Supplementary Material: Interactive Class-Agnostic Object Counting

1. Overview

In the supplementary, we first provide more details about our approach in section 2.1. Then provide more implementation details in section 2.2, analyze the time efficiency and conduct the ablation of the location of the feature refinement module in section 2.4, analyze our method's robustness in 2.3 and analyze the effectiveness of confidence scaling in the other two class-agnostic visual counters in section 2.5. After that, we introduce the interface of our interactive system in section 2.6, give more qualitative results in section 2.7 and section 2.8. Finally, we briefly discuss the limitation and future work in section 2.9. In addition to the supplementary material, we also provide a demo video of our interactive counting system.

2. Supplementary Material

2.1. Additional details for our approach

In this section, we provide additional details for the interaction loop and the IPSE density map segmentation.

2.1.1 Interaction loop

A detailed algorithm for the interaction loop is illustrated in Algorithm 1. The input visual counter contains the following components, a feature extractor f, a spatial-similarity learning module g, layers before the refinement module in regression head h_b, refinement module $\mathcal{R}_{\theta_r}$, and layers after the refinement module in regression head h_a. We need to update Ω with $\boldsymbol{D}$ in each gradient step because the summation over each region depends on the estimated density map.

2.1.2 IPSE density map segmentation

A detailed algorithm for IPSE is illustrated in Algorithm 2. The peak expansion algorithm is shown in Algorithm 3. In Algorithm 2, background splitting is simply expanding at a random background peak with iteratively including the neighbor pixels with the same upper bound, and the small region merging is merging some small region to its neighbor region. More specifically, the region size upper bound T_u is set to 1250, and the region size lower bound T_l in the objective function is set to 250.

Algorithm 1 Interaction loop

Input: Input image: $\mathbf{I}$, Exemplars: $\mathbf{E}$, Gradient steps: N, Adaptation learning rate γ, Interaction times: T.
Initalization: User feedback list: $\Omega = [\,]$.

$\mathbf{S} = g(f(\mathbf{I}), f(\mathbf{E}))$
$\boldsymbol{F} = h_b(\mathbf{S})$
Initialize $\mathcal{R}_{\theta_r}$ correspond to $\boldsymbol{F}$'s size
for T interactions **do**
    $\boldsymbol{D} = h_a(\mathcal{R}_{\theta_r}(\boldsymbol{F}))$
    Visualize $\mathbf{D}$ with IPSE
    Collect user feedback (R, c), Ω.append((R, c))
    $\gamma' = \gamma F_C(\Omega)$, $N' = \frac{N}{F_C(\Omega)}$
    for N' gradient steps **do**
        $\boldsymbol{F'} = \mathcal{R}_{\theta_r}(\boldsymbol{F})$
        $\boldsymbol{D} = h_b(\boldsymbol{F'})$
        Update Ω with $\boldsymbol{D}$
        $\theta_r \leftarrow \theta_r - \gamma' \nabla \mathcal{L}(\Omega)$

Algorithm 2 IPSE Density Map Segmentation Algorithm

Input: Density map: $\boldsymbol{D}$, Smooth kernel: $\boldsymbol{G}$, Objective function:$h(R)$, Region size upper bound: T_u.
Initalization: Foreground region set: $\mathbb{V}_f = \{\ \}$, Background region set: $\mathbb{V}_b = \{\ \}$

$\widetilde{\boldsymbol{D}} \leftarrow \boldsymbol{D} * \boldsymbol{G}$
$S \leftarrow sum(\boldsymbol{D})$
while $S \geq 1$ **do**
    $p = argmax(\widetilde{\boldsymbol{D}})$
    $\widetilde{\boldsymbol{D}}[p] \leftarrow -\infty$
    R = Peak Expansion($\boldsymbol{D}$, $\widetilde{\boldsymbol{D}}$, p, $h(R)$, T_u)
    $\mathbb{V}_f.append(R)$
    $S \leftarrow S - R_s$
$\mathbb{V}_b$ = Background Spliting $(\boldsymbol{D}, \widetilde{\boldsymbol{D}})$
$\mathbb{V} = \mathbb{V}_f \cup \mathbb{V}_b$
$\mathbb{V} \leftarrow$ Small Region Merging($\mathbb{V}$)
return $\mathbb{V}$

2.2. Additional implementation details

FamNet [2]. For FSC-147 [2] we used the released pretrained model. For FSCD-LVIS [1], we train it on one RTX A5000 machine for 150 epochs, and the learning rate is

Algorithm 3 Peak Expansion Algorithm

Input: Density map: $\boldsymbol{D}$, Smooth density map: $\widetilde{\boldsymbol{D}}$, Peak: p, Objective function:$h(R)$, Region size upper bound: T_u.
Initalization: $R_s = 0$, $R_i = 0$, Region pixel list $R_L = [\,]$, Optimal objective value:$P^* = \infty$, Foreground size:$F_i = 0$, Background size:$B_i = 0$.

$\hat{L} = [p]$, $L = [\,]$
while $\hat{L}$ is not empty **and** $R_i < T_u$ **do**
  $\hat{p} = \hat{L}.pop()$, $L.append(\hat{p})$
  for $\hat{p}_n \in \hat{p}$'s neighbour **do**
    if $\hat{p}_n$ not in any regions **then**
      if $\boldsymbol{D}[\hat{p}_n] > 0$ **then**
        $\hat{L}.append(\hat{p})$
        $R_s \leftarrow R_s + D[\hat{p}_n],\ R_i \leftarrow R_i + 1$
        $R_L \leftarrow L + \hat{L},\ F_i \leftarrow F_i + 1$
        $\widetilde{\boldsymbol{D}}[\hat{p}_n] \leftarrow -\infty$
        $P \leftarrow h(R)$
        if $P < P^*$ **then**
          $R^* \leftarrow R,\ P^* \leftarrow P$
      else
        if $F_n > B_n$ **then**
          $\hat{L}.append(\hat{p}_n)$
          $R_s \leftarrow R_s + D[\hat{p}_n],\ R_i \leftarrow R_i + 1$
          $R_L \leftarrow L + \hat{L},\ B_n \leftarrow B_n + 1$
          $\widetilde{\boldsymbol{D}}[\hat{p}_n] \leftarrow -\infty$
return R^*

1×10^{-6}. On FSC-147, following [2], we do the test-time adaptation, on FSCD-LVIS we do not do the test-time adaptation for time efficiency.

SAFECount [5]. For FSC-147 we used the released pre-trained model. For FSCD-LVIS, we train it on one RTX A5000 machine for 100 epochs, and the learning rate is 2×10^{-5}. The interactive adaptation gradient steps are set to 30, and the interactive adaptation learning rate is 0.001.

BMNet+ [3]. For FSC-147 we used the released pre-trained model. For FSCD-LVIS, we train it on one RTX A5000 machine for 100 epochs, and the learning rate is 1×10^{-5}. The interactive adaptation gradient steps are set to 30, and the interactive adaptation learning rate is 0.001.

DM-Count [4]. For ShanghaiTech and UCF-QNRF, we used the released pre-trained model.

2.3. Robustness experiment on crowd counting

Our experiment on crowd counting has demonstrated the effectiveness of our adaptation method from the computational perspective. But from the human perspective, it may be possible that the human user cannot easily provide feedback for the count ranges needed for crowd counting. This is not a concern for the small count limit and the count ranges used in the class-agnostic counting setting given the subitizing ability of humans. But for the count ranges $\{[-\infty, 0], (0, 10], \ldots, (40\ 50], (50\ \infty)\}$ used in this crowd counting experiment, a human user might make estimation mistakes leading to noisy feedback. We therefore perform an experiment to study the robustness of our method to noisy feedback. Specifically, we introduce random biases to the ground truth estimation to simulate mistakes. We consider two estimation biases. For moderate bias, a random noise of 30% of the count limit is added ([-15, 15]). For large bias, a random noise of 50% is added([-25, 25]). With a biased estimation, our approach still can reduce the MAE by approximately 30% on ShanghaiTech A and 10% on the other two datasets, as shown in Table 1.

		Level of feedback noise		
Dataset	Initial error	None	Moderate	Large
ShanghaiTech A	59.60	33.85±0.78	36.15±0.99	40.93±0.65
UCF-QNRF	85.65	58.13±1.04	78.72±2.36	78.14±1.34

Table 1. MAE of the proposed interactive counting method for different levels of feedback noise.

2.4. Additional ablation study

All additional ablation study is conduct on FSC-147 validation set or FSCD-LVIS validation set with FamNet as the visual counter.

2.4.1 Time efficiency analysis

Table 2 shows the time efficiency comparison with vanilla adaptation(Adapt the whole regression head). In Table 2 the average adaptation time(second) for one single click is reported. This experiment is run on RTX A5000, for FSC-147 both of them use 10 gradient steps for one adaptation, and for FSCD-LVIS is 20. We find that our approach is 11.88% faster than vanilla adaptation on FSC-147, and 8.92% faster on FSCD-LVIS. Our method is faster because our method requires less computation in feedforward, backpropagation, and parameter updating, as illustrated in Algorithm 1. In feedforward, we only need to compute the layer before the refinement module one time, and in backpropagation, we only need to compute the gradient for the layers after the refinement module. Also, we only need to update the parameters in the refinement module.

2.4.2 Location of the refinement module.

The ablation of the location of the refinement module is shown in Table 3. This experiment is conduct on FSC-147 validation set. Correlation map means directly refine the spatial correlation map between the exemplar and the input image. We can find that inserting at the shallow position

Benchmarks	FSC-147 Val	FSCD-LVIS Val
Vanilla Adaptation	0.179±0.00	0.56±0.04
Refinement Module	0.160±0.00	0.51±0.00

Table 2. Average adaptation time(second) for one single interaction. The mean and the standard error of five experiments with different seeds are reported.

Component	MAE	RMSE
Correlation map	18.71±0.78	64.15±9.69
After first conv	**12.79±0.16**	**47.21±2.05**
After second conv	13.63±0.34	48.11±3.95
After third conv	13.87±0.13	51.56±1.62

Table 3. Results of different locations of refinement module on the regression head of FamNet. The mean and the standard error of five experiments with different seeds are reported.

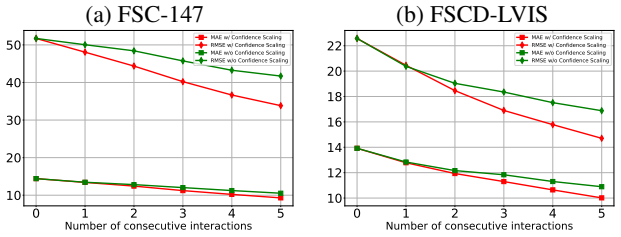


Figure 1. MAE and RMSE with respect to the number of feedback iterations on **SAFECount**. We find that confidence scaling can make the adaptation smoother and improve the final result significantly.

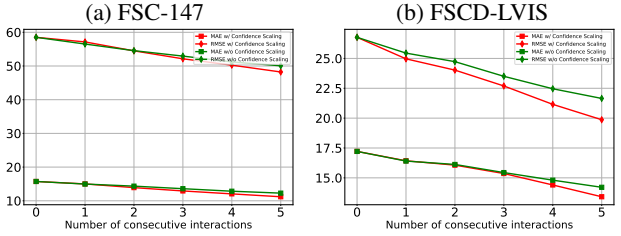


Figure 2. MAE and RMSE with respect to the number of feedback iterations on **BMNet+**. We find that confidence scaling can make the adaptation smoother and improve the final result significantly.

has better performance, and directly refining the correlation map doesn't work well.

2.5. Additional analysis on confidence scaling

In the ablation of the adaptation loss in our main paper, confidence scaling seems less important on FamNet. To further analyze its effectiveness, we conduct additional analysis of confidence scaling on SAFECount and BMNet+. As shown in Fig. 1 and Fig. 2 We can find that confidence scaling can make the adaptation smoother and improve the final result significantly.

2.6. Interactive Interface

The frontend interface of our interactive software is shown Fig. 3. In the visualization, We also provide approximate locations of the detected objects by putting some dots in the regions. The locations of these dots are found automatically, by iteratively selecting a peak of the density map and performing non-maximum suppression for the neighboring pixels. We also provide a demo video in the supplementary. In the demo video the running time for each interaction is around two seconds. This is because in the demo video one interaction includes four stages: adaptation, density map display, segmentation, and visualizing the final result(overlay the image with region boundary and approximate location for each counted object). Analysis of these stages, using images from our user study with three interactions each, shows a mean interaction time of 2.07 seconds. Breakdown: adaptation 0.52s, map display 0.50s, segmentation 0.40s, visualizing the final result 0.64s. Although segmentation takes less than a second, the full process lasts over two seconds due to the image save-load-visualize process. We aim to optimize our software for increased speed in the fut.

2.7. Qualitative results of refinement module.

The qualitative results of the feature refinement are shown in Fig. 4. In this figure, for each example, the first row shows the initial result, and the second row shows the result after one interaction. In each row, we show the prediction, the estimated density map, the refined feature map, and the scale parameters in the refinement module. From the last three columns, we can find that the spatial-wise refinement focuses on the local error that only the parameters close to the region are updated. Thus the spatial-wise refinement contributes more to the refinement of local error. We also find that channel-wise refinement can refine the feature map globally and can correct the global error. This also explains why the channel-wise refinement contributes more to the final result, as illustrated in the refinement module's ablation study in the main paper.

2.8. Additional Qualitative results.

Additional qualitative results on FSC-147 with FamNet is shown in Fig. 5 and Fig. 6.

2.9. Limitation and future work

Our approach has several limitations. First, the user's feedback is for the entire region, not individual objects. Second, the specified count is a range, not a precise number. Third, local adaptation may improve global error, due to the inconsistency between local and global errors. Despite these limitations, the proposed method provides a practical way for the user to provide feedback and reduce counting

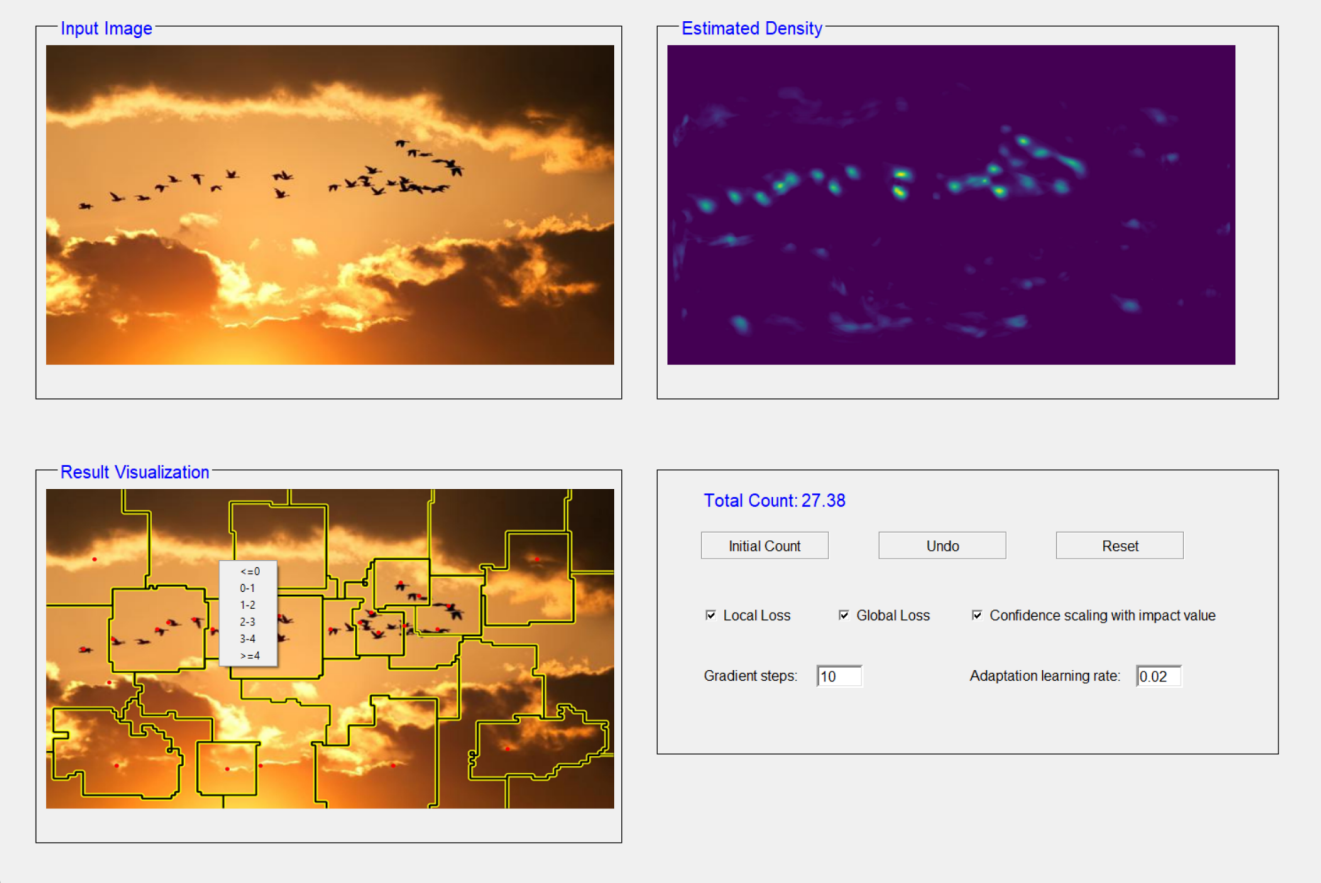


Figure 3. An illustrative graphical user interface for displaying results and collecting the user's feedback. The image is segmented into smaller regions, each region has a moderate size and small density sum. The user can provide feedback by clicking a region and selecting the count range for that region; a total of two clicks per iteration.

errors in most cases. Also important is the availability of an intuitive graphical user interface for the user to decide whether to trust the automated counting results before and after the adaptation.

In this work, we aim for a system that reduces the user's burden so that the user is not asked to delineate or localize objects. But we envision that localizing an object and delineating its spatial extent would be a stronger form of supervision, and it would be necessary for certain situations. This will be explored in our future work.

References

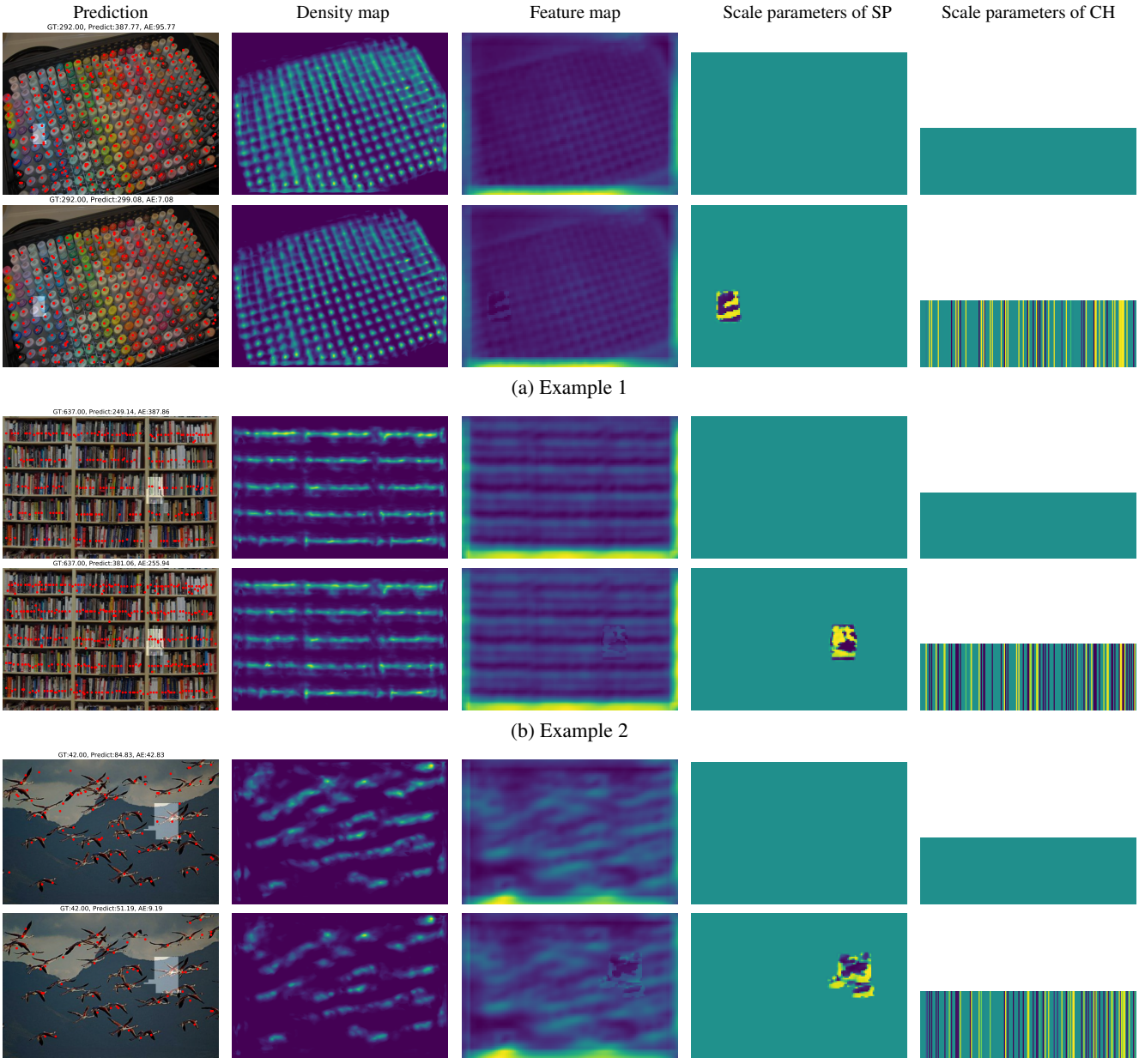
[1] Thanh Nguyen, Chau Pham, Khoi Nguyen, and Minh Hoai. Few-shot object counting and detection. In *Proceedings of the European Conference on Computer Vision*, 2022.

[2] Viresh Ranjan, Udbhav Sharma, Thu Nguyen, and Minh Hoai. Learning to count everything. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2021.

[3] Min Shi, Hao Lu, Chen Feng, Chengxin Liu, and Zhiguo Cao. Represent, compare, and learn: A similarity-aware framework for class-agnostic counting. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2022.

[4] Boyu Wang, Huidong Liu, Dimitris Samaras, and Minh Hoai. Distribution matching for crowd counting. In *Advances in Neural Information Processing Systems*, 2020.

[5] Zhiyuan You, Kai Yang, Wenhan Luo, Xin Lu, Lei Cui, and Xinyi Le. Few-shot object counting with similarity-aware feature enhancement. In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision*, 2023.



(a) Example 1

(b) Example 2

(c) Example 3

Figure 4. Qualitative results of the feature refinement. These three examples are from FSC-147 with FamNet as the visual counter. For each example, the first row is before having any interactive feedback, and the second row is after utilizing one interactive feedback. The first column shows the ground truth, prediction, and absolute error. The second column shows the estimated density map. The third column shows the feature map that the refinement module refines. We can know how the feature refinement refines the feature map from this column. The last two columns show the scale parameters of spatial-wise refinement and channel-wise refinement in the refinement module.

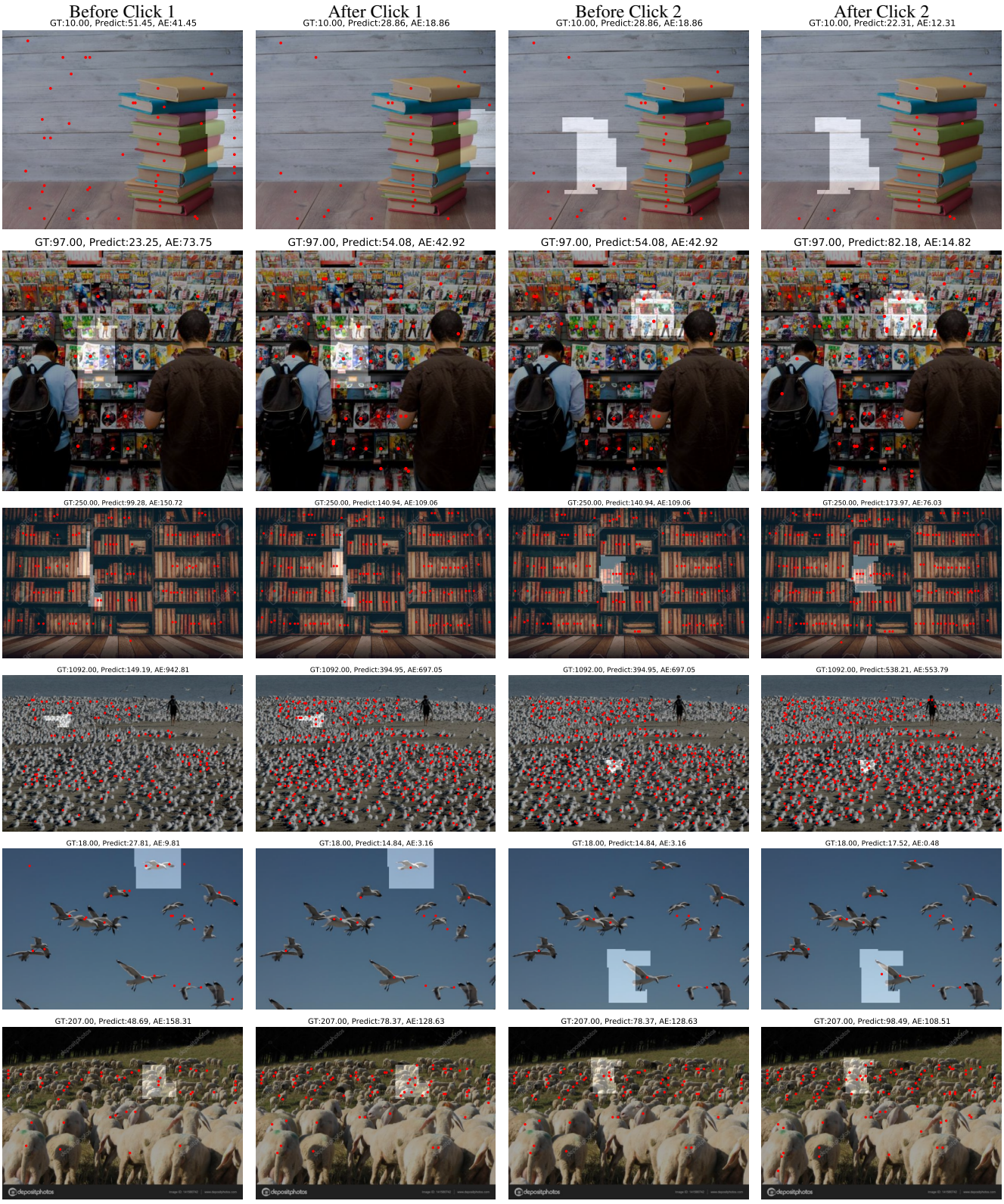


Figure 5. Additional qualitative results. The examples are from FSC-147 with FamNet as the visual counter. The brighter region is the selected region, and the red dot is the approximate location of each region generated by peak selection and non-maximum suppression on each region. Our approach can improve the counting result locally(the selected region) and globally(the whole image).

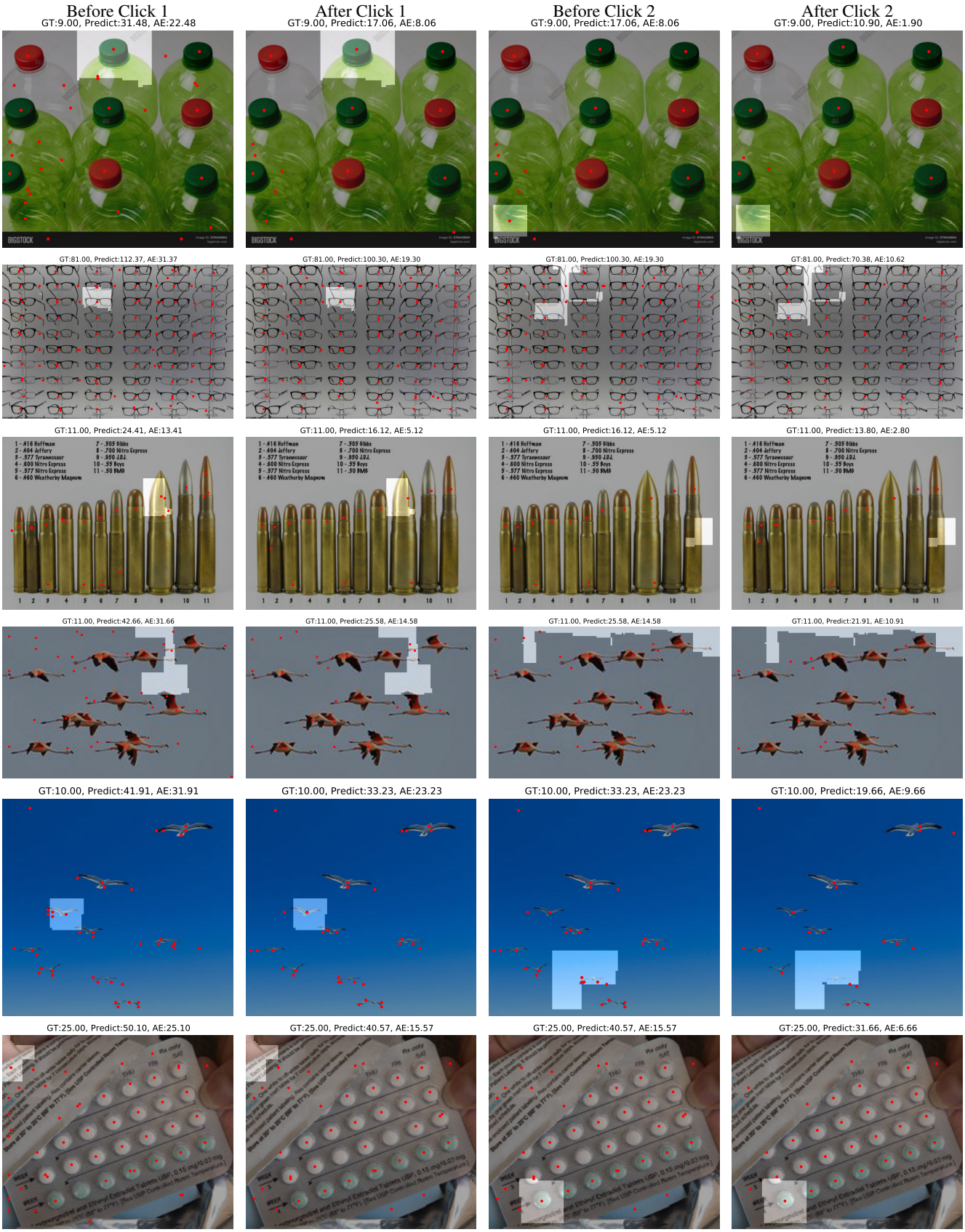


Figure 6. Additional qualitative results. The examples are from FSC-147 with FamNet as the visual counter.