Learning of Counting Crowded Birds of Various Scales via Novel Density Activation Maps

SAEHUN KIM AND MUNCHURL KIM, (Senior Member, IEEE)
Korea Advanced Institute of Science and Technology, Daejeon 34141, South Korea
Corresponding author: Munchurl Kim (mkimee@kaist.ac.kr)

This work was supported by the Ramsar Regional Center-East Asia (RRC-EA) under the project “Automatic Detection and Population Estimation of Birds in Wetland Images using Deep Convolutional Neural Networks”.

ABSTRACT
The previous counting methods trained by the density map regression scheme fail to precisely count the number of birds in crowded bird images of various scales. This is due to the coarseness of the manually created target density maps. In this paper, we propose a new counting scheme, called DAM counting, which generates our-first-proposed density activation map (DAM). DAM is a CNN perspective density map that has high activation values where the network focuses on for precise counting of birds. The network is trained to autonomously learn where and how much the DAM should be activated so that the sum of all values in the DAM estimates the number of birds. Moreover, our DAM counting scheme incorporates two segmentation regularizers that enable precise counting of birds with various scales and appearance. Our DAM counting scheme can effectively substitute the existing density map regression scheme, bringing in a remarkable increase of 45% in the counting accuracy. We also propose the first crowded bird dataset, called CBD-6000, which is very valuable for crowded bird counting research.

INDEX TERMS
Bird counting, density estimation, convolutional neural networks.

I. INTRODUCTION
Keeping track of the migration and population of birds is an essential scientific resource that can be used to monitor the ecosystem and identify potential problems of the climate, which can directly affect the human environment. So far, in the field of counting crowded birds, methods that rely on human observations are used, which require high costs but yield low precision.

Meanwhile, deep learning methods were applied to the detection of birds where they mostly adopt the RCNN [6] to detect birds in various circumstances with a bounding box surrounding each bird individually [2], [8], [20], [23], [24], [28]. However, their works are limited to situations where the birds are very sparse since the accuracy of bounding box detection methods decreases sharply when the objects get crowded.

For crowded object counting, many studies have focused on crowd counting, where they usually incorporate the density map regression scheme [9], [11], [13]–[15], [22]. The density map regression scheme trains the network to regress a manually created target density map by the MSE loss between the target density map and the output density map. For crowd counting, the target density map is constructed by placing a 2D Gaussian kernel centered on each dot location that always points to the head of a person. Since one Gaussian kernel sums up to the value one, the global count is obtained by summing up all the values inside the density map. However, when constructing target density maps for birds, using the circular 2D Gaussian kernel is not appropriate due to the random occlusion of bird parts (Fig. 1). Moreover, the fixed-width of the Gaussian kernel is insufficient to model the various scales of the birds in images.

Arteta et al. [1] was the first to apply the density map regression scheme in counting crowded penguins. Their network is denoted as PenNet for comparison in this paper. To solve the problem of using 2D Gaussian kernels for birds, they constructed a uniform target density map by firstly estimating the penguin regions and secondly assigning a uniform value to each pixel inside a connected penguin region where the value is the number of penguins inside the connected region divided by the number of pixels composing the region (Fig. 2-(d)). This way, the ground truth global count is equal to the sum of all the values in the uniform target density map. The PenNet is trained to regress this uniform target density map. However, the dataset proposed in [1] does not provide ground truth segmentation masks. Thus, they initially estimate the size of the penguins by the distribution of...
the multiple-dot annotations to roughly obtain the penguin regions, leading to inaccurate uniform target density maps. Moreover, the uniform target density map ignores the local density variation which causes confusion to the network since the pixel-wise MSE loss forces the network to regress the diverse local densities of the birds to uniform values (Fig. 2-(e)).

Since the target density maps are not adequate to properly model the density of birds, we propose a new counting scheme that does not require a target density map but outperforms the density map regression scheme. The contributions of our work are summarized as follows:

- We propose a new counting scheme, called DAM counting, which generates our-first-proposed density activation map (DAM). DAM is a CNN perspective density map that has high activation values where the network focuses on for precise counting of birds. The network is trained to autonomously learn where and how much the DAM should be activated so that the sum of all values in the DAM estimates the number of birds. DAM counting can effectively substitute the existing density map regression scheme, bringing a remarkable increase in counting accuracy;

- Our DAM counting scheme enables the network to count different birds of various scales via its inner two (strong and weak) segmentation regularizers, which effectively incorporates manually-created ground truth segmentation masks. Note that we are the first to apply ground truth segmentation masks of birds while the PenNet [1] predicts a rough segmentation mask by the distribution of the multiple-dot annotations, yielding imprecise counting;

- We propose DAM count loss that trains the network to autonomously generate DAMs, in which the sum of all values in the DAM leads to the precise estimation of the global count. Note that the count loss used in the previous models [7], [10], [19], [25], [27] train the network to regress the global counts directly via a fully connected network, which is known to be difficult to train due to the direct mapping from 2-d input images to 1-d global counts;

- We are the first to address the crowded bird counting problem with various scales of different bird species by proposing the first crowded bird dataset, CBD-6000, which includes 6,477 bird images for seven different species of birds where the maximum number of birds is 614 (Fig. 1).

II. RELATED WORK

A. BIRD COUNTING

Currently, in the business of bird counting, traditional bird counting methods are still being used to count crowded birds [5]. Since counting the birds individually in images is a very time-consuming job, they use techniques like grouping and grid counting, which are performed by highly trained humans (birders). Grouping is a technique where a birder counts five or ten at a time. Grid counting is a technique where a birder divides the scene into grids of uniform size, count the number of birds in one grid and multiply the count by the number of grids. However, these methods tend to have low precision due to the rough counting by birders.

Meanwhile, deep learning methods were developed to detect birds automatically. These studies [2], [8], [18], [20], [24] mostly used variations of RCNN [6] to detect birds by surrounding each bird with a bounding box. The number of birds can be obtained by counting the generated bounding boxes. Although bounding box detection-based methods perform well in sparse situations, they are not capable of counting birds in highly crowded and occluded scenes.

B. CROWD COUNTING

When counting crowded objects, especially crowded people, regression-based methods are preferred since they do not need to count the objects individually. [22] conceptually divided the regression-based counting methods by their different regression targets into the global count regression scheme and the density map regression scheme.

The global count regression scheme [10], [25], [26] gradually encodes the extracted features of the input image into smaller feature sizes and uses fully connected layers at the end to output a single number. This scheme uses a global count loss that regresses the network output to the global count. However, these methods have a limitation that pooling the entire image into a single number ignores the spatial information of the objects.
While the global count regression scheme transfers a 2D input image to a single number, the density map regression scheme [3], [9], [13]–[15], [21], [30] converts a 2D input image to a 2D density map which contains the spatial information of the objects. The global count is obtained by integrating the whole density map. The density map regression scheme requires a manually created target density map for the regression target. When counting crowded people in images, the target density map is created by placing a fixed-width 2D Gaussian kernel centered on each dot location that always points to the center of the head of each person. People with completely occluded heads are always ignored. The created target density map appropriately correlates to the distribution of human heads since they become most visible when seen from a higher altitude. However, the density of birds cannot be modeled with the fixed-width 2D Gaussian kernel because of the following reasons: Firstly, the prediction for the bird centers is usually difficult since their body is variously shaped and deformable; Secondly, due to the various motions and occlusions of birds, it is hard to consistently annotate the birds on the same part of the body (See the first image in Fig. 1). Thus, the 2D Gaussian kernels will be placed on various regions of the birds, depending on the exposed parts and poses. This can make the networks confused for learning, as shown in Fig. 2-(c). Finally, the shapes of the birds can extremely vary, especially between when the birds are largely scaled (first image in Fig. 1) and when they are scaled small (third image in Fig. 1).

C. DENSITY MAP REGRESSION FOR BIRDS

Arteta et al. [1] first applied the density map regression scheme to count crowded penguins. We denote their network as PenNet. As mentioned before, since penguins are not circular-shaped, they proposed a uniform target density map as the new regression target for the density map regression scheme (Fig. 2-(d)). The construction of the uniform target density map requires segmentation masks indicating the penguin regions. For each pixel in a connected area of the segmentation mask, equal value is assigned where the value is the number of penguins inside the connected area divided by the number of pixels composing the area. However, the dataset used to train the PenNet does not provide manually created segmentation masks. So, a rough segmentation mask is estimated according to the distribution of the multiple-dot annotations as an alternative.

The PenNet is designed to predict the segmentation mask, compute the uniform target density map, and perform density map regression via a multi-task learning framework. The first branch outputs a segmentation mask, and the second branch outputs a density map. The segmentation mask of the first branch is regressed by the rough segmentation mask computed from multiple-annotated dots. Note that the output segmentation mask is also very rough. The uniform target density map is constructed by the output segmentation mask and used as the regression target for the second branch, which performs density map regression. Note that the regression target for density map regression changes every time the output segmentation mask changes. However, as shown in Fig. 2-(d), the uniform target density map ignores the local density variations of the birds. This confuses the network, leading to imprecise counting of the birds (Fig. 2-(e)). In contrast, our proposed DAM counting scheme does not require a target density map. Also, it incorporates manually created ground truth segmentation masks for precise information of the scale and shape of the birds.

FIGURE 2. (a): An input bird image. (b): Target density map constructed by placing 2D Gaussian kernels. (c): Output density map of [15], which is trained by the density map regression scheme using (b) as the target. (d): Uniform target density map. (e): Output density map of [1] when trained by the density map regression scheme using (e) as the target. (f): Density activation map generated by the same network as (e) but trained with our DAM counting scheme.
Zhou et al. [29] proposed a class activation map (CAM) that activates the regions in which the CNN discriminates when classifying images. Inspired by [29], we propose a new counting scheme, called DAM counting, which produces a density activation map (DAM). DAM is a CNN perspective density map that has high activation values where the network focuses on for precise counting of birds. We apply our DAM counting scheme to previous state-of-the-art counting models [1], [12], [15], which were originally trained by the density map regression scheme.

A. OVERALL PIPELINE
Suppose there is a density map regression-based counting network where the network architecture consists of an encoder and decoder. We denote each part of the network as Baseline Encoder and Decoder 3 in Fig. 3. When our DAM counting scheme is applied to this counting network, two more decoder branches (Decoder 1 and Decoder 2) that have the same architecture as Decoder 3 are added. Decoder 1 outputs a segmentation mask while Decoder 2 and Decoder 3 outputs a DAM. Decoder 1 and Decoder 2 involve strong segmentation regularization, which produces a strong DAM. Decoder 3 involves weak segmentation regularization, which produces a weak DAM. Two DAMs are concatenated with the output (low-level features) of Baseline Encoder, which is then fed into the attention module. Inspired by [13], the attention module outputs two probabilistic attention maps, which are used to fuse the two DAMs as a weighted sum, yielding a final DAM. Finally, the global count is computed by summing all the values in the final DAM. Fig. 3 shows the pipeline of our proposed DAM counting scheme.

B. SEGMENTATION REGULARIZATION
We propose two (strong and weak) segmentation regularizers, where the strong and weak segmentation regularizer removes the non-bird regions aggressively and moderately, respectively. The objective is to control the values of non-bird regions in the DAM to zero so that the network focuses on bird regions only. Along with our proposed DAM count loss, the network will autonomously learn to activate bird regions so that the sum of all values in the DAM approaches the ground truth global count. Moreover, the segmentation regularizers incorporate the ground truth segmentation masks, which indicate the exact size and shape of the birds. Therefore, the network is able to take into account the various scales and the diverse species of the birds.

1) STRONG SEGMENTATION REGULARIZATION
As shown in Fig. 3, the strong segmentation regularization consists of Decoder 1 and Decoder 2. Decoder 1 outputs the segmentation mask \( s \), which is regressed by the cross-entropy loss \( \mathcal{L}_{ce} \) between \( s \) and the ground truth segmentation mask \( S \). Note that \( s \) and \( S \) are a binary mask indicating bird regions.
with the value 1 and non-bird regions with the value 0. Decoder 2 produces an intermediate DAM, \( D_{int} \), which contains intermingled bird and non-bird regions. To completely remove the non-bird regions, we perform element-wise multiplication between \( s \) and \( D_{int} \). The resulting DAM that contains the activation values of the bird regions only is denoted as strong DAM, \( D_{strong} \). By training the two decoders to be dedicated for separate tasks (one for segmentation masks and the other for intermediate DAM), a strong DAM can be learned in a robust way. The strong DAM is defined as:

\[
D_{strong} = s \odot D_{int}
\]  

(1)

where \( \odot \) indicates an element-wise multiplication operator. Next, all the values in \( D_{strong} \) are summed up to a number that represents the estimated global count for the input bird image. The DAM count loss between the strong DAM and the ground truth global count is defined as:

\[
L_{count}(D_{strong}, N_{gc}) = \frac{1}{B} \sum_{b}^{B} |N_{b, gc} - \sum_{i}^{W} \sum_{j}^{H} D_{b, strong}^{(i,j)}|
\]

(2)

where, \( B \) is the batch size, \( N_{b, gc} \) is the ground truth global count of the \( b \)-th image and \( D_{b, strong}^{(i,j)} \) is the \((i,j)\) pixel value of the \( H \times W \)-sized DAM of the \( b \)-th image. The double summation in the right-hand side of Eq. 2 refers to the estimated global count. Due to Eq. 1, the DAM count loss \( L_{count} \) is modulated by \( s \) so that only the bird regions of the DAM contribute to the propagation of the errors. Therefore, the network will only focus on activating the bird regions so that the estimated global count approaches \( N_{gc} \). The total loss of strong segmentation regularization is defined as:

\[
L_{strong} = L_{count}(D_{strong}, N_{gc}) + \omega L_{ce}(s, S)
\]

(3)

where we empirically set \( \omega = 1000 \) to match the scale ranges of the two loss functions. Note that this regularization is called strong segmentation regularization because the non-bird regions are completely removed by masking the intermediate DAM with the output segmentation mask \( s \), which aggressively enforces the network to focus on the bird regions.

In general, the strong segmentation regularization works effectively for most of the cases. However, there are two limitations. First, it is highly dependent on the predicted segmentation mask \( s \) of Decoder 1. If the predicted segmentation mask misses a bird region, it will never appear in the resulting strong DAM (the first image of Fig. 6-(b)). For highly occluded birds where it is hard to predict a correct segmentation mask, the missed bird regions may lead to undercounting. Second, for very small-scaled birds, picky activations are expected inside the small bird regions. However, such picky activations are very difficult for the network to learn (the second image of Fig. 6-(b)). To overcome these two limitations, we propose another segmentation regularization, called weak segmentation regularization, which is described in the following subsection.

2) WEAK SEGMENTATION REGULARIZATION

We propose weak segmentation regularization that can alleviate the limitations of the strong segmentation regularization. Weak segmentation regularization incorporates Decoder 3 and generates a weak DAM, \( D_{weak} \), as shown in Fig. 3. Instead of completely deleting the non-bird regions of the strong DAM by directly multiplying the predicted segmentation mask, the network is trained to moderately suppress the values in the non-bird regions of the weak DAM to zero according to the ground truth segmentation mask. For this, the MSE loss between the ground truth segmentation mask and the weak DAM is applied only on the non-bird regions and is given by

\[
L_{MSE}(D_{weak}, S) = \frac{1}{B} \sum_{b}^{B} (1 - S_b)(D_{b, weak} - S_b)^2
\]

(4)

where \( (1 - S_b) \) indicates the inverse of the binary ground truth segmentation mask. The MSE loss is modulated by \( (1 - S_b) \) so that only the non-bird regions contribute to the propagation of the errors.

We combine \( L_{MSE} \), with the DAM count loss in Eq. 2 for the total loss of weak segmentation regularization, which is defined as:

\[
L_{weak} = L_{count}(D_{weak}, N_{gc}) + \omega L_{MSE}(D_{weak}, S)
\]

(5)

where we empirically set \( \omega = 1000 \) to match the scale ranges of the two loss function values. To minimize \( L_{MSE} \), the network is guided to suppress the values of the weak DAM in non-bird regions to zero. Then, to minimize \( L_{count} \), the network will mostly focus on bird regions so that the estimated global count of the weak DAM approaches the ground truth global count.

This regularization is called weak segmentation regularization because the non-bird regions are not aggressively removed like the strong segmentation regularization. Instead, it moderately suppresses the non-bird regions via the modulated MSE loss, which lowers the possibility of missing bird regions in the DAMs. Thus, due to the fusion of the weak DAM and the strong DAM via the attention module, the weak DAM supplements the missed bird regions in the strong DAM, as shown in the first row of Fig. 6. Also, for very small-scaled birds, the moderate suppression of the weak segmentation regularization allows the network to activate some of the non-bird regions near the birds (Fig. 6-(c)). This feature is fused with the strong DAM and appears as weak activation clouds surrounding the strong activations in the final DAM, as shown in Fig. 6-(d). Thus, the weak DAM alleviates the second limitation of the strong segmentation regularization by allowing the network to generate more spread activations rather than peaky activations.

C. ATTENTION MODULE

We finally propose an attention module that fuses the weak DAM with the strong DAM to alleviate the limitations of the strong DAM. The two resulting DAMs, \( D_{strong} \) and \( D_{weak} \),
and the output (low-level features) of Baseline Encoder are concatenated and fed to the attention module. The attention module consists of 4 convolution layers and one softmax layer. The softmax layer outputs two probabilistic attention maps \((K_1,K_2)\) that contain the proportion values indicating how much the two DAMs contribute to the final DAM. The final DAM, \(D_{\text{final}}\), is computed as a weighted sum of the two DAMs in the guidance of the attention maps and defined as:

\[
D_{\text{final}} = \omega L_{\text{strong}} + (1 - \omega) L_{\text{weak}}
\]

The total loss, \(L_{\text{total}}\), of our proposed DAM counting scheme is given by

\[
L_{\text{total}} = L_{\text{strong}} + L_{\text{weak}} + \omega L_{\text{count}}(D_{\text{final}}, N_{gt})
\]

where we empirically set \(\omega = 2\).

### IV. EXPERIMENTS

#### A. CROWDED BIRD DATASET (CBD-6000)

We propose the first crowded bird dataset, called CBD-6000, which has 6,477 photos of 7 bird species taken from Google search. It also includes single dot-annotations of the birds and manually created segmentation masks for bird regions in each image. The dot-annotations are placed on the center of the visible part of each bird. Due to random occlusions of birds, the location of the dots varies. The number of birds ranges from 1 to 614 per image. The 6,067 images are used for training, and the remaining 410 images are used for testing. Specific details for CBD-6000 are shown in Table 1. Note that the dot-annotations are not used in our DAM counting scheme but are provided to construct target density maps, which are required for the existing density map regression scheme.

#### B. IMPLEMENTATION DETAILS

Training images are all resized to 256 × 400 to set a batch size greater than 1. Test images are evaluated at their original sizes. The multiple phases in the pipeline are trained in an end-to-end fashion. Experiments were conducted with Pytorch on the Titan Xp gpu. We denote the previous penguin counting model as PenNet (ECCV 2016) [1] and the state-of-the-art crowd counting models as CSRNet (CVPR 2018) [12] and CAN (CVPR 2019) [15].

The original PenNet [1] is modified to utilize the ground truth segmentation masks when constructing the uniform target density maps, instead of using rough estimates of them. Moreover, the original PenNet relies on an old segmentation network architecture, FCN8s [16]. For more harsh comparison, we substitute the FCN8s with a more recent SOTA segmentation architecture, DeepLabv3+ [4]. The modified PenNet is denoted as PenNet*.

The crowd counting models, CSRNet [12] and CAN [15], place a fixed-width 2D Gaussian kernel on the locations of the dot-annotations to construct a target density map. However, since the bird shapes are not circular in general, such a fixed-width 2D Gaussian kernel is not appropriate. If the uniform target density map were used as the regression target instead, the models would have achieved improved performances. To harshly compare our DAM counting scheme against the density map regression scheme, we additionally implement the two models, CSRNet [12] and CAN [15], so that they use the uniform target density map as the regression target and are denoted as CSRNet\(^*\) and CAN\(^*\), respectively.

In order to access the effectiveness of our proposed DAM counting scheme, it is incorporated (implemented) into the three previous state-of-the-art counting models, PenNet, CSRNet, and CAN, which are denoted as PenNet\(^*\)+DAM, CSRNet+DAM, and CAN+DAM, respectively.

#### C. RESULTS COMPARISON

We use the mean absolute error (MAE) as our evaluation metrics which represents the counting accuracy of the model and is given by

\[
\text{MAE} = \frac{1}{M} \sum_{m} |N_{m,gt} - N_{m,est}|
\]

where \(M\) is the number of test images, \(N_{gt}\) is the ground truth global count, \(N_{est}\) is the estimated global count.

We show the experimental results for each range of the number of birds, as depicted in Table 2. As we can see for all three networks (CSRNet, CAN, PenNet), the accuracy increases significantly when our proposed DAM counting scheme is substituted with the density map regression scheme. Notably, we see a substantial increase in the counting accuracy for the range of 100 ~ 614. In this range, the birds are very small-scaled, which may be modeled with a 2D Gaussian kernel. However, the results show that our method outperforms all the other methods in this range as well. Moreover, compared to a pre-trained Yolov3 detector [17] on our crowded bird dataset, our method also shows better performance even when the birds are sparse and large-scaled (1 ~ 10).

Since manual segmentation masks are expensive to obtain, we performed an experiment with a pre-trained segmentation network (deeplabv3 [4]). We used a fixed-size circle of a 5-pixel radius for birds that deeplabv3 could not detect. The resulting accuracies for the segmentation masks are shown
TABLE 2. Comparative results on our crowded bird dataset, CBD-6000.

| # of birds | 1 ~ 10 | 11 ~ 50 | 51 ~ 100 | 101 ~ 614 | Total |
|------------|--------|---------|----------|----------|-------|
| Yolov3 [18] | 0.86   | 10.93  | 17.85    | 208.89   | 9.56  |
| CSRNet [13] | 2.19   | 7.67   | 24.12    | 126.2    | 6.64  |
| CSRNet*    | 1.33   | 7.11   | 14.65    | 91.22    | 4.72  |
| CSRNet+DAM | 0.73   | 4.87   | 10.00    | 45.44    | 2.71  |
| CAN [16]   | 2.57   | 14.79  | 62.41    | 184.78   | 10.87 |
| CAN*       | 1.23   | 5.33   | 11.29    | 56.67    | 3.47  |
| CAN+DAM    | 0.64   | 4.38   | 10.18    | 40.33    | 2.48  |
| PenNet* [1] | 1.13   | 4.92   | 11.88    | 66.33    | 3.57  |
| PenNet+DAM | 0.59   | 3.48   | 9.24     | 26.89    | 1.95  |

Table 2 shows a comparison between different methods and manual segmentation. The results are comparable, indicating the effectiveness of our approach.

Also, we tested our method on the penguin dataset [1], as shown in Table 4. Even though very rough segmentation masks are used by simply drawing circles of 20-pixel radius around the dot annotations, the counting accuracy of our method is higher than that of the PenNet [1], which implies that our method is robust with cheaply obtained segmentation masks.

TABLE 3. Sensitivity experiment on our crowded bird dataset, CBD-6000.

PenNet* (manual) denotes the baseline network PenNet trained with manual segmentation masks. Ours (deeplabv3) denotes our method trained with a segmentation mask computed by the pre-trained segmentation network, deeplabv3 [4]. Ours (manual) denotes our method trained with manual segmentation masks.

| # of birds | 1 ~ 10 | 11 ~ 50 | 51 ~ 100 | 101 ~ 614 | Total |
|------------|--------|---------|----------|----------|-------|
| PenNet* [1] (manual) | 1.13   | 4.92   | 11.88    | 66.33    | 3.57  |
| Ours (deeplabv3) | 0.72   | 4.31   | 10.47    | 30.44    | 2.31  |
| Ours (manual) | 0.59   | 3.48   | 9.24     | 26.89    | 1.95  |

Table 3 compares the sensitivity of our method to different segmentation masks. The results show that our method is robust to rough segmentation masks.

Fig. 4 compares the generated density maps of the PenNet* [1] and the generated density activation maps of the PenNet+DAM. Visual results show that the PenNet+DAM autonomously learns where to activate from the perspective of precise bird counting. For large-scaled birds in the second image of Fig. 4-(a), it is interesting to see that the DAM is more highly activated in the heads of the birds than their body. This is similar to the human’s counting perspective, where humans will tend to count the head of the birds for this image. Surprisingly, the network autonomously learns this fact without any additional information related to the head regions. On the other hand, even though the PenNet* is trained with a uniform target density map (the right corner image of Fig. 4-(b)), the PenNet+ produces a density map with large variations over the entire bird region. This implies that the network is confused by the uniform target density map, which ignores local density variations (Fig. 2-(d)). Similarly, the third row of Fig. 4 shows that our DAM counting scheme can autonomously learn to activate necessary parts of the birds that lead to precise counting for diverse scenes. For small-scaled birds in the sixth and seventh images of Fig. 4-(a), which are too small to discriminate the head and body regions, the PenNet+DAM additionally activates some of the non-bird regions that are surrounding the small-scaled birds like a cloud. This is a feature fused from the weak DAM to supplement the limitations of the strong DAM by the attention module, which will be discussed in the following ablation study. We can see that the autonomous activation of our DAM counting scheme remarkably increases the counting accuracies compared to the density map regression scheme, which directly regresses the uniform target density map.

D. ABLATION STUDY

In this section, we perform ablation studies to show the limitations of each regularizer and the necessity of the attention module. As shown in Table 5, we can see that the segmentation regularization in our DAM counting scheme is essential, and the fusion via the attention module significantly increases the accuracy for all ranges of the number of birds. The strong segmentation regularizer generally performs better than the weak segmentation regularizer. However, the strong segmentation regularizer shows poor accuracy in the 51 ~ 100 range. The images in this range often have extreme occlusions and are difficult to detect the bird regions correctly. The strong dependence on the correctness of segmentation masks can lead to undercounting when it misses a bird region. The fusion of the strong DAM and the weak DAM via the attention module supplements the missed regions of the strong DAM, as shown in the first row of Fig. 6 which leads to more precise counting. The 101 ~ 614 range also experiences a significant increase in the counting accuracy due to the attention module. As shown in the second row of Fig. 6, the activation in the non-bird regions of the weak DAM is fused with the strong DAM and appears as weak activation clouds surrounding the birds.
FIGURE 4. Visual comparison. (a) Input bird image. (b) Density map generated by PenNet$. The small density map in the corner is the uniform target density map that the network should target. (c) Density activation map generated by PenNet$^+$+DAM.
the strong activations in the final DAM. By generating more spread activations around the peaky activations of the strong DAM, the network is able to count the number of small-scaled birds precisely.

V. CONCLUSION

In this paper, we propose a new counting scheme, called DAM counting, which generates a density activation map (DAM) to count crowded birds of various scales. Most current counting models are trained via the density map regression scheme; however, the regression target density maps are not appropriate in modeling the density of the birds. Our proposed DAM counting scheme does not require target density maps; instead, it allows the network to learn effective activation regions from a CNN perspective for bird counting. In order to handle crowded birds of various scales, strong and weak segmentation regularizers are proposed in conjunction with an attention module. The experimental results show that our proposed DAM counting scheme significantly boosts the counting performances of the existing SOTA counting models. Our proposed DAM counting scheme is expected to be applicable in counting other animals besides birds. Some challenging problems in birds counting include ambiguity of background and birds, unseen patterns of various bird occlusions, out-focused birds, and partial appearance along image boundaries, etc. Our future work will try to solve such challenging problems and to apply our method for other animals.

REFERENCES

[1] C. Arteta, V. Lempitsky, and A. Zisserman, “Counting in the wild,” in Proc. Eur. Conf. Comput. Vis. Cham, Switzerland: Springer, 2016, pp. 483–498.
[2] L. B. Boudaoud, F. Maussang, R. Garello, and A. Chevallier, “Marine bird detection based on deep learning using high-resolution aerial images,” in Proc. OCEANS, Jun. 2019, pp. 1–7.
[3] X. Cao, Z. Wang, Y. Zhao, and F. Su, “Scale aggregation network for accurate and efficient crowd counting,” in Proc. Eur. Conf. Comput. Vis. (ECCV), 2018, pp. 734–750.
[4] L. C. Chen, Y. Zhu, G. Papandreou, F. Schroff, and H. Adam, “Encoder-decoder with atrous separable convolution for semantic image segmentation,” in Proc. Eur. Conf. Comput. Vis. (ECCV), 2018, pp. 801–818.
[5] S. Delany, Guidance on Waterbird Monitoring Methodology: Field Protocol for Waterbird Counting. Wetlands International, 2011.
[6] R. Girshick, J. Donahue, T. Darrell, and J. Malik, “Rich feature hierarchies for accurate object detection and semantic segmentation,” in Proc. IEEE Conf. Comput. Vis. Pattern Recognit., Jun. 2014, pp. 580–587.
[7] K. Han, W. Han, H. Yao, and L. Hou, “Image crowd counting using convolutional neural network and Markov random field,” J. Adv. Comput. Intell. Inform., vol. 21, no. 4, pp. 632–636, Jul. 2017.

[8] S.-J. Hong, Y. Han, S.-Y. Kim, A.-Y. Lee, and G. Kim, “Application of deep-learning methods to bird detection using unmanned aerial vehicle imagery,” Sensors, vol. 19, no. 7, p. 1651, Apr. 2019.

[9] M. Hossain, M. Hosseinazedeh, O. Chanda, and Y. Wang, “Crowd counting using scale-aware attention networks,” in Proc. IEEE Winter Conf. Appl. Comput. Vis. (WACV), Jan. 2019, pp. 1280–1288.

[10] H. Idrees, M. Tayyab, K. Athrey, D. Zhang, S. Al-Maadeed, N. Rajpoot, and M. Shah, “Composition loss for counting, density map estimation and localization in dense crowds,” in Proc. Eur. Conf. Comput. Vis. (ECCV), 2018, pp. 532–546.

[11] V. Lempitsky and A. Zisserman, “Learning to count objects in images,” in Proc. Adv. Neural Inf. Process. Syst., 2010, pp. 1324–1332.

[12] Y. Li, X. Zhang, and D. Chen, “CSRNet: Dilated convolutional neural networks for understanding the highly congested scenes,” in Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit., Jun. 2018, pp. 1091–1100.

[13] J. Liu, C. Gao, D. Meng, and A. G. Hauptmann, “DecideNet: Counting varying density crowds through attention guided detection and density estimation,” in Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit., Jun. 2018, pp. 5197–5206.

[14] L. Liu, Z. Qiu, G. Li, S. Liu, W. Ouyang, and L. Lin, “Crowd counting with deep structured scale integration network,” in Proc. IEEE/CVF Int. Conf. Comput. Vis. (ICCV), Oct. 2019, pp. 1774–1783.

[15] W. Liu, M. Salzmann, and P. Fua, “Context-aware crowd counting,” in Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR), Jun. 2019, pp. 5099–5108.

[16] J. Long, E. Shelhamer, and T. Darrell, “Fully convolutional networks for semantic segmentation,” in Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR), Jun. 2015, pp. 3431–3440.

[17] J. Redmon and A. Farhadi, “YOLOv3: An incremental improvement,” 2018. arXiv:1804.02767. [Online]. Available: http://arxiv.org/abs/1804.02767

[18] K. Reese, Y. Zheng, and A. Elmaghraby, “A comparison of face detection algorithms in visible and thermal spectrums,” in Proc. Int. Conf. Adv. Comput. Sci. Appl., 2012, pp. 1–5.

[19] C. Shang, H. Ai, and B. Bai, “End-to-end crowd counting via joint learning local and global count,” in Proc. IEEE Int. Conf. Image Process. (ICIP), Sep. 2016, pp. 1215–1219.

[20] E. S. Simons and M. K. Hinders, “Automatic counting of birds in a bird deterrence field trial.” Ecol. Evol., vol. 9, no. 20, pp. 11878–11890, Oct. 2019.

[21] V. A. Sindagi and V. M. Patel, “CNN-based cascaded multi-task learning of high-level prior and density estimation for crowd counting,” in Proc. 14th IEEE Int. Conf. Adv. Video Signal Based Survell. (AVSS), Aug. 2017, pp. 1–6.

[22] V. A. Sindagi and V. M. Patel, “A survey of recent advances in CNN-based single image crowd counting and density estimation.” Pattern Recognit. Lett., vol. 107, pp. 3–16, May 2018.

[23] A. Takeki, T. T. Trinh, R. Yoshihashi, R. Kawakami, M. Iida, and T. Naemura, “Detection of small birds in large images by combining a deep detector with semantic segmentation,” in Proc. IEEE Int. Conf. Image Process. (ICIP), Sep. 2016, pp. 3977–3981.

[24] W. Wang, C. Rasmussen, and C. Song, “Fast, deep detection and tracking of birds and nests,” in Proc. Int. Symp. Vis. Comput. Cham, Switzerland: Springer, 2016, pp. 146–155.

[25] B. Wilke, S. Cahyawijaya, and W. Adiprawita, “CountNet: End to end deep learning for crowd counting,” Proc. Electr. Eng. Comput. Sci. Informat., vol. 5, no. 5, pp. 128–132, Nov. 2018.

[26] W. Xie, J. A. Noble, and A. Zisserman, “Microscopy cell counting and detection with fully convolutional regression networks,” Comput. Methods Biomech. Biomed. Eng., Imag. Vis., vol. 6, no. 3, pp. 283–292, May 2018.

[27] Y. Xue, N. Ray, J. Hugh, and G. Bigras, “Cell counting by regression using convolutional neural network,” in Proc. Eur. Conf. Comput. Vis. Cham, Switzerland: Springer, 2016, pp. 274–290.

[28] R. Yoshihashi, T. T. Trinh, R. Kawakami, S. You, M. Iida, and T. Naemura, “Differentiating objects by motion: Joint detection and tracking of small flying objects,” 2017, arXiv:1709.04666. [Online]. Available: http://arxiv.org/abs/1709.04666

[29] B. Zhou, A. Khosla, A. Lapedriza, A. Oliva, and A. Torralba, “Learning deep features for discriminative localization,” in Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR), Jun. 2016, pp. 2921–2929.

[30] L. Zhu, C. Li, Z. Yang, K. Yuan, and S. Wang, “Crowd density estimation based on classification activation map and patch density level,” Neural Comput. Appl., vol. 32, pp. 5105–5116, Jan. 2019.

---

SAEHUN KIM received the B.E. and M.E. degrees from the School of Electrical Engineering, Korea Advanced Institute of Science and Technology (KAIST), Daejeon, South Korea, in 2018 and 2020, respectively. His research interests include computer vision and image processing, such as detection, counting, style transfer, image-to-image translation, and deep learning.

MUNCHURL KIM (Senior Member, IEEE) received the B.E. degree in electronics from Kyungpook National University, Daegu, South Korea, in 1989, and the M.E. and Ph.D. degrees in electrical and computer engineering from the University of Florida, Gainesville, FL, USA, in 1992 and 1996, respectively. He joined the Electronics and Telecommunications Research Institute, Daejeon, South Korea, as a Senior Research Staff Member, where he led the Realistic Broadcasting Media Research Team. He joined the School of Engineering, Information and Communications University (ICU), Daejeon, as an Assistant Professor, in 2001. Since 2009, he has been with the School of Electrical Engineering, Korea Advanced Institute of Science and Technology, Daejeon, where he is currently a Full Professor. He was involved in scalable video coding and high-efficiency video coding in JCT-VC standardization activities of ITU-T VCEG and ISO/IEC MPEG. His current research interests include deep learning for image restoration and visual quality enhancement, deep video compression, perceptual video coding, visual quality assessments, computational photography, machine learning, and pattern recognition.