss function is optimized over the $1/4$ maps.

At testing, each keypoint location is predicted by transforming the highest heatvalue location from $1/4$ to the original image space and adjusting it with a quarter offset in the direction from the highest response to the second highest response [17].

We adopt HRNetV2-W18 for face landmark detection whose parameter and computation cost are similar to or smaller than models with widely-used backbones: ResNet-50 and Hourglass [72]. HRNetV2-W18: #parameters $=$ 9.3M, GFLOPs $=$ 4.3G; ResNet-50: #parameters $=$ 25.0M, GFLOPs $=$ 3.8G; Hourglass: #parameters $=$ 25.1M, GFLOPs $=$ 19.1G. The numbers are obtained on the input size 256×256. It should be noted that the facial landmark detection methods adopting ResNet-50 and Hourglass as backbones introduce extra parameter and computation overhead.

WFLW. The WFLW dataset [101] is a recently-built dataset based on the WIDER Face [112]. There are $7,500$ training and $2,500$ testing images with 98 manual annotated landmarks. We report the results on the test set and several subsets: large pose (326 images), expression (314 images), illumination (698 images), make-up (206 images), occlusion (736 images) and blur (773 images).

Table 9 provides the comparison of our method with state-of-the-art methods. Our approach is significantly better than other methods on the test set and all the subsets, including LAB that exploits extra boundary information [101] and PDB that uses stronger data augmentation [28].

AFLW. The AFLW [49] dataset is a widely used benchmark dataset, where each image has 19 facial landmarks. Following [134, 101], we train our models on $20,000$ training images, and report the results on the AFLW-Full set ($4,386$ testing images) and the AFLW-Frontal set (1314 testing images selected from 4386 testing images).

Table 10 provides the comparison of our method with state-of-the-art methods. Our approach achieves the best performance among methods without extra information and stronger data augmentation and even outperforms DCFE with extra 3D information. Our approach performs slightly worse than LAB that uses extra boundary information [101] and PDB [28] that uses stronger data augmentation.

Table 8. Comparison with the state-of-the-art single-model object detectors on COCO `test-dev` without mutli-scale training and testing. We obtain the results of Faster R-CNN and Cascade R-CNN by using our implementations publicly available from the mmdetection platform[11] except that * is from the original paper [9].

	backbone	size	LS	AP	AP_{50}	AP_{75}	AP_S	AP_M	AP_L
MLKP [98]	VGG16	-	-	28.6	52.4	31.6	10.8	33.4	45.1
STDN [131]	DenseNet-169	513	-	31.8	51.0	33.6	14.4	36.1	43.4
DES [124]	VGG16	512	-	32.8	53.2	34.6	13.9	36.0	47.6
CoupleNet [137]	ResNet-101	-	-	33.1	53.5	35.4	11.6	36.3	50.1
DeNet [95]	ResNet-101	512	-	33.8	53.4	36.1	12.3	36.1	50.8
RFBNet [64]	VGG16	512	-	34.4	55.7	36.4	17.6	37.0	47.6
DFPR [48]	ResNet-101	512	1×	34.6	54.3	37.3	-	-	-
PFPNet [46]	VGG16	512	-	35.2	57.6	37.9	18.7	38.6	45.9
RefineDet[121]	ResNet-101	512	-	36.4	57.5	39.5	16.6	39.9	51.4
Relation Net [41]	ResNet-101	600	-	39.0	58.6	42.9	-	-	-
C-FRCNN [18]	ResNet-101	800	1×	39.0	59.7	42.8	19.4	42.4	53.0
RetinaNet [63]	ResNet-101-FPN	800	1.5×	39.1	59.1	42.3	21.8	42.7	50.2
Deep Regionlets [109]	ResNet-101	800	1.5×	39.3	59.8	-	21.7	43.7	50.9
FitnessNMS [96]	ResNet-101	768	-	39.5	58.0	42.6	18.9	43.5	54.1
DetNet [57]	DetNet59-FPN	800	2×	40.3	62.1	43.8	23.6	42.6	50.0
CornerNet [52]	Hourglass-104	511	-	40.5	56.5	43.1	19.4	42.7	53.9
M2Det [129]	VGG16	800	∼ 10×	41.0	59.7	45.0	22.1	**46.5**	53.8
Faster R-CNN [62]	ResNet-101-FPN	800	1×	39.3	61.3	42.7	22.1	42.1	49.7
Faster R-CNN	HRNetV2p-W32	800	1×	39.5	61.2	43.0	23.3	41.7	49.1
Faster R-CNN [62]	ResNet-101-FPN	800	2×	40.3	61.8	43.9	22.6	43.1	51.0
Faster R-CNN	HRNetV2p-W32	800	2×	41.1	62.3	44.9	24.0	43.1	51.4
Faster R-CNN [62]	ResNet-152-FPN	800	2×	40.6	62.1	44.3	22.6	43.4	52.0
Faster R-CNN	HRNetV2p-W40	800	2×	42.1	63.2	46.1	24.6	44.5	52.6
Faster R-CNN [11]	X-101-64×4d-FPN	800	2×	41.1	62.8	44.8	23.5	44.1	52.3
Faster R-CNN	HRNetV2p-W48	800	2×	42.4	**63.6**	46.4	24.9	44.6	53.0
Cascade R-CNN [9]*	ResNet-101-FPN	800	∼ 1.6×	42.8	62.1	46.3	23.7	45.5	55.2
Cascade R-CNN	ResNet-101-FPN	800	∼ 1.6×	43.1	61.7	46.7	24.1	45.9	55.0
Cascade R-CNN	HRNetV2p-W32	800	∼ 1.6×	**43.7**	62.0	**47.4**	**25.5**	46.0	**55.3**

Table 9. Facial landmark detection results (NME) on WFLW `test` and 6 subsets: pose, expression (expr.), illumination (illu.), make-up (mu.), occlusion (occu.) and blur. LAB [101] is trained with extra boundary information (B). PDB [28] adopts stronger data augmentation (DA). Lower is better.

	backbone	test	pose	expr.	illu.	mu.	occu.	blur
ESR [10]	-	11.13	25.88	11.47	10.49	11.05	13.75	12.20
SDM [107]	-	10.29	24.10	11.45	9.32	9.38	13.03	11.28
CFSS [134]	-	9.07	21.36	10.09	8.30	8.74	11.76	9.96
DVLN [102]	VGG-16	6.08	11.54	6.78	5.73	5.98	7.33	6.88
Our approach	HRNetV2-W18	**4.60**	**7.94**	**4.85**	**4.55**	**4.29**	**5.44**	**5.42**
Model trained with `extra info.`								
LAB (w/ B) [101]	Hourglass	5.27	10.24	5.51	5.23	5.15	6.79	6.32
PDB (w/ DA) [28]	ResNet-50	5.11	8.75	5.36	4.93	5.41	6.37	5.81

Table 10. Facial landmark detection results (NME) on AFLW. DCFE [97] uses extra 3D information (3D). Lower is better.

	backbone	full	frontal
RCN [40]	-	5.60	5.36
CDM [116]	-	5.43	3.77
ERT [44]	-	4.35	2.75
LBF [80]	-	4.25	2.74
SDM [107]	-	4.05	2.94
CFSS [134]	-	3.92	2.68
RCPR [8]	-	3.73	2.87
CCL [135]	-	2.72	2.17
DAC-CSR [29]		2.27	1.81
TSR [69]	VGG-S	2.17	-
CPM + SBR [25]	CPM	2.14	-
SAN [24]	ResNet-152	1.91	1.85
DSRN [70]	-	1.86	-
LAB (w/o B) [101]	Hourglass	1.85	1.62
Our approach	HRNetV2-W18	**1.57**	**1.46**
Model trained with `extra info.`			
DCFE (w/ 3D) [97]	-	2.17	-
PDB (w/ DA) [28]	ResNet-50	1.47	-
LAB (w/ B) [101]	Hourglass	1.25	1.14

COFW. The COFW dataset [8] consists of $1,345$ training and 507 testing faces with occlusions, where each image has 29 facial landmarks.

Table 11 provides the comparison of our method with state-of-the-art methods. HRNetV2 outperforms other methods by a large margin. In particular, it achieves the better performance than LAB with extra boundary information and PDB with stronger data augmentation.

300W. The dataset [85] is a combination of HELEN [53],

LFPW [4], AFW [136], XM2VTS and IBUG datasets, where each face has 68 landmarks. Following [81], we use

Table 11. Facial landmark detection results on COFW `test`. The failure rate is calculated at the threshold 0.1. Lower is better for NME and $FR_{0.1}$.

	backbone	NME	$FR_{0.1}$
Human	-	5.60	-
ESR [10]	-	11.20	36.00
RCPR [8]	-	8.50	20.00
HPM [32]	-	7.50	13.00
CCR [27]	-	7.03	10.90
DRDA [118]	-	6.46	6.00
RAR [104]	-	6.03	4.14
DAC-CSR [29]	-	6.03	4.73
LAB (w/o B) [101]	Hourglass	5.58	2.76
Our approach	HRNetV2-W18	**3.45**	**0.19**
Model trained with `extra` *info.*			
PDB (w/ DA) [28]	ResNet-50	5.07	3.16
LAB (w/ B) [101]	Hourglass	3.92	0.39

Table 12. Facial landmark detection results (NME) on 300W: common, challenging and full. Lower is better.

	backbone	common	challenging	full
RCN [40]	-	4.67	8.44	5.41
DSRN [70]	-	4.12	9.68	5.21
PCD-CNN [51]	-	3.67	7.62	4.44
CPM + SBR [25]	CPM	3.28	7.58	4.10
SAN [24]	ResNet-152	3.34	6.60	3.98
DAN [50]	-	3.19	5.24	3.59
Our approach	HRNetV2-W18	**2.87**	**5.15**	**3.32**
Model trained with `extra` *info.*				
LAB (w/ B) [101]	Hourglass	2.98	5.19	3.49
DCFE (w/ 3D) [97]	-	2.76	5.22	3.24

the $3,148$ training images, which contains the training subsets of HELEN and LFPW and the full set of AFW. We evaluate the performance using two protocols, full set and test set. The full set contains 689 images and is further divided into a common subset (554 images) from HELEN and LFPW, and a challenging subset (135 images) from IBUG. The official test set, used for competition, contains 600 images (300 indoor and 300 outdoor images).

Table 12 provides the results on the full set, and its two subsets: common and challenging. Table 13 provides the results on the test set. In comparison to Chen et al. [17] that uses Hourglass with large parameter and computation complexity as the backbone, our scores are better except the $AUC_{0.08}$ scores. Our HRNetV2 gets the overall best performance among methods without extra information and stronger data augmentation, and is even better than LAB with extra boundary information and DCFE [97] that explores extra 3D information.

4.4. Empirical Analysis

We compare the modified networks, HRNetV2 and HRNetV2p, to the original network [91] (shortened as HRNetV1) on semantic segmentation and COCO object

Table 13. Facial landmark detection results on 300W `test`. DCFE [97] uses extra 3D information (3D). LAB [101] is trained with extra boundary information (B). Lower is better for NME, $FR_{0.08}$ and $FR_{0.1}$, and higher is better for $AUC_{0.08}$ and $AUC_{0.1}$.

|	backbone	NME	$AUC_{0.08}$	$AUC_{0.1}$	$FR_{0.08}$	$FR_{0.1}$
Balt. et al. [3]	-	-	19.55	-	38.83	-
ESR [10]	-	8.47	26.09	-	30.50	-
ERT [44]	-	8.41	27.01	-	28.83	-
LBF [80]	-	8.57	25.27	-	33.67	-
Face++ [130]	-	-	32.81	-	13.00	-
SDM [107]	-	5.83	36.27	-	13.00	-
CFAN [119]	-	5.78	34.78	-	14.00	-
Yan et al. [110]	-	-	34.97	-	12.67	-
CFSS [134]	-	5.74	36.58	-	12.33	-
MDM [94]	-	4.78	45.32	-	6.80	-
DAN [50]	-	4.30	47.00	-	2.67	-
Chen et al. [17]	Hourglass	3.96	**53.64**	-	2.50	-
Deng et al. [21]	-	-	-	47.52	-	5.50
Fan et al. [26]	-	-	-	48.02	-	14.83
DReg + MDM [35]	ResNet101	-	-	52.19	-	3.67
JMFA [22]	Hourglass	-	-	54.85	-	1.00
Our approach	HRNetV2-W18	**3.85**	52.09	**61.55**	**1.00**	**0.33**
Model trained with `extra` *info.*						
LAB (w/ B) [101]	Hourglass	-	-	58.85	-	0.83
DCFE (w/ 3D) [97]	-	3.88	52.42	-	1.83	-
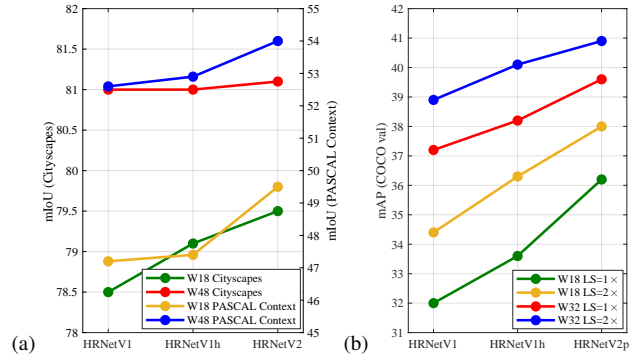
Figure 4. Empirical analysis. (a) Segmentation on Cityscapes `val` and PASCAL-Context `test` for comparing HRNetV1 and its variant HRNetV1h, and HRNetV2 (single scale and no flipping). (b) Object detection on COCO `val` for comparing HRNetV1 and its variant HRNetV1h, and HRNetV2p (LS = learning schedule).

detection. The segmentation and object detection results, given in Figure 4 (a) and Figure 4 (b), imply that HRNetV2 outperforms HRNetV1 significantly, except that the gain is minor in the large model case in segmentation for Cityscapes. We also test a variant (denoted by HRNetV1h), which is built by appending a 1×1 convolution to increase the dimension of the output high-resolution representation. The results in Figure 4 (a) and Figure 4 (b) show that the variant achieves slight improvement to HRNetV1, implying that aggregating the representations from low-resolution parallel convolutions in our HRNetV2 is essential for increasing the capability.

5. Conclusions

In this paper, we empirically study the high-resolution representation network in a broad range of vision applications with introducing a simple modification. Experimental results demonstrate the effectiveness of strong high-resolution representations and multi-level representations learned by the modified networks on semantic segmentation, facial landmark detection as well as object detection. The project page is https://jingdongwang2017.github.io/Projects/HRNet/.

Appendix: Network Pretraining

We pretrain our network, which is augmented by a classification head shown in Figure 5, on ImageNet [84]. The classification head is described as below. First, the four-resolution feature maps are fed into a bottleneck and the output channels are increased from C, $2C$, $4C$, and $8C$ to 128, 256, 512, and 1024, respectively. Then, we downsample the high-resolution representation by a 2-strided 3×3 convolution outputting 256 channels and add it to the representation of the second-high-resolution. This process is repeated two times to get 1024 feature channels over the small resolution. Last, we transform the 1024 channels to 2048 channels through a 1×1 convolution, followed by a global average pooling operation. The output 2048-dimensional representation is fed into the classifier.

We adopt the same data augmentation scheme for training images as in [39], and train our models for 100 epochs with a batch size of 256. The initial learning rate is set to 0.1 and is reduced by 10 times at epoch 30, 60 and 90. We use SGD with a weight decay of 0.0001 and a Nesterov momentum of 0.9. We adopt standard single-crop testing, so that 224×224 pixels are cropped from each image. The top-1 and top-5 error are reported on the validation set.

Table 14 shows our ImageNet classification results. As a comparison, we also report the results of ResNets. We consider two types of residual units: One is formed by a bottleneck, and the other is formed by two 3×3 convolutions. We follow the PyTorch implementation of ResNets and replace the 7×7 convolution in the input stem with two 2-strided 3×3 convolutions decreasing the resolution to $1/4$ as in our networks. When the residual units are formed by two 3×3 convolutions, an extra bottleneck is used to increase the dimension of output feature maps from 512 to 2048. One can see that under similar #parameters and GFLOPs, our results are comparable to and slightly better than ResNets.

In addition, we look at the results of two alternative schemes: (i) the feature maps on each resolution go through a global pooling separately and then are concatenated together to output a $15C$-dimensional representation vector, named HRNet-Wx-Ci; (ii) the feature maps on each resolution are fed into several 2-strided residual units (bottleneck,
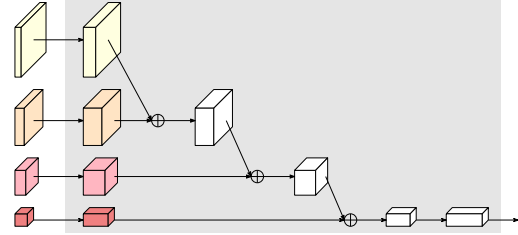

Figure 5. Representation for ImageNet classification. The input of the box is the representations of four resolutions.

each dimension is increased to the double) to increase the dimension to 512, and concatenate and average-pool them together to reach a 2048-dimensional representation vector, named HRNet-Wx-Cii, which is used in [91]. Table 15 shows such an ablation study. One can see that the proposed manner is superior to the two alternatives.

Table 14. ImageNet Classification results of HRNet and ResNets. The proposed method is named HRNet-Wx-C.

	#Params.	GFLOPs	top-1 err.	top-5 err.
Residual branch formed by two 3×3 convolutions				
ResNet-38	28.3M	3.80	24.6%	7.4%
HRNet-W18-C	21.3M	3.99	**23.1**%	**6.5**%
ResNet-72	48.4M	7.46	23.3%	6.7%
HRNet-W30-C	37.7M	7.55	**21.9**%	**5.9**%
ResNet-106	64.9M	11.1	22.7%	6.4%
HRNet-W40-C	57.6M	11.8	**21.1**%	**5.6**%
Residual branch formed by a bottleneck				
ResNet-50	25.6M	3.82	23.3%	6.6%
HRNet-W44-C	21.9M	3.90	**23.0**%	**6.5**%
ResNet-101	44.6M	7.30	21.6%	5.8%
HRNet-W76-C	40.8M	7.30	**21.5**%	**5.8**%
ResNet-152	60.2M	10.7	21.2%	5.7%
HRNet-W96-C	57.5M	10.2	**21.0**%	**5.6**%

Table 15. Ablation study on ImageNet classification by comparing our approach (abbreviated as HRNet-Wx-C) with two alternatives: HRNet-Wx-Ci and HRNet-Wx-Cii (residual branch formed by two 3×3 convolutions).

	#Params.	GFLOPs	top-1 err.	top-5 err.
HRNet-W27-Ci	21.4M	5.55	26.0%	7.7%
HRNet-W25-Cii	21.7M	5.04	24.1%	7.1%
HRNet-W18-C	21.3M	3.99	**23.1**%	**6.5**%
HRNet-W36-Ci	37.5M	9.00	24.3%	7.3%
HRNet-W34-Cii	36.7M	8.29	22.8%	6.3%
HRNet-W30-C	37.7M	7.55	**21.9**%	**5.9**%
HRNet-W45-Ci	58.2M	13.4	23.6%	7.0%
HRNet-W43-Cii	56.3 M	12.5	22.2%	6.1%
HRNet-W40-C	57.6M	11.8	**21.1**%	**5.6**%

References

[1] A. Arnab, S. Jayasumana, S. Zheng, and P. H. S. Torr. Higher order conditional random fields in deep neural networks. In *ECCV*, pages 524–540, 2016. 5

[2] V. Badrinarayanan, A. Kendall, and R. Cipolla. Segnet: A deep convolutional encoder-decoder architecture for im-

age segmentation. *IEEE Trans. Pattern Anal. Mach. Intell.*, 39(12):2481–2495, 2017. 1, 2

[3] T. Baltrusaitis, P. Robinson, and L. Morency. Constrained local neural fields for robust facial landmark detection in the wild. In *ICCVW*, pages 354–361, 2013. 8

[4] P. N. Belhumeur, D. W. Jacobs, D. J. Kriegman, and N. Kumar. Localizing parts of faces using a consensus of exemplars. *IEEE Trans. Pattern Anal. Mach. Intell.*, 35(12):2930–2940, 2013. 7

[5] A. Bulat and G. Tzimiropoulos. Human pose estimation via convolutional part heatmap regression. In *ECCV*, volume 9911 of *Lecture Notes in Computer Science*, pages 717–732. Springer, 2016. 2

[6] A. Bulat and G. Tzimiropoulos. Binarized convolutional landmark localizers for human pose estimation and face alignment with limited resources. In *ICCV*, pages 3726–3734. IEEE Computer Society, 2017. 2

[7] A. Bulat and G. Tzimiropoulos. How far are we from solving the 2d & 3d face alignment problem? (and a dataset of 230, 000 3d facial landmarks). In *ICCV*, pages 1021–1030, 2017. 2

[8] X. P. Burgos-Artizzu, P. Perona, and P. Dollár. Robust face landmark estimation under occlusion. In *ICCV*, pages 1513–1520, 2013. 6, 7, 8

[9] Z. Cai and N. Vasconcelos. Cascade R-CNN: delving into high quality object detection. In *CVPR*, pages 6154–6162, 2018. 2, 7

[10] X. Cao, Y. Wei, F. Wen, and J. Sun. Face alignment by explicit shape regression. In *CVPR*, pages 2887–2894, 2012. 7, 8

[11] K. Chen, J. Pang, J. Wang, Y. Xiong, X. Li, S. Sun, W. Feng, Z. Liu, J. Shi, W. Ouyang, C. C. Loy, and D. Lin. mmdetection. https://github.com/open-mmlab/mmdetection, 2018. 6, 7

[12] L. Chen, G. Papandreou, I. Kokkinos, K. Murphy, and A. L. Yuille. Semantic image segmentation with deep convolutional nets and fully connected crfs. *CoRR*, abs/1412.7062, 2014. 2

[13] L. 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 Trans. Pattern Anal. Mach. Intell.*, 40(4):834–848, 2018. 1, 2, 4, 5

[14] L. Chen, G. Papandreou, F. Schroff, and H. Adam. Rethinking atrous convolution for semantic image segmentation. *CoRR*, abs/1706.05587, 2017. 4

[15] L. Chen, Y. Yang, J. Wang, W. Xu, and A. L. Yuille. Attention to scale: Scale-aware semantic image segmentation. In *CVPR*, pages 3640–3649, 2016. 2

[16] L. Chen, Y. Zhu, G. Papandreou, F. Schroff, and H. Adam. Encoder-decoder with atrous separable convolution for semantic image segmentation. In *ECCV*, pages 833–851, 2018. 2, 4, 5

[17] Y. Chen, C. Shen, X. Wei, L. Liu, and J. Yang. Adversarial posenet: A structure-aware convolutional network for human pose estimation. In *ICCV*, pages 1221–1230, 2017. 6, 8

[18] Z. Chen, S. Huang, and D. Tao. Context refinement for object detection. In *ECCV*, pages 74–89, 2018. 7

[19] 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 *CVPR*, pages 3213–3223, 2016. 4

[20] J. Dai, K. He, and J. Sun. Boxsup: Exploiting bounding boxes to supervise convolutional networks for semantic segmentation. In *ICCV*, pages 1635–1643, 2015. 5

[21] J. Deng, Q. Liu, J. Yang, and D. Tao. M^{3}CSR: multi-view, multi-scale and multi-component cascade shape regression. *Image Vision Comput.*, 47:19–26, 2016. 8

[22] J. Deng, G. Trigeorgis, Y. Zhou, and S. Zafeiriou. Joint multi-view face alignment in the wild. *CoRR*, abs/1708.06023, 2017. 2, 8

[23] H. Ding, X. Jiang, B. Shuai, A. Q. Liu, and G. Wang. Context contrasted feature and gated multi-scale aggregation for scene segmentation. In *CVPR*, pages 2393–2402, 2018. 5

[24] X. Dong, Y. Yan, W. Ouyang, and Y. Yang. Style aggregated network for facial landmark detection. In *CVPR*, pages 379–388, 2018. 7, 8

[25] X. Dong, S. Yu, X. Weng, S. Wei, Y. Yang, and Y. Sheikh. Supervision-by-registration: An unsupervised approach to improve the precision of facial landmark detectors. In *CVPR*, pages 360–368, 2018. 7, 8

[26] H. Fan and E. Zhou. Approaching human level facial landmark localization by deep learning. *Image Vision Comput.*, 47:27–35, 2016. 8

[27] Z. Feng, P. Huber, J. Kittler, W. J. Christmas, and X. Wu. Random cascaded-regression copse for robust facial landmark detection. *IEEE Signal Process. Lett.*, 22(1):76–80, 2015. 8

[28] Z. Feng, J. Kittler, M. Awais, P. Huber, and X. Wu. Wing loss for robust facial landmark localisation with convolutional neural networks. In *CVPR*, pages 2235–2245. IEEE Computer Society, 2018. 6, 7, 8

[29] Z. Feng, J. Kittler, W. J. Christmas, P. Huber, and X. Wu. Dynamic attention-controlled cascaded shape regression exploiting training data augmentation and fuzzy-set sample weighting. In *CVPR*, pages 3681–3690. IEEE Computer Society, 2017. 7, 8

[30] D. Fourure, R. Emonet, É. Fromont, D. Muselet, A. Trémeau, and C. Wolf. Residual conv-deconv grid network for semantic segmentation. In *BMVC*, 2017. 1, 2, 3, 4

[31] J. Fu, J. Liu, Y. Wang, and H. Lu. Stacked deconvolutional network for semantic segmentation. *CoRR*, abs/1708.04943, 2017. 2

[32] G. Ghiasi and C. C. Fowlkes. Occlusion coherence: Localizing occluded faces with a hierarchical deformable part model. In *CVPR*, pages 1899–1906. IEEE Computer Society, 2014. 8

[33] G. Ghiasi and C. C. Fowlkes. Laplacian pyramid reconstruction and refinement for semantic segmentation. In *ECCV*, pages 519–534, 2016. 4

[34] K. Gong, X. Liang, X. Shen, and L. Lin. Look into person: Self-supervised structure-sensitive learning and A new

benchmark for human parsing. *CoRR*, abs/1703.05446, 2017. 4, 5

[35] R. A. Güler, G. Trigeorgis, E. Antonakos, P. Snape, S. Zafeiriou, and I. Kokkinos. Denseres: Fully convolutional dense shape regression in-the-wild. In *CVPR*, pages 2614–2623, 2017. 8

[36] J. Guo, J. Deng, N. Xue, and S. Zafeiriou. Stacked dense u-nets with dual transformers for robust face alignment. In *BMVC*, page 44, 2018. 2

[37] K. He, R. B. Girshick, and P. Dollár. Rethinking imagenet pre-training. *CoRR*, abs/1811.08883, 2018. 6

[38] K. He, G. Gkioxari, P. Dollár, and R. B. Girshick. Mask R-CNN. In *ICCV*, pages 2980–2988, 2017. 2, 5

[39] K. He, X. Zhang, S. Ren, and J. Sun. Deep residual learning for image recognition. In *CVPR*, pages 770–778, 2016. 9

[40] S. Honari, J. Yosinski, P. Vincent, and C. J. Pal. Recombinator networks: Learning coarse-to-fine feature aggregation. In *CVPR*, pages 5743–5752, 2016. 2, 7, 8

[41] H. Hu, J. Gu, Z. Zhang, J. Dai, and Y. Wei. Relation networks for object detection. In *CVPR*, pages 3588–3597, 2018. 7

[42] M. A. Islam, M. Rochan, N. D. B. Bruce, and Y. Wang. Gated feedback refinement network for dense image labeling. In *CVPR*, pages 4877–4885, 2017. 2

[43] X. Jin, X. Li, H. Xiao, X. Shen, Z. Lin, J. Yang, Y. Chen, J. Dong, L. Liu, Z. Jie, J. Feng, and S. Yan. Video scene parsing with predictive feature learning. In *ICCV*, pages 5581–5589, 2017. 4

[44] V. Kazemi and J. Sullivan. One millisecond face alignment with an ensemble of regression trees. In *CVPR*, pages 1867–1874, 2014. 7, 8

[45] T. Ke, J. Hwang, Z. Liu, and S. X. Yu. Adaptive affinity fields for semantic segmentation. In *ECCV*, pages 605–621, 2018. 4

[46] S. Kim, H. Kook, J. Sun, M. Kang, and S. Ko. Parallel feature pyramid network for object detection. In *ECCV*, pages 239–256, 2018. 7

[47] S. Kong and C. C. Fowlkes. Recurrent scene parsing with perspective understanding in the loop. In *CVPR*, pages 956–965, 2018. 4

[48] T. Kong, F. Sun, W. Huang, and H. Liu. Deep feature pyramid reconfiguration for object detection. In *ECCV*, pages 172–188, 2018. 7

[49] M. Köstinger, P. Wohlhart, P. M. Roth, and H. Bischof. Annotated facial landmarks in the wild: A large-scale, real-world database for facial landmark localization. In *ICCV*, pages 2144–2151, 2011. 6

[50] M. Kowalski, J. Naruniec, and T. Trzcinski. Deep alignment network: A convolutional neural network for robust face alignment. *CoRR*, abs/1706.01789, 2017. 2, 8

[51] A. Kumar and R. Chellappa. Disentangling 3d pose in a dendritic CNN for unconstrained 2d face alignment. In *CVPR*, pages 430–439. IEEE Computer Society, 2018. 8

[52] H. Law and J. Deng. Cornernet: Detecting objects as paired keypoints. In *ECCV*, pages 765–781, 2018. 7

[53] V. Le, J. Brandt, Z. Lin, L. D. Bourdev, and T. S. Huang. Interactive facial feature localization. In *ECCV (3)*, volume 7574 of *Lecture Notes in Computer Science*, pages 679–692. Springer, 2012. 7

[54] H. Li, P. Xiong, J. An, and L. Wang. Pyramid attention network for semantic segmentation. In *BMVC*, page 285, 2018. 4

[55] X. Li, Z. Liu, P. Luo, C. C. Loy, and X. Tang. Not all pixels are equal: Difficulty-aware semantic segmentation via deep layer cascade. In *CVPR*, pages 6459–6468, 2017. 4

[56] Z. Li, Y. Chen, G. Yu, and Y. Deng. R-FCN++: towards accurate region-based fully convolutional networks for object detection. In *AAAI*, pages 7073–7080, 2018. 6

[57] Z. Li, C. Peng, G. Yu, X. Zhang, Y. Deng, and J. Sun. Detnet: Design backbone for object detection. In *ECCV*, pages 339–354, 2018. 2, 7

[58] X. Liang, K. Gong, X. Shen, and L. Lin. Look into person: Joint body parsing & pose estimation network and A new benchmark. *CoRR*, abs/1804.01984, 2018. 5

[59] X. Liang, H. Zhou, and E. Xing. Dynamic-structured semantic propagation network. In *CVPR*, pages 752–761, 2018. 4

[60] G. Lin, A. Milan, C. Shen, and I. D. Reid. Refinenet: Multipath refinement networks for high-resolution semantic segmentation. In *CVPR*, pages 5168–5177, 2017. 2, 4, 5

[61] G. Lin, C. Shen, A. van den Hengel, and I. D. Reid. Efficient piecewise training of deep structured models for semantic segmentation. In *CVPR*, pages 3194–3203, 2016. 4, 5

[62] T. Lin, P. Dollár, R. B. Girshick, K. He, B. Hariharan, and S. J. Belongie. Feature pyramid networks for object detection. In *CVPR*, pages 936–944, 2017. 2, 4, 6, 7

[63] T. Lin, P. Goyal, R. B. Girshick, K. He, and P. Dollár. Focal loss for dense object detection. In *ICCV*, pages 2999–3007, 2017. 5, 7

[64] S. Liu, D. Huang, and Y. Wang. Receptive field block net for accurate and fast object detection. In *ECCV*, pages 404–419, 2018. 7

[65] S. Liu, L. Qi, H. Qin, J. Shi, and J. Jia. Path aggregation network for instance segmentation. In *CVPR*, pages 8759–8768, 2018. 6

[66] T. Liu, T. Ruan, Z. Huang, Y. Wei, S. Wei, Y. Zhao, and T. Huang. Devil in the details: Towards accurate single and multiple human parsing. *CoRR*, abs/1809.05996, 2018. 5

[67] J. Long, E. Shelhamer, and T. Darrell. Fully convolutional networks for semantic segmentation. In *CVPR*, pages 3431–3440, 2015. 2

[68] Y. Luo, Z. Zheng, L. Zheng, T. Guan, J. Yu, and Y. Yang. Macro-micro adversarial network for human parsing. In *ECCV*, pages 424–440, 2018. 5

[69] J. Lv, X. Shao, J. Xing, C. Cheng, and X. Zhou. A deep regression architecture with two-stage re-initialization for high performance facial landmark detection. In *CVPR*, pages 3691–3700, 2017. 2, 7

[70] X. Miao, X. Zhen, X. Liu, C. Deng, V. Athitsos, and H. Huang. Direct shape regression networks for end-to-end face alignment. In *CVPR*, pages 5040–5049, 2018. 7, 8

[71] R. Mottaghi, X. Chen, X. Liu, N. Cho, S. Lee, S. Fidler, R. Urtasun, and A. L. Yuille. The role of context for object

detection and semantic segmentation in the wild. In *CVPR*, pages 891–898, 2014. 4, 5

[72] A. Newell, K. Yang, and J. Deng. Stacked hourglass networks for human pose estimation. In *ECCV*, pages 483–499, 2016. 1, 2, 6

[73] X. Nie, J. Feng, and S. Yan. Mutual learning to adapt for joint human parsing and pose estimation. In *ECCV*, pages 519–534, 2018. 5

[74] H. Noh, S. Hong, and B. Han. Learning deconvolution network for semantic segmentation. In *ICCV*, pages 1520–1528, 2015. 1, 2

[75] C. Peng, T. Xiao, Z. Li, Y. Jiang, X. Zhang, K. Jia, G. Yu, and J. Sun. Megdet: A large mini-batch object detector. In *CVPR*, pages 6181–6189, 2018. 6

[76] C. Peng, X. Zhang, G. Yu, G. Luo, and J. Sun. Large kernel matters - improve semantic segmentation by global convolutional network. In *CVPR*, pages 1743–1751, 2017. 2, 4

[77] X. Peng, R. S. Feris, X. Wang, and D. N. Metaxas. A recurrent encoder-decoder network for sequential face alignment. In *ECCV (1)*, volume 9905 of *Lecture Notes in Computer Science*, pages 38–56. Springer, 2016. 1, 2

[78] T. Pohlen, A. Hermans, M. Mathias, and B. Leibe. Full-resolution residual networks for semantic segmentation in street scenes. In *CVPR*, pages 3309–3318, 2017. 2, 4

[79] L. Qi, S. Liu, J. Shi, and J. Jia. Sequential context encoding for duplicate removal. In *NeurIPS*, pages 2053–2062, 2018. 6

[80] S. Ren, X. Cao, Y. Wei, and J. Sun. Face alignment at 3000 FPS via regressing local binary features. In *CVPR*, pages 1685–1692, 2014. 7, 8

[81] S. Ren, X. Cao, Y. Wei, and J. Sun. Face alignment via regressing local binary features. *IEEE Trans. Image Processing*, 25(3):1233–1245, 2016. 7

[82] S. Ren, K. He, R. B. Girshick, and J. Sun. Faster R-CNN: towards real-time object detection with region proposal networks. volume abs/1506.01497, 2015. 5

[83] O. Ronneberger, P. Fischer, and T. Brox. U-net: Convolutional networks for biomedical image segmentation. In *MICCAI*, pages 234–241, 2015. 1, 2

[84] O. Russakovsky, J. Deng, H. Su, J. Krause, S. Satheesh, S. Ma, Z. Huang, A. Karpathy, A. Khosla, M. S. Bernstein, A. C. Berg, and F. Li. Imagenet large scale visual recognition challenge. *International Journal of Computer Vision*, 115(3):211–252, 2015. 9

[85] C. Sagonas, G. Tzimiropoulos, S. Zafeiriou, and M. Pantic. 300 faces in-the-wild challenge: The first facial landmark localization challenge. In *ICCV Workshops*, pages 397–403. IEEE Computer Society, 2013. 6, 7

[86] S. Saxena and J. Verbeek. Convolutional neural fabrics. In *NIPS*, pages 4053–4061, 2016. 1, 2

[87] P. Sermanet, D. Eigen, X. Zhang, M. Mathieu, R. Fergus, and Y. LeCun. Overfeat: Integrated recognition, localization and detection using convolutional networks. *CoRR*, abs/1312.6229, 2013. 2

[88] E. Shelhamer, J. Long, and T. Darrell. Fully convolutional networks for semantic segmentation. *IEEE Trans. Pattern Anal. Mach. Intell.*, 39(4):640–651, 2017. 5

[89] B. Singh and L. S. Davis. An analysis of scale invariance in object detection SNIP. In *CVPR*, pages 3578–3587, 2018. 6

[90] B. Singh, M. Najibi, and L. S. Davis. SNIPER: efficient multi-scale training. In *NeurIPS*, pages 9333–9343, 2018. 6

[91] K. Sun, B. Xiao, D. Liu, and J. Wang. Deep high-resolution representation learning for human pose estimation. In *CVPR*, 2019. 1, 2, 3, 4, 8, 9

[92] Y. Sun, X. Wang, and X. Tang. Deep convolutional network cascade for facial point detection. In *CVPR*, pages 3476–3483. IEEE Computer Society, 2013. 2

[93] Z. Tang, X. Peng, S. Geng, L. Wu, S. Zhang, and D. N. Metaxas. Quantized densely connected u-nets for efficient landmark localization. In *ECCV*, pages 348–364, 2018. 2

[94] G. Trigeorgis, P. Snape, M. A. Nicolaou, E. Antonakos, and S. Zafeiriou. Mnemonic descent method: A recurrent process applied for end-to-end face alignment. In *CVPR*, pages 4177–4187, 2016. 2, 8

[95] L. Tychsen-Smith and L. Petersson. Denet: Scalable real-time object detection with directed sparse sampling. In *ICCV*, pages 428–436, 2017. 7

[96] L. Tychsen-Smith and L. Petersson. Improving object localization with fitness NMS and bounded iou loss. In *CVPR*, pages 6877–6885, 2018. 7

[97] R. Valle, J. M. Buenaposada, A. Valdés, and L. Baumela. A deeply-initialized coarse-to-fine ensemble of regression trees for face alignment. In *ECCV*, pages 609–624, 2018. 2, 7, 8

[98] H. Wang, Q. Wang, M. Gao, P. Li, and W. Zuo. Multi-scale location-aware kernel representation for object detection. In *CVPR*, pages 1248–1257, 2018. 7

[99] P. Wang, P. Chen, Y. Yuan, D. Liu, Z. Huang, X. Hou, and G. W. Cottrell. Understanding convolution for semantic segmentation. In *WACV*, 2018. 4

[100] Z. Wojna, J. R. R. Uijlings, S. Guadarrama, N. Silberman, L. Chen, A. Fathi, and V. Ferrari. The devil is in the decoder. In *BMVC*, 2017. 2

[101] W. Wu, C. Qian, S. Yang, Q. Wang, Y. Cai, and Q. Zhou. Look at boundary: A boundary-aware face alignment algorithm. In *CVPR*, pages 2129–2138, 2018. 2, 6, 7, 8

[102] W. Wu and S. Yang. Leveraging intra and inter-dataset variations for robust face alignment. In *CVPR*, pages 2096–2105, 2017. 7

[103] Z. Wu, C. Shen, and A. van den Hengel. Wider or deeper: Revisiting the resnet model for visual recognition. *CoRR*, abs/1611.10080, 2016. 4

[104] S. Xiao, J. Feng, J. Xing, H. Lai, S. Yan, and A. A. Kassim. Robust facial landmark detection via recurrent attentive-refinement networks. In *ECCV*, pages 57–72, 2016. 2, 8

[105] T. Xiao, Y. Liu, B. Zhou, Y. Jiang, and J. Sun. Unified perceptual parsing for scene understanding. In *ECCV*, pages 432–448, 2018. 2

[106] S. Xie and Z. Tu. Holistically-nested edge detection. In *ICCV*, pages 1395–1403, 2015. 2

[107] X. Xiong and F. D. la Torre. Supervised descent method and its applications to face alignment. In *CVPR*, pages 532–539. IEEE Computer Society, 2013. 7, 8

[108] D. Xu, W. Ouyang, X. Wang, and N. Sebe. Pad-net: Multi-tasks guided prediction-and-distillation network for simultaneous depth estimation and scene parsing. In *CVPR*, pages 675–684, 2018. 4

[109] H. Xu, X. Lv, X. Wang, Z. Ren, N. Bodla, and R. Chellappa. Deep regionlets for object detection. In *ECCV*, pages 827–844, 2018. 7

[110] J. Yan, Z. Lei, D. Yi, and S. Z. Li. Learn to combine multiple hypotheses for accurate face alignment. In *ICCVW*, pages 392–396, 2013. 8

[111] J. Yang, Q. Liu, and K. Zhang. Stacked hourglass network for robust facial landmark localisation. In *CVPR*, pages 2025–2033, 2017. 2

[112] S. Yang, P. Luo, C. C. Loy, and X. Tang. WIDER FACE: A face detection benchmark. In *CVPR*, pages 5525–5533. IEEE Computer Society, 2016. 6

[113] C. Yu, J. Wang, C. Peng, C. Gao, G. Yu, and N. Sang. Bisenet: Bilateral segmentation network for real-time semantic segmentation. In *ECCV*, pages 334–349, 2018. 4

[114] C. Yu, J. Wang, C. Peng, C. Gao, G. Yu, and N. Sang. Learning a discriminative feature network for semantic segmentation. In *CVPR*, pages 1857–1866, 2018. 4

[115] F. Yu, V. Koltun, and T. A. Funkhouser. Dilated residual networks. *CoRR*, abs/1705.09914, 2017. 2

[116] X. Yu, J. Huang, S. Zhang, W. Yan, and D. N. Metaxas. Pose-free facial landmark fitting via optimized part mixtures and cascaded deformable shape model. In *ICCV*, pages 1944–1951. IEEE Computer Society, 2013. 7

[117] H. Zhang, K. J. Dana, J. Shi, Z. Zhang, X. Wang, A. Tyagi, and A. Agrawal. Context encoding for semantic segmentation. In *CVPR*, pages 7151–7160, 2018. 5

[118] J. Zhang, M. Kan, S. Shan, and X. Chen. Occlusion-free face alignment: Deep regression networks coupled with decorrupt autoencoders. In *CVPR*, pages 3428–3437. IEEE Computer Society, 2016. 8

[119] J. Zhang, S. Shan, M. Kan, and X. Chen. Coarse-to-fine auto-encoder networks (CFAN) for real-time face alignment. In *ECCV (2)*, volume 8690 of *Lecture Notes in Computer Science*, pages 1–16. Springer, 2014. 2, 8

[120] R. Zhang, S. Tang, Y. Zhang, J. Li, and S. Yan. Scale-adaptive convolutions for scene parsing. In *ICCV*, pages 2050–2058, 2017. 4

[121] S. Zhang, L. Wen, X. Bian, Z. Lei, and S. Z. Li. Single-shot refinement neural network for object detection. In *CVPR*, pages 4203–4212, 2018. 7

[122] T. Zhang, G. Qi, B. Xiao, and J. Wang. Interleaved group convolutions. In *ICCV*, pages 4383–4392, 2017. 3

[123] Z. Zhang, P. Luo, C. C. Loy, and X. Tang. Facial landmark detection by deep multi-task learning. In *ECCV*, pages 94–108, 2014. 2

[124] Z. Zhang, S. Qiao, C. Xie, W. Shen, B. Wang, and A. L. Yuille. Single-shot object detection with enriched semantics. In *CVPR*, pages 5813–5821, 2018. 7

[125] Z. Zhang, X. Zhang, C. Peng, X. Xue, and J. Sun. Exfuse: Enhancing feature fusion for semantic segmentation. In *ECCV*, pages 273–288, 2018. 2

[126] H. Zhao, J. Shi, X. Qi, X. Wang, and J. Jia. Pyramid scene parsing network. In *CVPR*, pages 6230–6239, 2017. 1, 2, 4, 5

[127] H. Zhao, Y. Zhang, S. Liu, J. Shi, C. C. Loy, D. Lin, and J. Jia. Psanet: Point-wise spatial attention network for scene parsing. In *ECCV*, pages 270–286, 2018. 4

[128] J. Zhao, J. Li, X. Nie, F. Zhao, Y. Chen, Z. Wang, J. Feng, and S. Yan. Self-supervised neural aggregation networks for human parsing. In *CVPRW*, pages 1595–1603, 2017. 5

[129] Q. Zhao, T. Sheng, Y. Wang, Z. Tang, Y. Chen, L. Cai, and H. Ling. M2det: A single-shot object detector based on multi-level feature pyramid network. *CoRR*, abs/1811.04533, 2018. 7

[130] E. Zhou, H. Fan, Z. Cao, Y. Jiang, and Q. Yin. Extensive facial landmark localization with coarse-to-fine convolutional network cascade. In *ICCVW*, pages 386–391, 2013. 8

[131] P. Zhou, B. Ni, C. Geng, J. Hu, and Y. Xu. Scale-transferrable object detection. In *CVPR*, pages 528–537, 2018. 7

[132] Y. Zhou, X. Hu, and B. Zhang. Interlinked convolutional neural networks for face parsing. In *ISNN*, pages 222–231, 2015. 1, 2, 3

[133] Z. Zhou, M. M. R. Siddiquee, N. Tajbakhsh, and J. Liang. Unet++: A nested u-net architecture for medical image segmentation. In *MICCAI*, pages 3–11, 2018. 2, 4, 5

[134] S. Zhu, C. Li, C. C. Loy, and X. Tang. Face alignment by coarse-to-fine shape searching. In *CVPR*, pages 4998–5006. IEEE Computer Society, 2015. 6, 7, 8

[135] S. Zhu, C. Li, C. C. Loy, and X. Tang. Unconstrained face alignment via cascaded compositional learning. In *CVPR*, pages 3409–3417, 2016. 7

[136] X. Zhu and D. Ramanan. Face detection, pose estimation, and landmark localization in the wild. In *CVPR*, pages 2879–2886. IEEE Computer Society, 2012. 7

[137] Y. Zhu, C. Zhao, J. Wang, X. Zhao, Y. Wu, and H. Lu. Couplenet: Coupling global structure with local parts for object detection. In *ICCV*, pages 4146–4154, 2017. 7