Supplementary Material of Improving Point-based Crowd Counting and Localization Based on Auxiliary Point Guidance

I-Hsiang Chen[1], Wei-Ting Chen[1,2], Yu-Wei Liu[1], Ming-Hsuan Yang[2,3], and Sy-Yen Kuo[1,4]

[1] National Taiwan University, Taiwan
[2] University of California at Merced, USA
[3] Google, USA
[4] Chang Gung University, Taiwan
{f09921058,f05943089,r12943109}@ntu.edu.tw, mhyang@ucmerced.edu, sykuo@ntu.edu.tw

1 More Experimental Results

1.1 Proposal Selection Analysis

We validate the changes in every proposal considered as part of the crowd and its offset after implementing Auxiliary Point Guidance (APG). By providing auxiliary positive and negative points, APG offers a more defined learning target for proposals, encouraging the model to favor those closer to the actual ground truth points. This not only concentrates each matched proposal, identified as part of the crowd, more around the actual ground truth, preventing conflicts where different proposals match the same ground truth but also ensures that the overall selected proposals are closer to the target heads (with a smaller average offset distance), as illustrated in Figure 1, where dots represent the original positions of proposals and arrows indicate the offsets. The proposed APG simplifies and enhances the training of offset predictions and confidence levels, thereby achieving improved counting performance.

Furthermore, for samples with significant viewpoint differences or occlusions, not utilizing APG might result in multiple points mapping to the same head, leading to overestimation. Overestimation is particularly likely in situations with significant viewpoint differences (as indicated by the red circles in the first row of Figure 2 without using APG) and arm occlusions (as shown by the red circles in the second row of Figure 2 without using APG). In contrast, APGCC offers strong cues for selecting the nearest proposal and avoiding farther ones (indicated by the green circles), thereby playing a crucial guiding role in the learning process of point-based methods. This ensures a more accurate and conflict-free proposal selection.

Moreover, using the protocol in Dan et al. [6], we evaluate errors in overestimation and underestimation within our counting results. The results are presented in Table 1, which demonstrate improvements in reducing overestimation

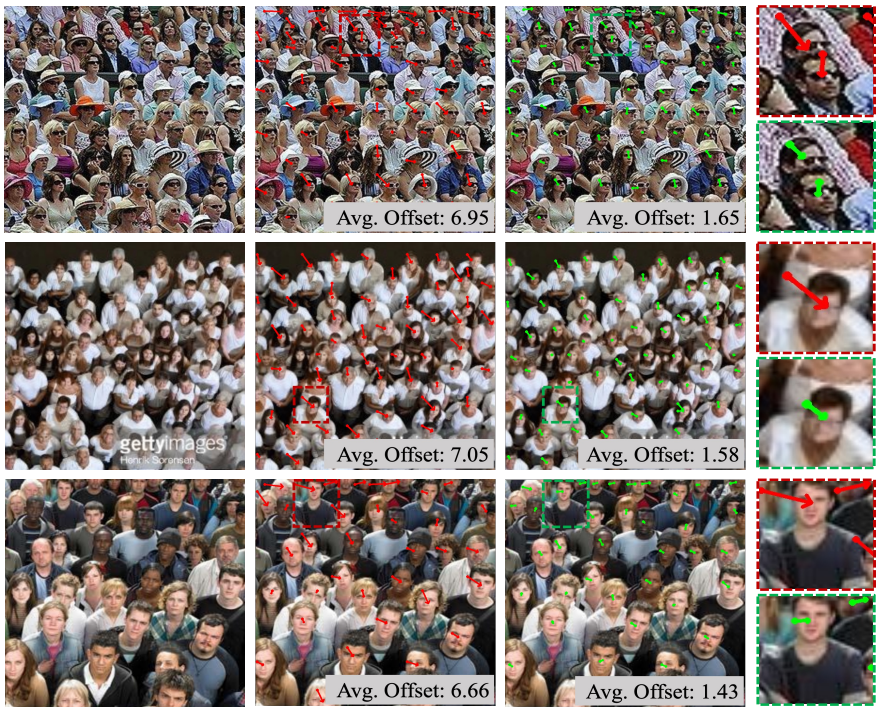

Fig. 1: Visualization of proposal selection. The first column shows input images, the second column depicts models trained without APG using red arrows to mark the proposals, and the third column shows models trained with APG, using green arrows for marking. The fourth column offers a magnified comparison, illustrating that without APG, the selected proposal positions are not on the same person but rather on others or the background, using larger offsets to match the designated individual. In contrast, the proposed APGCC achieves shorter offsets, with the selected proposal positions accurately located on the same person.

using APG. Integrating post-processing techniques such as non-maximum suppression or distance-based filtering could further refine this method for future work.

1.2 Analysis of Crowd Counting Across Various Scales

We further validate the performance of APGCC on crowds of different scales using the NWPU [8] dataset, which includes both large-scale and small-scale crowds (denoted as -L and -S). We compare APGCC with existing methods such as CSRNet [2], Bayesian+ [4], S-DCNet [9], DM-Count [7], and P2PNet [5], with the comparison results shown in Table 2. We observe that APGCC achieves better performance in both small and large scales. This is attributed to our proposed

Table 1: Effectiveness of APG for Misestimation on the SHHA [10] dataset.

Method	Overestimate	Underestimate	Overall
w/o APG	17.68	28.24	54.04
w/ APG	**14.82**	**24.88**	**48.84**

Inputs | w/o APG MAE=54.04 | APGCC MAE=48.84

Fig. 2: Illustration of the overestimation issue in crowd counting. The first row's circles represent individuals' heads from different angles (side or back). In the counting process, point-based methods might count multiple surrounding proposals as pointing to the same individual's head. The second row presents situations with occlusions, where point-based methods also treat multiple proposals as matched to the same individual's head. However, this issue can be avoided by using APG to assist network training.

APGCC architecture, which integrates Implicit Feature Interpolation (IFI) with Atrous Spatial Pyramid Pooling (ASPP) [1], demonstrating significant advancement. By offering more precise feature representation across various scales and locations, APGCC ensures more balanced and enhanced performance for different head sizes.

This enhanced capability is further showcased in scenes with varying densities, as illustrated in Figure 3, where APGCC consistently delivers precise localization and counting across a range of conditions.

Table 2: Comparison of crowd counting across different crowd scales on the NWPU [8] dataset. APGCC demonstrates superior performance on both large scales (MAE-L) and small scales (MAE-S) compared to existing methods, showcasing its efficiency and accuracy in handling crowd counting of varying sizes.

Method	MAE-O ↓	MSE-O ↓	MAE-L ↓	MAE-S ↓
CSRNet	121.3	387.8	112.0	522.7
Bayesian+	105.4	454.2	115.8	750.5
S-DCNet	90.2	370.5	<u>82.9</u>	567.8
DM-Count	88.4	388.6	88.0	<u>498.0</u>
P2PNet	<u>77.4</u>	<u>362.0</u>	83.2	553.9
APGCC	**71.1**	**284.4**	**72.4**	**454.6**

Table 3: Impact of various auxiliary point configurations on training resource usage and model performance on the SHHA [10] dataset with a V100 GPU.

(k_{pos}, k_{neg})	Training Time	GPU Memory	Epoch	Time to Baseline	Time to Best	MAE
(0, 0)	13.73 s/ep	13G	2123	-	8.10 hrs	54.04
(1, 1)	20.91 s/ep	13G	1483	4.54 hrs	8.61 hrs	49.24
(2, 2)	26.88 s/ep	13G	1148	3.32 hrs	8.57 hrs	48.84
(3, 3)	32.81 s/ep	13G	1002	3.34 hrs	9.13 hrs	48.83
(5, 5)	44.58 s/ep	13G	886	3.61 hrs	10.97 hrs	48.81

Table 4: Different Ratio of Auxiliary Point on the SHHA [10] dataset.

(k_{pos}, k_{neg})	w/o APG	(2, 1)	(2, 2)	(2, 3)	(2, 4)
MAE	54.04	49.18	48.84	48.93	49.84

1.3 Effect of Auxiliary Point Quantities on Training and Performance

We investigate the impact of utilizing different quantities of auxiliary points on training resources and the ultimate performance. To this end, a series of experiments are conducted on a V100 GPU, based on the SHHA [10] dataset with a batch size of 32, comparing the GPU memory usage and training time across various auxiliary point configurations. These results are summarized in Table 3. The results reveal that while increasing the number of auxiliary points leads to longer training time per epoch, it can improve the convergence speed and allow the model to surpass the baseline performance in a shorter time. Moreover, the model can perform better as the number of training epochs increases. Further experimental results as Table 4 shown, using an equal number of positive and negative auxiliary point (2, 2) achieves the best balance between training efficiency and model performance. This balanced approach demonstrates APGCC's capability to improve model performance without imposing excessive computational burdens, emphasizing the importance of optimizing the number of auxiliary points for efficient and effective training. On the other hand, since the

Table 5: Comparison of different randomness ranges for auxiliary points.

(n_{pos}, n_{neg})	(1, 8)	(2, 4)	(2, 8)	(3, 8)	(2, 12)
MAE	49.24	49.07	48.84	50.17	49.73

Table 6: Impact of different randomness ranges on various head scales.

(n_{pos}, n_{neg})	(2, 4)	(2, 8)	(2, 12)	(2, 16)
Small Heads	137.60	145.75	158.45	171.16
Large Heads	23.22	21.79	21.09	19.03
Overall	57.77	56.43	58.76	61.91

auxiliary points are only used during the training stage, they do not introduce additional computational costs during the inference process.

1.4 Analysis of various Randomness Range on APG

For the choice of (n_{pos}, n_{neg}), its values are closely related to the distribution (i.e., distance) of point proposals. Similar to P2PNet [5], we adopted a grid layout strategy for mapping proposals, where each pixel on the feature map corresponds to an "s×s" sized area at the image level, evenly divided into "k" parts. For instance, with a stride of s=8 and a reference point count of k=4, each point proposal effectively corresponds to a 4x4 area at the image level. Consequently, we define $n_{pos} = 2$ to maximize the spatial range for selecting positive points (±2), without involving other proposal areas. Additionally, as each proposal's length is 4, we adjusted the setting of n_{neg} to be four times this, creating various coverage ranges. The results are shown as Table 5, if the coverage range is too small, the effect is limited; if set too high, it may interfere with selecting other positive points. Thus, we use the setting (2, 8) to achieve the best balance and effectiveness.

Furthermore, to analyze the dependency of data variability on randomness range. We use bounding box labels from the validation set of NWPU dataset [8] to perform a statistical analysis of head sizes, selecting the smallest and largest 20% of samples as extreme values for ablation studies. Experimental results as the Table 6 shown, the configuration $(2, 8)$ yields the best overall performance. While a smaller n_{neg} enhances stability in areas with small heads, it may neglect to duplicate predictions for larger heads. Conversely, a larger n_{neg} effectively mitigates this issue but can adversely affect predictions in dense areas. Dynamically adjusting the randomness range will be part of our future work.

1.5 Detail of Implicit Feature Interpolation

For any arbitrary image-level coordinate $p_i(x, y)$, it is first converted into the feature-level coordinate $p_f(x, y)$ by dividing by the stride. Using this transformed coordinate, we identify the nearest four latent features $Z_i^* | i \in \{1, \ldots, 4\}$ from the feature map and calculate their corresponding Euclidean distances. To extract features more accurately, we introduce a 48-dimensional periodic spatial encoding $\phi(p)$ that extends beyond the conventional 2D distance map. A Multi-Layer Perceptron is then used to perform continuous transformations of these latent features, integrating them using bilinear interpolation, as detailed in (7).

Table 7: Analysis of Different Interpolation Strategies with Various Strides on the SHHA [10] Dataset.

	MAE (Δ with IFI)			
Stride S	2	4	8	16
Nearest Neighbor	58.63 (+5.92)	55.61 (+4.71)	53.16 (+4.32)	55.45 (+4.13)
Bilinear Interpolation	55.98 (+3.27)	53.52 (+2.62)	51.25 (+2.41)	53.59 (+2.27)
IFI	**52.71**	**50.90**	**48.84**	**51.32**

Table 8: Effectiveness of APG for other point-based methods on the SHHA [10] dataset.

	P2PNet [43]		CLTR [20]		APGCC
Metric	w/o APG	w/ APG	w/o APG	w/ APG	
MAE	53.1	50.4	58.2	54.7	**48.8**
MSE	85.3	80.1	97.6	87.3	**76.7**
Instability Rate (IR)	0.71	0.44	0.82	0.47	**0.36**

The Table 7 evaluate the effectiveness of different interpolation strategies by measuring their MAE across various stride settings. The results show that IFI consistently outperforms other interpolation methods at different strides. Note that smaller strides require more precise feature interpolation to achieve accurate results.

1.6 Applying APG to other point-based methods

We apply APG to existing point-based methods, including P2PNet [5] and CLTR [3]. With the results shown in Table 8, using APG enhances the stability (IR) and performs better (MAE and MSE) across different methods.

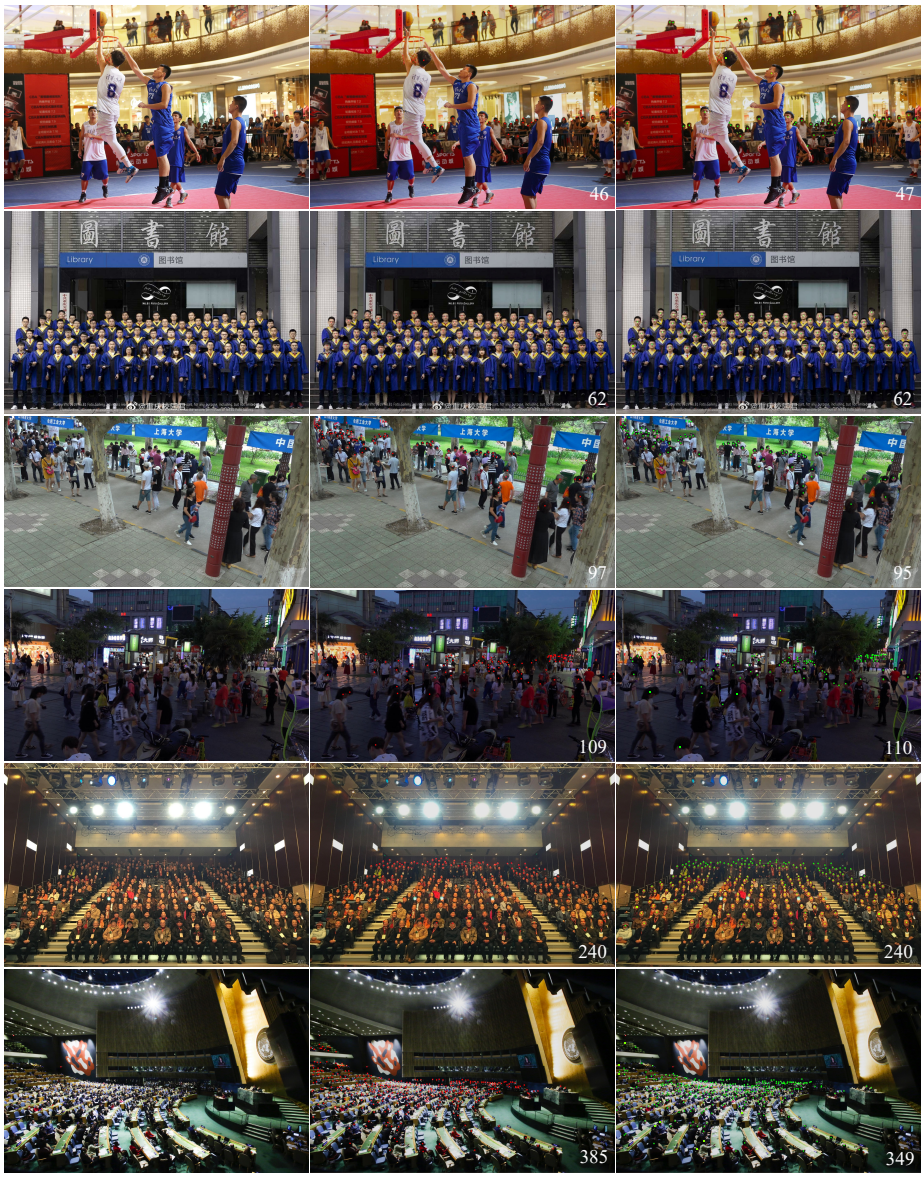

Fig. (3): Localization and counting results in low-density scenes (0-400 people). This figure demonstrates the accuracy of APGCC in scenarios with relatively few individuals, showcasing the model's ability to capture details in sparse environments.

Fig. (3): Localization and counting results in medium-density scenes (400-1000 people). Here, we highlight the model's effectiveness in accurately estimating head counts in moderately crowded settings, emphasizing its robustness across varying population densities. (continued)

Fig. (3): Localization and counting results in high-density scenes (over 1000 people). This figure illustrates APGCC's superior performance in densely populated areas, proving its scalability and precision in handling extreme crowd conditions. (continued)

References

1. Florian, L.C., Adam, S.H.: Rethinking atrous convolution for semantic image segmentation. In: CVPR (2017)
2. Li, Y., Zhang, X., Chen, D.: Csrnet: Dilated convolutional neural networks for understanding the highly congested scenes. In: CVPR (2018)
3. Liang, D., Xu, W., Bai, X.: An end-to-end transformer model for crowd localization. In: ECCV (2022)
4. Ma, Z., Wei, X., Hong, X., Gong, Y.: Bayesian loss for crowd count estimation with point supervision. In: ICCV (2019)
5. Song, Q., Wang, C., Jiang, Z., Wang, Y., Tai, Y., Wang, C., Li, J., Huang, F., Wu, Y.: Rethinking counting and localization in crowds: A purely point-based framework. In: ICCV (2021)
6. Song, Q., Wang, C., Wang, Y., Tai, Y., Wang, C., Li, J., Wu, J., Ma, J.: To choose or to fuse? scale selection for crowd counting. In: AAAI (2021)
7. Wang, B., Liu, H., Samaras, D., Nguyen, M.H.: Distribution matching for crowd counting. NeurIPS (2020)
8. Wang, Q., Gao, J., Lin, W., Li, X.: Nwpu-crowd: A large-scale benchmark for crowd counting and localization. TPAMI (2020)
9. Xiong, H., Lu, H., Liu, C., Liu, L., Cao, Z., Shen, C.: From open set to closed set: Counting objects by spatial divide-and-conquer. In: ICCV (2019)
10. Zhang, Y., Zhou, D., Chen, S., Gao, S., Ma, Y.: Single-image crowd counting via multi-column convolutional neural network. In: CVPR (2016)