DENSE POINT PREDICTION: A SIMPLE BASELINE FOR CROWD COUNTING AND LOCALIZATION

Yi Wang, Xinyu Hou, and Lap-Pui Chau

School of Electrical and Electronics Engineering, Nanyang Technological University, Singapore
E-mail: wang1241@e.ntu.edu.sg, houx0008@e.ntu.edu.sg, elpchau@ntu.edu.sg

ABSTRACT
In this paper, we propose a simple yet effective crowd counting and localization network named SCALNet. Unlike most existing works that separate the counting and localization tasks, we consider those tasks as a pixel-wise dense prediction problem and integrate them into an end-to-end framework. Specifically, for crowd counting, we adopt a counting head supervised by the Mean Square Error (MSE) loss. For crowd localization, the key insight is to recognize the keypoint of people, i.e., the center point of heads. We propose a localization head to distinguish dense crowds trained by two loss functions, i.e., Negative-Suppressed Focal (NSF) loss and False-Positive (FP) loss, which balances the positive/negative examples and handles the false-positive predictions. Experiments on the recent and large-scale benchmark, NWPU-Crowd, show that our approach outperforms the state-of-the-art methods by more than 5% and 10% improvement in crowd localization and counting tasks, respectively. The code is publicly available at https://github.com/WangyiNTU/SCALNet.

Index Terms— Crowd counting, crowd localization, convolutional neural network (CNN), dense prediction, keypoint estimation

1. INTRODUCTION
Crowd counting and crowd localization tasks, with a wide range of real-life applications, have attracted much attention in recent years. The count and location information are crucial for analyzing crowded pedestrians, traffic flow, and cells [1]. Many researchers applied Deep Neural Networks (DNNs) to achieve high performance in crowd counting. Recently, some researchers commence locating crowded objects beyond counting [2]. Examples of crowd counting and localization in dense crowds are shown in Fig. 1.

Conventionally, crowd counting and localization are considered as two separate tasks. State-of-the-art crowd counting methods are based on regression networks which focus on regressing the counts but ignore the locations of object instances. These methods aim to predict a density map wherein the count is calculated by the integral over the map [3][5], as shown in Fig. 1(b). Congested Scene Recognition Network (CSRNet) [6] utilizes VGG-16 [7] as the frontend, followed by the dilated convolution layers as the backend to generate a density map. Reference [8] emphasized the contextual information of the whole image by adding a spatial-/channel-wise attention regression (SCAR) to CNN. Chen et al. [9] proposed a two-stage architecture consisting of a band-pass stage and a rolling guidance stage to better extract multiple-scale features with multi-level information. Other regression-based works (like [10][13]) and detailed comparisons can refer to the survey paper [1] and benchmark [14].

Owing to the highly-occluded crowds and point-level annotations (heads are labeled by points), traditional object detection ([15][16]) methods produce low recall of crowds, damaging the counting performance. Recently, crowd localization methods have been developed for dense crowds, like [2][17][18], showing that those methods can localize crowds as well as obtain a comparable counting result. The localization process is to produce the keypoints [2][19], blobs [17], or bounding boxes [18][20][21] on each head (see Fig. 1(a)), and

Fig. 1. Illustration of crowd counting and localization. (a) Crowd localization (yellow circles) by the localization head of the proposed method. (b) Crowd counting by the counting head of the proposed method, where the predicted counts (1222 and 740) are shown at the bottom-left corner for the top and bottom images, respectively. The ground-truth counts are 1186 and 773. Zoom in the figure for better viewing.
We consider the crowd localization problem as a dense prediction (i.e., pixel-wise binary classification) and the crowd counting problem as a dense regression, such that many dense prediction networks can be employed as the backbone. Our network architecture is shown in Fig. 2.

**Backbone Network.** Based on CenterNet [25], we modify the Deep Layer Aggregation (DLA-34) [26] structure as the backbone network. The backbone provides a stage-wise fusion manner for multi-scale features from the stride of 32 to 2. Small-scale features are gradually fused with $2 \times$ up large-scale features, leading to the fusion of semantic information and spatial information. Instead of the original 4-stride DLA-34, we utilize the 2-stride features in the fusion step to retain more spatial information, as shown at the top row of the network in Fig. 2. Finally, different-level features are aggregated into 2-stride features by the deconvolutional layers and convolutional layers, as shown at the right column of the network. This modification significantly improves the localization ability with a little inference time increase (see Section 3.3).

**Localization Head Network (LHN).** Since crowds are densely distributed, the distance between two heads may be less than several pixels in an image. To distinguish those heads, LHN should output the full-resolution heatmap. Specifically, we propose a Simple yet effective Counting And Localization Network, named SCALNet, as shown in Fig. 2. For crowd counting, we build a simple counting head network supervised by the Mean Square Error (MSE) loss for density map regression. For crowd localization, we propose a keypoint localization network to recognize each human head of crowds. The supervisory information is the point-level annotations. To stably train the network, we propose to use two classification losses, i.e., Negative-Suppressed Focal (NSF) loss and False-Positive (FP) loss, which can balance the positive/negative examples and handle the false-positive predictions. The human heads’ locations can be obtained by decoding the local peaks on the output heatmap.

We evaluate our approach on the recent large-scale benchmark, NWPU-Crowd [14]. Experimental results show that our approach outperforms state-of-the-art methods by more than 5% improvement in crowd localization and 18% (resp. 12%) improvement on MAE (resp. MSE) in crowd counting.

The remainder of this paper is organized as follows. In Section 2, we present our approach in detail. Then, we conduct experiments and discuss the effectiveness of our approach in Section 3. Finally, we conclude our paper in Section 4.

### 2. PROPOSED METHOD

In this section, we present the proposed method, including network architecture, supervision, and loss functions.

**2.1. Network Architecture**

We consider the crowd localization problem as a dense prediction problem and build an end-to-end CNN to predict two dense feature maps. One contains the continuous labels (i.e., the density map), and the other contains the discrete labels (i.e., the heatmap with pixel-wise classification). Specifically, we propose a Simple yet effective Counting And Localization Network, named SCALNet, as shown in Fig. 2. For crowd counting, we build a simple counting head network supervised by the Mean Square Error (MSE) loss for density map regression. For crowd localization, we propose a keypoint localization network to recognize each human head of crowds. The supervisory information is the point-level annotations. To stably train the network, we propose to use two classification losses, i.e., Negative-Suppressed Focal (NSF) loss and False-Positive (FP) loss, which can balance the positive/negative examples and handle the false-positive predictions. The human heads’ locations can be obtained by decoding the local peaks on the output heatmap.

We evaluate our approach on the recent large-scale benchmark, NWPU-Crowd [14]. Experimental results show that our approach outperforms state-of-the-art methods by more than 5% improvement in crowd localization and 18% (resp. 12%) improvement on MAE (resp. MSE) in crowd counting.

The remainder of this paper is organized as follows. In Section 2, we present our approach in detail. Then, we conduct experiments and discuss the effectiveness of our approach in Section 3. Finally, we conclude our paper in Section 4.

2. **PROPOSED METHOD**

In this section, we present the proposed method, including network architecture, supervision, and loss functions.

2.1. **Network Architecture**

We consider the crowd localization problem as a dense classification (i.e., pixel-wise binary classification) and the crowd counting problem as a dense regression, such that many dense prediction networks can be employed as the backbone. Our network architecture is shown in Fig. 2.

**Backbone Network.** Based on CenterNet [25], we modify the Deep Layer Aggregation (DLA-34) [26] structure as the backbone network. The backbone provides a stage-wise fusion manner for multi-scale features from the stride of 32 to 2. Small-scale features are gradually fused with $2 \times$ up large-scale features, leading to the fusion of semantic information and spatial information. Instead of the original 4-stride DLA-34, we utilize the 2-stride features in the fusion step to retain more spatial information, as shown at the top row of the network in Fig. 2. Finally, different-level features are aggregated into 2-stride features by the deconvolutional layers and convolutional layers, as shown at the right column of the network. This modification significantly improves the localization ability with a little inference time increase (see Section 3.3).

**Localization Head Network (LHN).** Since crowds are densely distributed, the distance between two heads may be less than several pixels in an image. To distinguish those heads, LHN should output the full-resolution heatmap. Specifically, the heatmap size is set to be the same as the input
size. Connected on the top layer of the backbone network, LHN is the sequential layers of Dconv(k4-s2)-Conv(k3-s1)-BN-ReLU-Conv(k1-s1)-Sigmoid, where Dconv(k4-s2) is a deconvolutional layer with the kernel of 4 and stride of 2, Conv is a convolutional layer, BN is the batch normalization, and ReLU (Rectified Linear Unit) and Sigmoid are the activation functions.

**Counting Head Network (CHN).** CHN aims to produce a density map by regression. The density map can accumulate the density values of crowded objects, so the low-resolution map satisfies the counting task. Specifically, CHN is fed by the output features of the backbone network and generates a 2-stride density map. The structure of CHN consists of Conv(k3-s1)-PReLU-Conv(k1-s1)-PReLU, where PReLU stands for the parametric ReLU.

**Inference.** Throughout a forward pass of the network, the output heatmap contains the human head’s estimate. The heads exist in the high-confidence points (peaks) of the heatmap. We can see that the heatmap in Fig. 2 contains sharp peaks, demonstrating a good localization ability. Following [25], we employ a 3×3 local maximum operation to filter out noisy points around the peaks and select the peaks by a confidence thresholding as heads’ estimate. The threshold is obtained by searching a confidence interval [0.3, 0.5] where the selected threshold achieves the best localization performance on the validation set. The heads’ locations are obtained by the coordinates of the peaks. From the counting head network, the headcount is obtained by simply summing all density values over the output density map.

2.2. Supervision

**Localization supervision.** We first generate the ground-truth dot map from the point-level annotations of the benchmark. The point annotations provide the coordinates of heads in an image. To be specific, we initialize each pixel value to 0 on the dot map, representing the negative labels. We set the pixel values to 1 according to the heads’ coordinates, representing the positive labels. Then, we follows [25] to generate the heatmap as localization supervision. The Gaussian kernels are placed at the head locations in order to smooth the supervision around the positive labels. We select the maximum value when two Gaussian kernels overlap. The Gaussian kernel is defined by

\[
K^l_{xy} = \exp \left( -\frac{(x - c_x)^2 + (y - c_y)^2}{2\sigma_d^2} \right),
\]

where \((x, y)\) is the pixel coordinate, \((c_x, c_y)\) is the coordinate of a head, and \(\sigma_d\) is the scale-related standard deviation calculated by its distance to the 3 nearest heads [3], i.e.,

\[
\sigma_d = 0.1 \cdot \sum_{i=1}^{3} d_i,
\]

where \(d_i\) is the distance to the \(i\)-th head.

**Counting supervision.** We generate the ground-truth density map as counting supervision. A normalized Gaussian kernel with the integration of 1 is accumulated at a head location. The normalized Gaussian kernel is defined by

\[
K^c_{xy} = \frac{1}{\sqrt{2\pi}\sigma_c} \exp \left( -\frac{(x - c_x)^2 + (y - c_y)^2}{2\sigma_c^2} \right),
\]

where \(\sigma_c\) is a constant standard deviation. We empirically set \(\sigma_c = 3\) for all experiments, which makes the kernel sharp.

2.3. Loss Functions

**Localization loss.** We employ a pixel-wise classification loss. The loss needs to handle the imbalance of training examples and the complex background. First, the number of positive examples (heads) is much less than that of negative examples (background). For the case of 1,000 pedestrians in a 1080×1080 image, the ratio of positive/negative is about 1:1000. Such an imbalance brings a challenge to the training process. Second, the complex background may cause many false-positive head detections.

In view of these issues, we propose to utilize two complementary loss functions, i.e., Negative-Suppressed Focal (NSF) loss and False-Positive (FP) loss, to stabilize the training process. The NSF loss adopts the focal cross-entropy loss with negative example suppression, like [21]:

\[
L_{nsf} = -\frac{1}{M} \sum_{j \in T} \left\{ (1 - \hat{p}_j)^{\gamma} \log(\hat{p}_j), \quad \text{if } p = 1, \\
\frac{1}{16} (1 - \hat{p}_j)^{\delta} \log(1 - \hat{p}_j), \quad \text{otherwise}, \right. \]

where \(M\) is the number of positive examples in image \(I\), \(\hat{p}_j\) is the predicted probability, \(p_j\) is the ground truth with 1 for the head and 0 for the background, \(\gamma\) is the parameter of the focal loss [27] to control the weights of well-classified examples, and \(\delta\) is the parameter to decrease the punishments of the negative labels around the positive ones. \(\frac{1}{16}\) is the down-weight to suppress the contribution of negative examples.

However, the NFS loss is a holistic suppression of the negative examples, but the false-positive peaks still appear in the heatmap due to the complex background. Thus, we propose an FP loss to emphasize false-positive detections. Specifically, we find the false-positive region according to the

Table 1. Two basic crowd tasks addressed by dense regression-based methods and dense classification-based methods.

| Method | Task | Crowd counting | Crowd localization |
|--------|------|----------------|-------------------|
| Density map prediction | [1,3,5,6,8,13] | Dense regression | Post-processing of density map [22,23] |
| Keypoint, blob, or box prediction | [2,17,21] | Confidence thresholding | Dense classification |
ground-truth heatmap and the predicted heatmap, denoted by $H$ and $H^\phi$, respectively. The background region is defined by $B = \{(x, y)|H_{xy} = 0\}$, and the positive region of $H$ is $P = \{(x, y)|H_{xy} > 0.1\}$. Hence, the false-positive region $F$ is formulated by:

$$F = B \cap P.$$  \hfill (4)

Formally, the FP loss is employed in the false-positive region:

$$L_{fp} = -\frac{1}{|F|} \sum_{j \in F} (\hat{p}_j)^\gamma \log(1 - \hat{p}_j),$$  \hfill (5)

where $|F|$ denotes the total number of pixels in $F$, and $\hat{p}_j$ stands for the predicted probability of the $j$-th pixel of $F$. The FP loss encourages the network to reduce the false-positive detections.

**Regression loss.** We adopt the MSE loss for density map regression. Hence, the regression loss is given as follow:

$$L_r = \frac{1}{N} \sum_{i \in N} \|D_j(I_i) - D_j^{gt}\|^2_2$$  \hfill (6)

where $N$ is the number of pixels in the density map, $I_i$ is the $i$-th image, and $D$ and $D^{gt}$ are the predicted and ground-truth density map, respectively.

The total loss is the sum of the NFS, FP, regression losses:

$$L = L_{nfs} + \lambda_1 L_{fp} + \lambda_2 L_r,$$  \hfill (7)

where $\lambda_1$ and $\lambda_2$ are the hype-parameters to control the weight between those losses.

## 3. EXPERIMENTAL RESULTS

In this section, we first describe the experimental configurations. Then, we compare our approach with state-of-the-art localization and counting methods. Finally, we discuss the effectiveness of the proposed modules.

### 3.1. Experimental Configurations

**Dataset.** We evaluate our method on the NWPU-Crowd [14] benchmark. The benchmark contains 5109 images and more than 2 million heads with the point and box labels. It is split into the training, validation, and test sets with 3109, 500, and 1500 images, respectively. We train our method in the training set and choose the best model by the validation set. The results in the testing set are obtained by submitting the predictions to the benchmark website.

**Metrics.** We employ the evaluation protocol [14] of NWPU-Crowd. For crowd localization, the Precision (Pre.), Recall (Rec.), and F1-measure (F1-m.) are calculated. The higher the value is, the better the performance achieves. The predicted point (the center point of a head) is true positive when its distance to the ground-truth point is less than a predefined distance threshold. Two thresholds are defined by $\sigma_s = \min(h, w)/2$ and $\sigma_l = \sqrt{h^2 + w^2}/2$, where $h$ and $w$ are the height and width of the head, respectively. The former is harder to reach than the latter. For crowd counting, we used the Mean Absolute Error (MAE), root Mean Squared Error (MSE), and mean Normalized Absolute Error (NAE). The lower the value is, the better the performance achieves.

**Implementation details.** The network was initialized by the pre-trained weights of DLA-34 [26] on ImageNet. We trained the network with the Adam optimizer for 180 epochs. The learning rate was set to 1e-4 and then decayed by 10 at the 135th epoch. The batch size was 32. In Eqs. (3) and (5), $\gamma$ and $\delta$ were set to 2 and 4, respectively. In Eq. (7), $\lambda_1$ and $\lambda_2$ were empirically set to 1 and 1000, respectively. We down-scale the longer side of images to 1600 pixels if it is larger than 1600. The training examples were augmented by random horizontal flipping, random cropping (320$\times$320), and normalization. The training and testing run on a single RTX 3080 Ti GPU.

### 3.2. Comparison with State-of-the-art Methods

**Crowd localization.** We first evaluate our SCALNet in crowd localization. Table 2 shows the comparison with the state-of-the-art crowd localization methods. With point training labels, the proposed approach achieves the best performance in F1-measure and Recall. The F1-measure and Recall are improved by more than 5% when compared with the recent method, Crowd-SDNet [21]. Although Faster RCNN [15] obtains the best Precision, it has the extremely low F1-measure and Recall, which means Faster RCNN precisely detects large-scale persons and fails in most dense crowds (see [14]). Some qualitative results of our method are shown in the second row of Fig. 3, illustrating that our method can predict the center points of heads accurately in crowded scenes.

**Crowd counting.** Table 3 shows the counting performance. It can be seen that our method outperforms the state-of-the-art methods in both MAE and MSE by a significant margin, with 18% and 12% relative improvement when compared with the second-best BL [12] and CANNet [11], respectively. The NAE score of our method is comparable to BL [12]. Note that our method only employs a simple backbone with counting

| Method          | Training Labels | F1-m. | Pre. | Rec. |
|-----------------|-----------------|-------|------|------|
| Faster RCNN*    | Box             | 6.7   | 95.8 | 3.5  |
| TinyFaces*      | Box             | 56.7  | 52.9 | 61.1 |
| VGG+GPR*        | Point           | 52.5  | 55.8 | 49.6 |
| RAZ_Loe*        | Point           | 59.8  | 66.6 | 54.3 |
| Crowd-SDNet     | Point           | 63.7  | 65.1 | 62.4 |
| SCALNet (ours)  | Point           | 69.1  | 69.2 | 69.0 |

Table 2. Comparison with the state-of-the-art crowd localization algorithms on the NWPU-Crowd test set. The results are calculated under threshold $\sigma_l$. "*" means the results are provided by NWPU-Crowd.
and localization heads. The third row of Fig. 3 shows the density maps and headcounts predicted by our approach.

### 3.3. Ablation study

We conducted the ablation studies to evaluate the effectiveness of the proposed modules, including the backbone with the stride of 2, and the counting and localization head networks (CHN and LHN). Due to the limitation of submissions on the NWPU-Crowd test set, we present the counting and localization results on the validation set, as shown in Table 4.

#### Backbone with the stride of 2

Our backbone modified the stride of 4 (S4) in DLA-34 to 2 (S2). This modification can retain more spatial information beneficial to small and crowded object detection. From the 1st and 4th entries of Table 4, we can observe that our modification improves the F1-measure by 3.5% and 4.6% under threshold $\sigma_l$ and $\sigma_s$, respectively. Moreover, the inference time of S4 and S2 is 25ms and 29ms per image (with the 1600-pixel longer side), respectively, implying that such a modification only takes a little inference time increase (4ms).

**CHN and LHN.** We separately test the effectiveness of the counting head and localization head networks. With the CHN only (see the 2nd entry), the crowd counting errors increase significantly compared with the model with both heads (see the 4th entry), which illustrates that the localization head provides complementary features for the counting task, reducing the errors (e.g., the MAE drops from 81.8 to 64.4). Besides, the localization performance of the LHN only (see the 3rd entry) is competitive with that of both heads, which shows that the counting head affects the localization task marginally. By comparing the 2nd, 3rd, and 4th entries, we conclude that integrating the CHN and LHN into an end-to-end dense prediction network and simultaneously performing the two tasks not only retains the localization performance but also improves the counting ability.
4. CONCLUSION

This paper has presented a dense point prediction method for crowd counting and localization. With the simple yet effective modified DLA, counting and localization head networks, and the corresponding loss functions, the model achieved outstanding performance compared with state-of-the-art methods. We believe our method can be a baseline for further research. One valuable future research is to study an image-aware or object-aware threshold to select the predicted keypoints rather than the simple thresholding process, which will benefit crowd localization performance.

5. REFERENCES

[1] Guangshuai Gao, Junyu Gao, Qingjie Liu, Qi Wang, and Yunhong Wang, “Cnn-based density estimation and crowd counting: A survey,” arXiv preprint arXiv:2003.12783, 2020.

[2] C. Liu, X. Weng, and Y. Mu, “Recurrent attentive zooming for joint crowd counting and precise localization,” in CVPR, 2019.

[3] Y. Zhang, D. Zhou, S. Chen, S. Gao, and Y. Ma, “Single-image crowd counting via multi-column convolutional neural network,” in CVPR, 2016.

[4] Lingbo Liu, Zhilin Qiu, Guanbin Li, Shufan Liu, Wanli Ouyang, and Liang Lin, “Crowd counting with deep structured scale integration network,” ICCV, 2019.

[5] Yi Wang, Junhui Hou, and Lap-Pui Chau, “Object counting in video surveillance using multi-scale density map regression,” in ICASSP, 2019, pp. 2422–2426.

[6] Yuhong Li, Xiaofan Zhang, and Deming Chen, “Csrnet: Dilated convolutional neural networks for understanding the highly congested scenes,” in CVPR, 2018.

[7] Karen Simonyan and Andrew Zisserman, “Very deep convolutional networks for large-scale image recognition,” in ICLR, 2014.

[8] Junyu Gao, Qi Wang, and Yuan Yuan, “Scar: Spatial-channel-wise attention regression networks for crowd counting,” Neurocomputing, vol. 363, pp. 1–8, 2019.

[9] Ying Chen, Chengying Gao, Zhuo Su, Xiangjian He, and Ning Liu, “Scale-aware rolling fusion network for crowd counting,” in ICME, 2020.

[10] Xinkun Cao, Zhipeng Wang, Yanyun Zhao, and Fei Su, “Scale aggregation network for accurate and efficient crowd counting,” in ECCV, 2018.

[11] Weizhe Liu, Mathieu Salzmann, and Pascal Fua, “Context-aware crowd counting,” in CVPR, 2019.

[12] Zhiheng Ma, Xing Wei, Xiaopeng Hong, and Yihong Gong, “Bayesian loss for crowd count estimation with point supervision,” in ICCV, 2019.

[13] Qi Wang, Junyu Gao, Wei Lin, and Yuan Yuan, “Learning from synthetic data for crowd counting in the wild,” in CVPR, 2019.

[14] Qi Wang, Junyu Gao, Wei Lin, and Xuelong Li, “Nwpu-crowd: A large-scale benchmark for crowd counting and localization,” IEEE Trans. Pattern Anal. Mach. Intell., 2020.

[15] S. Ren, K. He, R. Girshick, and J. Sun, “Faster r-cnn: Towards real-time object detection with region proposal networks,” IEEE Trans. Pattern Anal. Mach. Intell., vol. 39, no. 6, pp. 1137–1149, 2017.

[16] Peiyun Hu and Deva Ramanan, “Finding tiny faces,” in CVPR, 2017.

[17] Issam H Laradji, Negar Rostamzadeh, Pedro O Pinheiro, David Vazquez, and Mark Schmidt, “Where are the blobs: Counting by localization with point supervision,” in ECCV, 2018.

[18] Y. Liu, M. Shi, Q. Zhao, and X. Wang, “Point in, box out: Beyond counting persons in crowds,” in CVPR, 2019.

[19] D. B. Sam, S. Vishwanath Peri, N. S. Mukuntha, and R. Venkatesh Babu, “Going beyond the regression paradigm with accurate dot prediction for dense crowds,” in WACV, 2020.

[20] Deepak Babu Sam, Skand Vishwanath Peri, Mukuntha Narayanan Sundararaman, Amogh Kamath, and R. Venkatesh Babu, “Locate, size and count: Accurately resolving people in dense crowds via detection,” IEEE Trans. Pattern Anal. Mach. Intell., 2020.

[21] Yi Wang, Junhui Hou, Xinyu Hou, and Lap-Pui Chau, “A self-training approach for point-supervised object detection and counting in crowds,” IEEE Trans. Image Process., vol. 30, pp. 2876–2887, 2021.

[22] Haroon Idrees, Muhammd Tayyab, Kishan Athrey, Dong Zhang, Somaya Al-Maadeed, Nasir Rajpoot, and Mubarak Shah, “Composition loss for counting, density map estimation and localization in dense crowds,” in ECCV, 2018, pp. 532–546.

[23] Junyu Gao, Tao Han, Qi Wang, and Yuan Yuan, “Domain-adaptive crowd counting via inter-domain features segregation and gaussian-prior reconstruction,” arXiv preprint arXiv:1912.03677, 2019.

[24] Guosheng Lin, Fayao Liu, Anton Milan, Chunhua Shen, and Ian Reid, “Refinenet: Multi-path refinement networks for dense prediction,” IEEE Trans. Pattern Anal. Mach. Intell., vol. 42, no. 5, pp. 1228–1242, 2019.

[25] Xingyi Zhou, Dequan Wang, and Philipp Krähenbühl, “Objects as points,” in arXiv preprint arXiv:1904.07850, 2019.

[26] F. Yu, D. Wang, E. Shelhamer, and T. Darrell, “Deep layer aggregation,” in CVPR, 2018.

[27] T. Lin, P. Goyal, R. Girshick, K. He, and P. Dollár, “Focal loss for dense object detection,” IEEE Trans. Pattern Anal. Mach. Intell., vol. 42, no. 2, pp. 318–327, 2020.