ATTENTION-BASED DEEP BAYESIAN COUNTING FOR AI-AUGMENTED AGRICULTURE

A Thesis

by

YUCHENG WANG

Submitted to the Office of Graduate and Professional Studies of
Texas A&M University
in partial fulfillment of the requirements for the degree of

MASTER OF SCIENCE

Chair of Committee,	Xiaoning Qian
Committee Members,	Nicholas Duffield
	Byung-Jun Yoon
	Mengmeng Gu
Head of Department,	Miroslav M. Begovic

May 2021

Major Subject: Electrical Engineering

ABSTRACT

Object counting in images has been studied extensively, in particular using deep network models recently. The existing counting models typically output the point estimates of the object counts in given images. However, none of these can provide reliable uncertainty quantification of the derived count estimates, which is critical for consequent decision making when adopting these counting models in real-world applications. In this thesis, we propose a novel deep counting model in a Bayesian framework. With the designed Bayesian attention module and Bayesian counting loss function, our deep Bayesian counting model not only improves the accuracy of count estimates with varying object and background appearance, as well as image quality; but also enables their uncertainty quantification. We specifically focus on plant counting, which plays important roles in AI-augmented agriculture, for example crop yield estimates and field management. Our ablation studies and experiments with the real-world agriculture data, including the Global Wheat dataset, have demonstrated that our deep Bayesian counting model obtains high count estimation accuracy as well as reliable uncertainty quantification. In addition, with the integrated Bayesian attention modules, it may help improve the interpretability of the derived count estimates, especially when the distribution of the interested plants in images is heterogeneous.

DEDICATION

To my mother, my father, my grandfather, and my grandmother.

ACKNOWLEDGMENTS

I would like to thank my advisor, Professor Xiaoning Qian, for his helpful advice and continuous support to my research and writing. I also wish to thank my committee members, Professor Nicholas Duffield, Professor Byung-Jun Yoon and Professor Mengmeng Gu for their valuable advice for my thesis revision.

Thanks also to my teammate Ziyu Xiang, and my friends Hongyu Shen and Ziyi Zhang for their encouragement and help in Texas A&M University.

Lastly, I would like to thank my family for providing financial and moral support during my graduate study.

CONTRIBUTORS AND FUNDING SOURCES

Contributors

This work was supported by a thesis committee consisting of Professor Xiaoning Qian, Professor Nicholas Duffield and Professor Byung-Jun Yoon of the Department of Electrical and Computer Engineering and Professor Mengmeng Gu of the Department of Horticultural Sciences.

The field experiment data was provided by Professor Mengmeng Gu of the Department of Horticultural Sciences and labeled by previous group student Xueting Liu.

All other work conducted for the thesis was completed by the student independently.

Funding Sources

Graduate study was supported by graduate merit scholarship from Department of Electrical and Computer Engineering of Texas A&M University, and National Science Foundation Award IIS-1812641.

NOMENCLATURE

CNN	Convolutional Neural Network
DME	Density Map Estimation
MAE	Mean Absolute Error
MSE	Mean Squared Error
ReLU	Rectified Linear Unit
DNN	Deep Neural Networks
CBAM	Convolutional Block Attention Module
UAV	Unmanned Aerial Vehicles
AI	Artificial Intelegence
ML	Machine Learning
SSIM	Structure Similarity Index Measure
MLP	Multi-Layer Perceptron

TABLE OF CONTENTS

Page

LIST OF FIGURES

FIGURE Page

LIST OF TABLES

TABLE Page

1. INTRODUCTION

Object counting has been drawing more and more attention in computer vision research due to its diverse applications in event detection and daily decision making. For example, reliable object counting can help predicting the density of crowd in rallies [2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16], traffic volume in transportation [17] [18], and cell counts in biomedical microscopy images [19] [20], just name a few. When the objects or events of interest are sparse in scenes, traditional detection models can be applied. When the density is high, however, these traditional models become unsuitable due to the decrease of accuracy as well as speed when applying them for counting. Thanks to the recent research advancements of deep neural networks (DNNs) and end-to-end density-map-estimation (DME) methods based on deep models, it is expected that we may develop more accurate counting methods to deal with images with more complex background appearances and a higher density of objects with promising computation speed.

One key challenge that recent DME methods face is that both the object and background appearances can vary significantly across different images. The variability could exist in object shape, scale, resolution and objects can appear with different background. In crowd counting, for example, the images might be taken in different distance and different angle, causing the changes of object shape, scale, and appearance. If the objects in a specific image are either larger or sparser than the others, often these end-to-end DME methods with DNNs may not be able to adapt due to its limit receptive field, leading the higher estimated count in that area.

This thesis focuses on the application of automated object counting in agriculture applications, where the deployment of drones or unmanned aerial vehicles (UAVs) to monitor growing fields in farms and ranches is becoming commonplace [21, 22, 23]; however, these drone-captured site images also pose unique challenges to accurate and reliable counting due to their relatively more difficult quality control. For example, if the plant in test image is a different subspecies, or in a different growth stage, the model may not be able to produce an accurate prediction.

More critically, in addition to addressing these challenges when analyzing UAV plant images,

one important focus of this thesis is to enable uncertainty quantification in counting. In agriculture, counting is required so that better decision making regarding planting, fertilizing, irrigation, and other farm management for example, can be derived to help minimize the cost and risk while maximizing the potential yields. In addition to accurate count estimates, it is desirable to also have reliable uncertainty estimates so that robust and sustainable decision making can be achieved when uncertainty arises [24] due to possible data noise, abnormal UAV image quality, the limitation of adopted machine learning (AI/ML) models, or when deploying AI/ML models that are trained using different data sources, especially considering the challenges of collecting annotated UAV images with the specific crops or plants of interest to different growing fields, farms, or ranches.

The existing counting models typically output only the point estimates of the object counts in given images. However, none of the existing method, to the best of our knowledge, can provide reliable uncertainty quantification of the derived count estimates, which is critical for consequent decision making when adopting these counting models in real-world applications. In this thesis, we propose a novel deep counting method in a Bayesian framework. To achieve accurate and robust object counting, it is important for the model to tackle challenges in analyzing UAV plant images, especially considering the significant variability in object and background appearance, as well as image quality. What's more, with the designed Bayesian attention module and Bayesian counting loss function, our deep Bayesian counting model not only improves the accuracy of count estimates with varying object and background appearance, as well as image quality; but also enables their uncertainty quantification. With this Bayesian counting model, uncertainty quantification for plant counting may play important roles in AI-augmented agriculture, for example crop yield estimates and field management.

Our ablation studies and experiments with the real-world agriculture data, including the Global Wheat dataset, have demonstrated that our deep Bayesian counting model obtains high count estimation accuracy as well as reliable uncertainty quantification. In addition, with the integrated Bayesian attention modules, it may help improve the interpretability of the derived count estimates, especially when the distribution of the interested plants in images is heterogeneous.

2. LITERATURE REVIEW

In this chapter, we first give a review of literature about the existing object counting models.

2.1 Existing Object Counting Methods

Some of early works [10, 11] in object counting involve the detection or semantic segmentation using handcraft features. Although those methods can give more detailed predictions in terms of object size and location, the performance and speed may degrade as the number of objects in one image increases. Regression-based methods, on the other hand, ignore the location and estimate the number of objects directly [12, 25, 26]. Those methods can achieve better performance on high density imaging data, but they cannot efficiently utilize the labelling information provided by point annotations that are typically done when constructing training sets for objective counting.

Density-map-estimation (DME) methods, proposed first by [20], predict a density map of a given image. The value of each pixel in the density map denotes the estimated probability of having the object in the corresponding image region. We will then calculate the number of objects by summing over the whole density map. In this way, these DME methods can both preserve the location information and deal with images potentially with a high density of objects that may have high overlapping. More recently, the auther of [27] propose DME methods based on convoluational neural network (CNN), and demonstrated its superior performance over traditional object counting methods based on handcrafted features. The counting performance has been further improved to achieve the state-of-the-art performance. Some multi-branch CNN [28, 29] are proposed to capture the scale variance. Recent end-to-end DME methods based on deep models [15, 2] utilized bounding box predictions while generating estimated density map predictions. To deal with the issues due to the lack of labelled data, recent research efforts have also been made to explore unsupervised, weakly-supervised or semi-supervised object counting methods using unlabelled or partially labelled data [13, 14, 16, 30]. Those CNN-based DME methods, though mainly aimed at solving the crowd counting problems, can also be applied for vehicle counting [17, 18], counting

in cell microscopy images [19, 20], and remote sensing [31].

2.2 Attention Mechanisms in Object Counting

Attention mechanisms can put different weights to corresponding features to further refine the derived feature maps and highlight features that are important to help make better model predictions. In many computer vision tasks, attention mechanisms are introduced to refine the extracted image features at different levels, capturing long-term dependence and dealing with the limited receptive field of convolutional neural networks. Residual Attention modules [32] insert an encoder-decoder network in the residual branch to generate an attention map. SENet [33] generates channel attention weights by pooling the image features over the spatial dimension and recalibrates the derived features. Convolutional Block Attention Modules (CBAM) [34] add a spatial attention map to SENet and thus can further refine the derived features. Those methods have achieved good performance on image classification, detection and semantic segmentation tasks.

Many recent research efforts have been made to apply attention mechanisms to object counting models [2, 3, 4, 6]. In [2], the author jointly learn a regression-based density map and a detection-based density map guided by an attention network. In [3], the authors used both global and local attention branches to scale the whole density map and finetune pixel values in local image regions respectively. In [4], the features extracted by applying convolution operations with different dilation rates are fused to enlarge the receptive field and capture features at different scales. In [6], the input image is segmented into sparse and dense regions and the count estimates are derived in these regions respectively. Although these methods can deal with the image appearance variation and achieve good counting accuracy, most of them are trying to use an attention branch to focus on their assigned regions, specifically, dense or sparse ones, to adaptively estimate the corresponding object counts. Some of those attention networks require to be trained separately.

In this thesis, we want our model to be computationally simple using plug-in attention modules and adaptively learn where to pay attention for accurate counting. More critically, we would like to modify such attention modules to allow uncertainty quantification without incurring significant computation burden or sacrificing counting accuracy.

3. BAYESIAN COUNTING

In this section, we will provide a detailed introduction of different components of our Bayesian counting model that addresses unique challenges for plant counting in UAV images. We start our discussion by introducing the density estimation problem and most commonly used CNN-based models in the existing literature. We introduce the baseline model for plant counting in Section 3.1 and then illustrate potential problems when these approaches are applied to agriculture applications in Section 3.2. To solve these problems, in Section 3.3, we propose our attention-based method for plant counting. All the components of our Bayesian counting model are also detailed in Section 3.3.

Our Bayesian counting model contains a modified ResNet-18 [35] to extract the features and Convolutional Block Attention Module (CBAM) [34] to refine the feature map. Inspired by Bayesian Attention modules [36], we model the attention weights in a stochastic way to further enable uncertainty quantification of estimated counts. Our model can not only provide accurate and robust object counting in agriculture, but also output how confident the prediction is.

3.1 Supervised Density-Map-Estimation (DME) for Object Counting

Given an image $\mathbf{I}$ with $\{\mathbf{x}_j \in \mathbb{R}^2,\ j = 1, 2, \ldots, J\}$ denoting the corresponding pixel location in the domain of $\mathbf{I}$, let $\{\mathbf{y}_n \in \mathbb{R}^2, n = 1, 2, \ldots, N_{\mathbf{I}}\}$ be the corresponding locations of the $N_{\mathbf{I}}$ objects of interest in the given image $\mathbf{I}$. The objective of density-map-estimation (DME) is to learn a mapping f from the input image $\mathbf{I}(\mathbf{x}_j)$ to a density map $\mathbf{D}_{\mathbf{I}}(\mathbf{x}_j)$ across the image domain. The pixel value of the density map should denote the aggregated probability of interesting objects at that pixel. The estimated number of objects in $\mathbf{I}$ can then be calculated by integrating the density map over the image domain: $N_{\mathbf{I}}^{est} = \sum_{j=1}^{J} \mathbf{D}_{\mathbf{I}}(\mathbf{x}_j)$.

Traditional supervised DME methods often assume that the ground-truth density map of interesting objects in a given image $\mathbf{I}$ and annotated object locations $\{\mathbf{y}_n\}_1^{N_{\mathrm{I}}}$, denoted as $\mathbf{D}_{\mathbf{I}}^{gt}$, can be modelled by the summation of 2-D Gaussian functions with the mean at $\mathbf{y}_n$ and the variance σ^2

corresponding to each object in $\mathbf{I}$:

$$\mathbf{D}_{\mathbf{I}}^{gt}(\mathbf{x}_j) = \sum_{n=1}^{N_{\mathbf{I}}} \mathcal{N}(\mathbf{x}_j; \mathbf{y}_n, \sigma^2), \tag{3.1}$$

where the Gaussian function $\mathcal{N}(\mathbf{x}_j; \mathbf{y}_n, \sigma^2)$ models the probability of the nth object appearing at the corresponding pixel locations in the given image.

The counting problem now can be transferred to learn a mapping from a given image $\mathbf{I}$ to the corresponding density map $\mathbf{D}_{\mathbf{I}}^{gt}(\mathbf{x}_j)$. In this thesis, we adopt the same framework of recent end-to-end deep learning DME methods to model the mapping function f using deep neural networks with the encoder-decoder architecture. The encoder network will generate an intermediate feature map $\mathbf{F}$ and then the decoder network will incorporate those features and produce an estimated density map $\mathbf{D}_{\mathbf{I}}^{est}(\mathbf{x}_j)$. We use $f(\omega)$ to denote the deep neural network parametrized by model parameters ω.

To train this neural network, the loss function is typically based on the difference between the estimated density map $\mathbf{D}_{\mathbf{I}}^{est}(\mathbf{x}_j)$ and the ground-truth density map $\mathbf{D}_{\mathbf{I}}^{gt}(\mathbf{x}_j)$ for the images in the training set. Given a distance function $\mathcal{F}$, the loss function can be defined as:

$$L = \frac{1}{A} \sum_{a=1}^{A} \sum_{j=1}^{J} \mathcal{F}(\mathbf{D}_{\mathbf{I}_a}^{est}(\mathbf{x}_j), \mathbf{D}_{\mathbf{I}_a}^{gt}(\mathbf{x}_j)), \tag{3.2}$$

where A is the total number of annotated training images. Once the density mapping neural network $f(\omega)$ is trained, the estimated count for a new testing image $\mathbf{I}_{test}$ can be calculated by the integral of the predicted density map $\mathbf{D}_{\mathbf{I}_{test}}^{est}(\mathbf{x}_j)$ over the image domain as explained previously.

The pixel-wise Euclidean distance is the most widely used distance function as $\mathcal{F}$ in the loss function. Researchers recently argue that such pixel-wise supervision may ignore the local correlation between neighboring pixels and bias the model with empirical demonstrations [8]. Instead of measuring a pixel-level error, the structure similarity (SSIM) loss function and its variations [8, 9] further impose local constraints. More recently, the authors in [7] have proposed a novel Bayesian loss function, combining local constraints in a Bayesian framework. In this thesis, we will adopt

this Bayesian loss for more robust plant counting. Given an input image $\mathbf{I}$ and its corresponding labelled object locations $\{\mathbf{y}_n\}_1^{N_\mathbf{I}}$, the Bayesian loss for each training image $\mathbf{I}$ is calculated as:

$$L_{Bayes}(\mathbf{I}) = \sum_{n=1}^{N_\mathbf{I}} \left| 1 - \frac{\mathcal{N}(\mathbf{x}_j; \mathbf{y}_n, \sigma^2)}{\sum_{j=1}^{J} \mathcal{N}(\mathbf{x}_j; \mathbf{y}_n, \sigma^2)} \mathbf{D}_{\mathbf{I}_a}^{est}(\mathbf{x}_j) \right|. \tag{3.3}$$

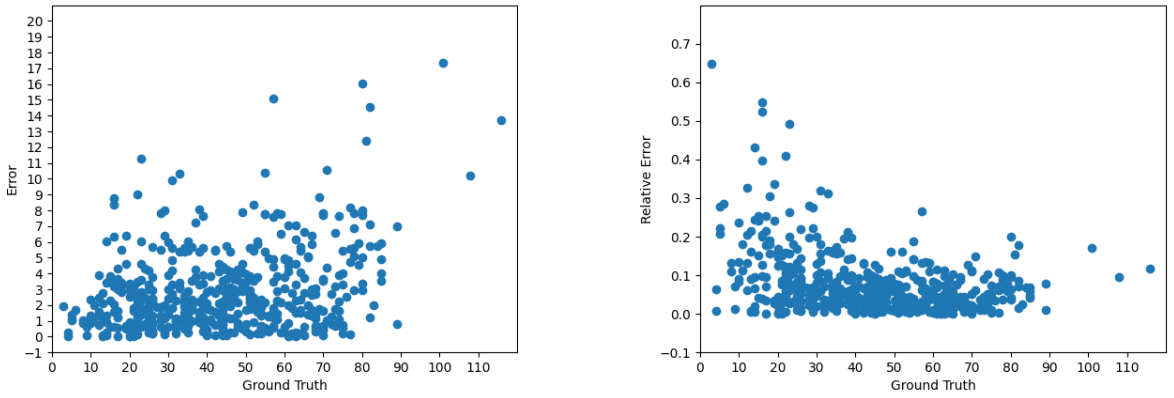
Instead of assigning a 2-D Gaussian function to each object in the image as the ground truth for pixel-wise supervision, Bayesian loss imposes local constraints by measuring the count expectation considering the local neighborhoods of each presenting thus leads to more reliable supervision [7].

3.2 Motivating Examples

DME methods have been widely used for crowd density prediction. However, compared to analyzing images with human crowds, counting plants of interest in given images, the focus of this thesis, faces unique challenges that require customized components for reliable automated plant counting. Often these plant images are taken by UAV cameras, which may generate images of significantly varying quality and appearance. Before we start discussing our method, we first visualize the corresponding challenges faced in plant image analysis. We illustrate the potential problems of directly applying crowd counting models to plant counting by training a ResNet18 counting network [37], which we will also use as a baseline method in our expreiments, and analyze the counting results on the Global Wheat Dataset [1]. We will give more details about our baseline method and training-testing data division in Section 4.

Figure 3.1 shows some example images in the Global Wheat Dataset. The objects of interest in this dataset are wheat heads. We can easily notice that those objects of interest can be significantly different in shape, color, and scale. In some images, objects of interest are severely occluded. The background and illumination also varies in each images. The weed and shade in background may easily be confused with wheat heads, our objects of interest.

Moreover, due to relatively difficult imaging quality control, some plant images may be very different to the others, which makes the dataset highly unbalance distributed. Figure 3.3 shows some images in which there exist severe counting errors. The wheat heads are very sparse in the



(a) Absolute counting error

(b) Relative counting error

Figure 3.1: Counting errors with respect to the ground-truth wheat head counts by the ResNet18 baseline model.

left two images, while highly dense in the right two. The objects of interest in the right two images are even out of focus. Although those kind of images only account for a small portion of the whole dataset, they hurts the overall counting performance severely. In real application of automated plant counting, we can expect more unbalance distributed data as well as out-of-distribution(OOD) data. Instead of making a prediction with high confidence, we would prefer our model being uncertain about its estimation in those images and thus, we can count the plant manually.

3.3 Bayesian counting

The overall architecture of our proposed Bayesian counting model is illustrated in Figure 3.4. The network architecture is composed of two backbone components: (1) Encoder: a modified ResNet18 with its last two residual blocks, and the following pooling layer and fully connected (FC) layer removed to extract the intermediate feature map **F**, (2) Decoder: a density estimator contains two convolution layers and a upsample layer to derive from the feature map **F** to output a density map. In order to overcome the aforementioned challenges when analyzing UAV plant images, we further include attention modules to alleviate the possible counting bias due to the

Figure 3.2: Plant images vary significantly in appearance, scale, background, and illumination. Reprint from [1]

Figure 3.3: Counting errors are severe in very sparse images, highly dense images, and out of focus images. Reprint from [1]

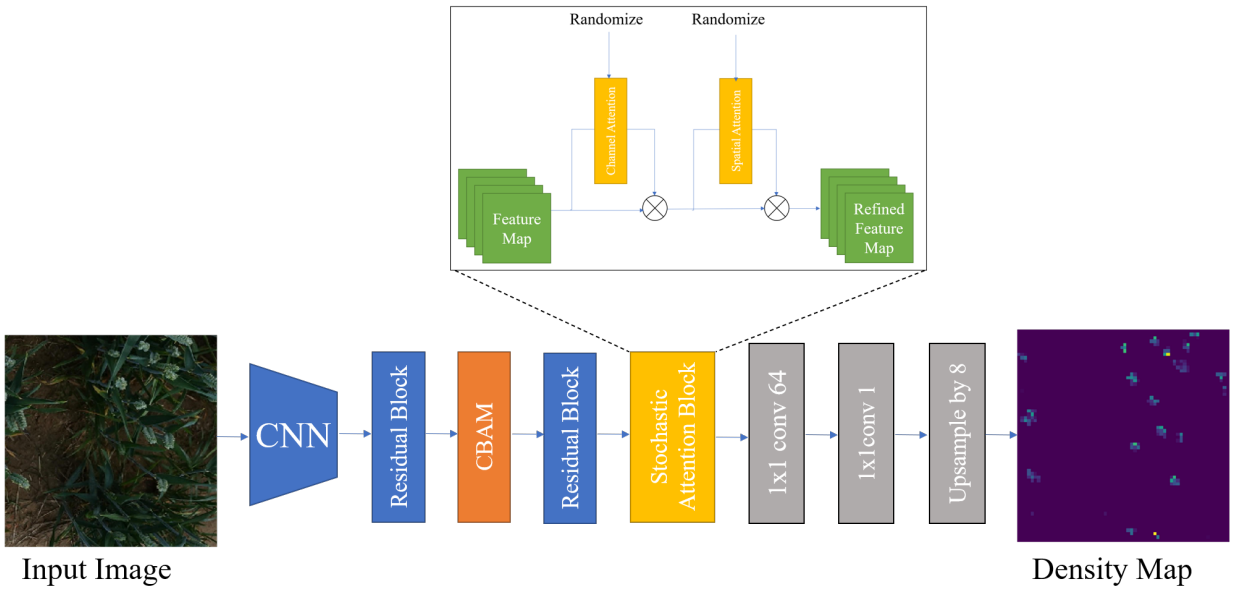


Figure 3.4: The overall architecture of our proposed model.

image resolution, object density, as well as background variability in given images. More critically, due to the agriculture applications as our ultimate operational goal, we equip the attention modules with the capability of uncertainty quantification of the predicted counts by deploying their Bayesian modifications with stochastic attention weights.

3.3.1 Refine the feature map with attention modules

In computer vision, attention mechanisms can typically be classified as channel attention, which highlight some features while suppress others, and spatial attention, which highlight a specific area of the feature map while suppress the remaining. Given an intermediate feature map $F \in \mathbb{R}^{C \times H \times W}$, where C, H, and W denote channel, height and width of the feature map, respectively, we would like to integrate attention mechanisms to help capture local and global dependence of image pixels. Different existing attention modules can be adopted such as Residual Attention [32], SE-Net [33], and Convolution Block Attention Module [34]. SE-Net can be seen as an example of channel attention, while Residual Attention and Convolution Block Attention Module involves both spatial attention and channel attention. To convey the key idea, we present the imple-

mentation with the Convolution Block Attention Module (CBAM) [34]. With CBAM incorporated, the refined feature map $\mathbf{F}''$ is calculated in the following way:

$$F' = M^{\text{Channel}}(F) \otimes F, \tag{3.4}$$

$$F'' = M^{\text{Spatial}}(F') \otimes F', \tag{3.5}$$

$M^{\text{Channel}}(F) \in \mathbb{R}^{C\times 1\times 1}$ is the channel attention module, and $M^{\text{Spatial}}(F') \in \mathbb{R}^{1\times H\times W}$ is the spatial attention module. $\otimes$ denotes the elementwise multiplication operator. The channel and spatial attention module map $M^{\text{Channel}}(F)$ and $M^{\text{Spatial}}(F')$ is calculated by applying a sigmoid activation function to each intermediate feature map T:

$$\begin{aligned} M^{\text{Channel}}(F) &= Sigmoid(T^{\text{Channel}}(F)), \\ M^{\text{Spatial}}(F') &= Sigmoid(T^{\text{Spatial}}(F')). \end{aligned} \tag{3.6}$$

The channel attention module will first pool the intermediate feature map F along width and height axis. The intermediate channel attention weights T^{Channel} are then calculated by forwarding two pooling features through a shared multi-layer perceptron (MLP) with one hidden layer and summing them element-wise:

$$T^{\text{Channel}}(F) = MLP(AvgPool(F)) + MLP(MaxPool(F)), \tag{3.7}$$

While the spatial attention module will pool the feature map refined by channel attention module F' along channel axis. The intermediate channel attention weights T^{Spatial} are then calculated by forwarding two pooling features through a convolutional layer:

$$T^{\text{Spatial}}(F') = CK^{7\times 7}([AvgPool'(F'), MaxPool'(F')], \tag{3.8}$$

Here we follow [34] and use both average pooling features and MaxPooling features. The

kernel size of convolutional layer CK is set to be 7×7.

3.3.2 Bayesian attention modules with stochastic weights

As we have discussed previously, the plant image data can be highly unbalance distributed. In real application, we can expect more out-of-distribution data. Therefore, we want our model to be capable of quantifying the uncertainty of the prediction. Inspired by the recently developed Bayesian Attention Modules [36], we further modify the integrated attention modules by modelling the attention weights in a stochastic way. With stochastic attention weights, it naturally enables uncertainty quantification of the estimated density map and thereafter the count predictions.

Still taking CBAM as the implementation, instead of having deterministic attention weights in the corresponding channel and spatial attention maps M^{Channel} and M^{Spatial}, we model them as random variables.

Given the training set $\mathcal{D}$ with all the training images $\mathbf{I}_a$ and annotated object locations $\{\mathbf{y}_n\}_1^{N_{\mathbf{I}_a}}$, we would like to derive the posterior distribution of the attention weights $p_\theta(M|\{\mathbf{I}_a, \{\mathbf{y}_n\}_1^{N_{\mathbf{I}_a}}\}_1^A)$. According to Bayes' theorem,

$$\begin{aligned} p_\theta(M|\{\mathbf{I}_a, \{\mathbf{y}_n\}_1^{N_{\mathbf{I}_a}}\}_1^A) &= \frac{p(\{\mathbf{I}_a, \{\mathbf{y}_n\}_1^{N_{\mathbf{I}_a}}\}_1^A|M)p_\eta(M)}{\int_M p(\{\mathbf{I}_a, \{\mathbf{y}_n\}_1^{N_{\mathbf{I}_a}}\}_1^A|M)p_\eta(M)} \\ &= \frac{p(\{\{\mathbf{y}_n\}_1^{N_{\mathbf{I}_a}}\}_1^A|\{\mathbf{I}_a\}_1^A, M)p_\eta(M)}{p(\{\{\mathbf{y}_n\}_1^{N_{\mathbf{I}_a}}\}_1^A|\{\mathbf{I}_a\}_1^A)}. \end{aligned} \tag{3.9}$$

The second equality is derived by assuming the independence of M and $\{\mathbf{I}_a\}_1^A$.

Following [36], we resort to variational inference and define $q_\phi(M)$ to approximate $p_\theta(M|\{\mathbf{I}_a, \{\mathbf{y}_n\}_1^{N_{\mathbf{I}_a}}\}_1^A)$

by minimizing the KL-Divergence between $q_\phi(M)$ and $p_\theta(M|\{\mathbf{I}_a, \{\mathbf{y}_n\}_1^{N_{\mathbf{I}_a}}\}_1^A)$:

$$
\begin{aligned}
&D_{KL}(q_\phi(M)||p_\theta(M|\{\mathbf{I}_a, \{\mathbf{y}_n\}_1^{N_{\mathbf{I}_a}}\}_1^A) \\
&= \mathbb{E}_{q_\phi(M)}[\log q_\phi(M)] - \mathbb{E}_{q_\phi(M)}[\log p_\theta(M|\{\mathbf{I}_a, \{\mathbf{y}_n\}_1^{N_{\mathbf{I}_a}}\}_1^A)] \\
&= \mathbb{E}_{q_\phi(M)}[\log q_\phi(M)] - \mathbb{E}_{q_\phi(M)}[\log p(\{\{\mathbf{y}_n\}_1^{N_{\mathbf{I}_a}}\}_1^A|\{\mathbf{I}_a\}_1^A, M)] - \mathbb{E}_{q_\phi(M)}[\log p_\eta(M)] \\
&+ \mathbb{E}_{q_\phi(M)}[p(\{\{\mathbf{y}_n\}_1^{N_{\mathbf{I}_a}}\}_1^A|\{\mathbf{I}_a\}_1^A)] \\
&= -\mathcal{L}(\{\mathbf{I}_a\}_1^A, \{\{\mathbf{y}_n\}_1^{N_{\mathbf{I}_a}}\}_1^A) + \mathbb{E}_{q_\phi(M)}[p(\{\{\mathbf{y}_n\}_1^{N_{\mathbf{I}_a}}\}_1^A|\{\mathbf{I}_a\}_1^A)].
\end{aligned}
\tag{3.10}
$$

The exact Bayesian inference is often computationally intractable. Instead of directly minimizing the KL-divergence between $q_\phi(M)$ and $p_\theta(M|\{\mathbf{I}_a, \{\mathbf{y}_n\}_1^{N_{\mathbf{I}_a}}\}_1^A)$, we maximize $\mathcal{L}(\{\mathbf{I}_a\}_1^A, \{\{\mathbf{y}_n\}_1^{N_{I_a}}\}_1^A)$, which is also known as the evidence lower bound (ELBO) of the log likelihood $\log p_\theta(\{\{\mathbf{y}_n\}_1^{N_{\mathbf{I}_a}}\}_1^A|\{\mathbf{I}_a\}_1^A)$:

$$
\begin{aligned}
\mathcal{L}(\{\mathbf{I}_a\}_1^A, \{\{\mathbf{y}_n\}_1^{N_{\mathbf{I}_a}}\}_1^A) &= \mathbb{E}_{q_\phi(M)}[\log p_\theta(\{\{\mathbf{y}_n\}_1^{N_{\mathbf{I}_a}}\}_1^A|\{\mathbf{I}_a\}_1^A, M)] - D_{KL}(q_\phi(M)||p_\eta(M)) \\
&= \mathbb{E}_{q_\phi(M)}[\log \frac{p_\theta(\{\{\mathbf{y}_n\}_1^{N_{\mathbf{I}_a}}\}_1^A|\{\mathbf{I}_a\}_1^A, M)p_\eta(M)}{q_\phi(M)}].
\end{aligned}
\tag{3.11}
$$

We learn the approximated posterior distribution of attention weights $q_\phi(M)$ by optimizing $\mathcal{L}(\{\mathbf{I}_a\}_1^A, \{\{\mathbf{y}_n\}_1^{N_{\mathbf{I}_a}}\}_1^A)$ w.r.t θ,ϕ and η. The KL-divergence between approximated distribution and prior distribution $D_{KL}(q_\phi(M)||p_\eta(M))$ can be treated as a regularization term. We can insert our belief of the distribution of M by assigning a different prior $p_\eta(M)$.

3.3.3 Modeling attention weights M as random variables

Let S be intermediate attention weights, and attention weights M are determined only by S. Once the distribution $p_\theta(S|\{\mathbf{I}_a, \{\mathbf{y}_n\}_1^{N_{\mathbf{I}_a}}\}_1^A)$ is given, $p_\theta(M|\{\mathbf{I}_a, \{\mathbf{y}_n\}_1^{N_{\mathbf{I}_a}}\}_1^A)$ can be determined. Let $q_\phi(S)$ be the approximated posterior distribution which depends on the same parameter ϕ as $q_\phi(M)$. Instead of directly modeling $q_\phi(M)$, we model $q_\phi(S)$ using some family of distribution. Based on the result provided by [36], we can model S as Weibull random variables. Next, we will briefly introduce Weibull distribution and its property, which is important to our discussion.

Property of the Weibull distribution: Let the random variable $s \sim \text{Weibull}(k, \lambda)$. The shape

parameter $k \in (0, +\infty)$ and the scale parameter $\lambda \in (0, +\infty)$. The probabilistic density function (PDF) of a Weibull random variable s is $f(s) = \frac{k}{\lambda}(\frac{s}{\lambda})^{k-1}\exp(-(s/k)^k)$, $s > 0$. Its mean is $\lambda\Gamma(1 + 1/k)$ and variance is $\lambda^2[\Gamma(1 + 2/k) - (\Gamma(1 + 1/k))^2]$. The Weibull distribution can be reparameterized in term of a uniform distribution: sample $s \sim$ Weibull(k, λ) is equivalent to sample $\epsilon \sim$ Uniform$(0, 1)$ and let $s = \tilde{g}(\epsilon) = \lambda(-\log(1 - \epsilon)^{1/k})$. The KL-divergence between Weibull distribution and Gamma distribution has analytic form:

$$\begin{aligned} &D_{KL}(\text{Weibull}(k, \lambda)||\text{Gamma}(\alpha, \beta)) \\ &= \frac{\gamma\alpha}{k} - \alpha\log\lambda + \log k + \beta\lambda\Gamma(1 + \frac{1}{k}) - \gamma - 1 - \alpha\log\beta + \log\Gamma(\alpha). \end{aligned} \tag{3.12}$$

As we have discussed, channel attention M^{Channel} and spatial attention M^{Spatial} are two types of attention mechanism in computer vision. Next, we will discuss how each of channel attention M^{Channel} and spatial attention M^{Spatial} can be modeled in a stochastic way. We will take CBAM as a concrete example.

Modeling the channel attention weights M^{Channel} to be stochastic: In deterministic case, M^{Channel} is the activation of T^{Channel} in (3.7), both of which are matrices of dimension $C \times 1 \times 1$. We use upper-case letter with subscript index M_c^{Channel} to denotes the $c, 1, 1$-th entry of matrix M^{Channel}. To model M^{Channel} in a stochastic way, here we introduce S^{Channel}, an intermediate random matrix with each of its entry S_c^{Channel} being a Weibull random variable. M^{Channel} is the activation of S^{Channel}.

We treat shape parameter k as a hyperparameter. And $\lambda_c^{\text{Channel}}$, the scale parameter of S_c^{Channel}, is $\lambda_c^{\text{Channel}} = \frac{ReLU(T_c^{\text{Channel}})}{\Gamma(1+1/k)}$. We add a Rectified Linear Unit (ReLU) activation function here since the scale parameter of Weibull distribution $\lambda > 0$. Then, by applying the reparametrization, each entry of the intermediate channel attention vector S_c^{Channel} can be sampled as:

$$\begin{aligned} S_c^{\text{Channel}} &= \tilde{g}(\epsilon_c^{\text{Channel}}) = \lambda_c^{\text{Channel}}(-\log(1 - \epsilon_c^{\text{Channel}})^{1/k}) \\ &= \frac{ReLU(T_c^{\text{Channel}})}{\Gamma(1 + 1/k)}(-\log(1 - \epsilon_c^{\text{Channel}}))^{1/k}, \end{aligned} \tag{3.13}$$

where $\epsilon^{\text{Channel}} \sim$ Uniform$(0, 1)$. $\epsilon^{\text{Channel}} \in \mathbb{R}^{C\times 1\times 1}$.

Modeling the spatial attention weights M^{Spatial} to be stochastic: In deterministic case, M^{Spatial} is the activation of T^{Spatial} in (3.8), both of which are matrices of dimension $1 \times H \times W$. Similarly, we use upper-case letter with subscript indices $M^{\text{Spatial}}_{h,w}$ to denotes the $1, h, w$-th entry of matrix M^{Spatial}. To model M^{Spatial} in a stochastic way, we introduce S^{Spatial}, an intermediate random matrix with each of its entry $S^{\text{Spatial}}_{h,w}$ being a Weibull random variable. M^{Spatial} is the activation of S^{Spatial}.

Like in channel counterpart, we are still treating k as a hyperparameter. And $\lambda^{\text{Spatial}}_{h,w}$, the scale parameter of $S^{\text{Spatial}}_{h,w}$, is $\lambda^{\text{Channel}}_{h,w} = \frac{ReLU(T^{\text{Spatial}}_{h,w})}{\Gamma(1+1/k)}$. By applying reparameterization, we can sample S^{Spatial} as:

$$\begin{aligned} S^{\text{Spatial}}_{h,w} &= \tilde{g}(\epsilon^{\text{Spatial}}_{h,w}) = \lambda^{\text{Spatial}}_{h,w}(-\log(1-\epsilon^{\text{Spatial}}_{h,w})^{1/k}) \\ &= \frac{T^{\text{Spatial}}_{h,w}}{\Gamma(1+1/k)}(-\log(1-\epsilon^{\text{Spatial}}_{h,w}))^{1/k}, \end{aligned} \tag{3.14}$$

where $\epsilon^{Spatial} \sim \text{Uniform}(0,1)$. $\epsilon^{\text{Spatial}} \in \mathbb{R}^{1\times H\times W}$.

Until now, we have introduced intermediate Weibull random matrix S, and M is the activation of S. We can reparameterize the random variables M using a differentiable transformation $g(\cdot)$ of auxiliary variables ϵ:

$$M = g(\epsilon) = Sigmoid(\tilde{g}(\epsilon)), \text{with } \epsilon \sim \text{Uniform}(0,1). \tag{3.15}$$

Note that when both of spatial and channel attention weights are modeled as random variables, or more than one attention modules are modeled as random variables, due to the interdependency of different stochastic attention modules, the KL-divergence $D_{KL}(q_\phi(S)||p_\eta(S))$ will not have a analytic form [36]. However, by making KL-divergence semi-analytic and plugging in the analytic part, the Monte Carlo estimation variance can be reduced [36].

3.3.4 Learning objective function

Based on previous discussion and by applying reparameterization, we can get our final learning objective. When spatial or channel attention weights are modeled as random variables, we have the analytical form of KL-divergence term $D_{KL}(q_\phi(S)||p_\eta(S))$. The gradient estimator of the evidence lower bound (ELBO) is as follows:

$$\mathcal{L}(\{\mathbf{I}_a\}_1^A, \{\{\mathbf{y}_n\}_1^{N_{\mathbf{I}_a}}\}_1^A) = \mathbb{E}_\epsilon[\log p_\theta(\{\{\mathbf{y}_n\}_1^{N_{\mathbf{I}_a}}\}_1^A|\{\mathbf{I}_a\}_1^A, \tilde{g}_\phi(\epsilon))] - D_{KL}(q_\phi(S)||p_\eta(S)). \quad (3.16)$$

While when both of spatial and channel attention weights are modeled as random variables, our estimate the gradient using:

$$\begin{aligned}&\mathcal{L}(\{\mathbf{I}_a\}_1^A, \{\{\mathbf{y}_n\}_1^{N_{\mathbf{I}_a}}\}_1^A) = \mathbb{E}_\epsilon[\log p_\theta(\{\{\mathbf{y}_n\}_1^{N_{\mathbf{I}_a}}\}_1^A|\{\mathbf{I}_a\}_1^A, \tilde{g}_\phi(\epsilon))]- \\ &\mathbb{E}_\epsilon[D_{KL}(q_\phi(S^{\text{Spatial}}|\tilde{g}(\epsilon^{\text{Channel}})||p_\eta(S^{\text{Spatial}}|\tilde{g}(\epsilon^{\text{Channel}}))))] + D_{KL}(q_\phi(S^{\text{Channel}})||p_\eta(S^{\text{Channel}})).\end{aligned} \quad (3.17)$$

It is similar when more than one attention modules are modeled as random variables. Integrating with the auto-differentiation modules in PyTorch, we maximize $\mathcal{L}(\{\mathbf{I}_a\}_1^A, \{\{\mathbf{y}_n\}_1^{N_{\mathbf{I}_a}}\}_1^A)$ by computing the gradient of $\mathcal{L}(\{\mathbf{I}_a\}_1^A, \{\{\mathbf{y}_n\}_1^{N_{\mathbf{I}_a}}\}_1^A)$ with backpropagation to train θ, ϕ, η.

4. EXPERIMENTS

In this chapter, we evaluate our Bayesian counting model on the Global Wheat dataset [1] To deal with the challenge due to insufficient annotated training image data, we augment the training set using random cropping and random flipping. We compare our Bayesian counting method with several baseline models. Ablation studies are also performed to validate the effect of different model components. To evaluate the counting accuracy and uncertainty estimation reliability, we calculate the mean error and variance of the predicted counts. In the following sections, we first detail the experimental setups and then present our experimental results with discussion.

4.1 Datasets

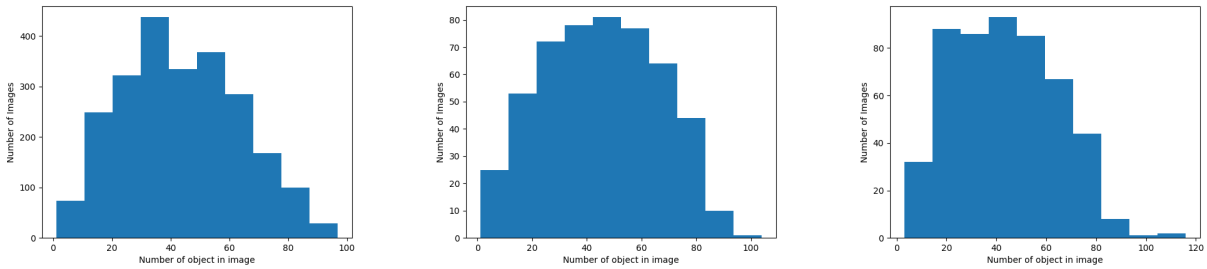
4.1.1 Global Wheat Dataset

The Global Wheat Dataset [1] is a large-scale dataset for benchmarking wheat head detection and count estimation. It contains about 4,700 high resolution images and 190,000 wheat head labels. In our experiments, we only focus on predicting the number of wheat heads in each image. In 3,373 images that have annotated wheat heads and counts openly accessible to the public, we randomly select 2,362 images for training, 506 images for validation, and 505 images for testing.

Figure 4.1 illustrates the wheat head count distributions of our training, validation, and testing images. The training images contain on average 43.59 wheat heads, with the standard deviation 20.13. The validation images contain on average 45.20 wheat heads, with the standard deviation 21.20. The test images contain on average 43.59 wheat heads, with the standard deviation 20.58.

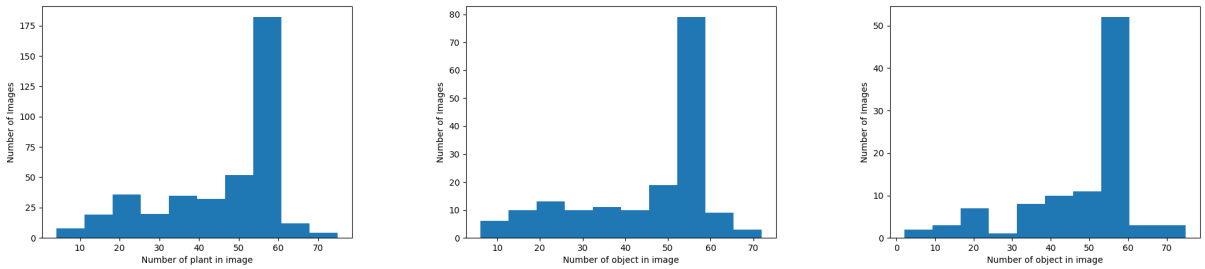
4.1.2 Field experiment data

We have also collected pictures by flying a low-altitude unmanned aerial vehicle (UAV) with high resolution camera on a local horticulture nursery farmfarm – TreeTownUSA – in Houston, Texas (latitude: $29.33°$, longitude: $-96.20°$). The drone (UAV) images were taken at Tree-TownUSA during the time of the growing seasons from 2017 to 2019. More than a hundred varieties of plant species are growing in the nursery farm. Here in our project, we mainly focus on



(a) training set (b) validation set (c) testing set

Figure 4.1: Histograms of wheat head counts in Global Wheat Dataset.



(a) training set (b) validation set (c) testing set

Figure 4.2: Histograms of plant counts from TreeTownUSA drone images.

three fields inside the nursery farm: the "West field" with the plant species of mainly oaks with the size around 95 acres; the "Area1" with multiple plant species and the size of round 200 acres, and "Area2" with around 12 acres of multiple plant species. The pictures are stitched into a whole-view map and then segmented according to their SKU. It contains a total of 684 1024×320 images. We randomly select 400 images for training, 170 images for validation and 100 images for testing.

Figure 4.2 illustrate the plant count distribution of our field experiment data. The training images contain on average 45.88 wheat heads, with the standard deviation 14.89. The validation images contain on average 45.44 wheat heads, with the standard deviation 15.65. The test images contain on average 48.02 wheat heads, with the standard deviation 14.72.

4.2 Evaluation metrics

4.2.1 Evaluate the counting performance

Mean Absolute Error (MAE) and Root Mean Squared Error (RMSE) are two widely used metrics for object counting, which are defined as follows:

$$MAE = \frac{1}{A}\sum_{a=1}^{A}|N_{\mathbf{I}_a} - N_{\mathbf{I}_a}^{est}|, \tag{4.1}$$

$$RMSE = \sqrt{\frac{1}{A}\sum_{a=1}^{A}|N_{\mathbf{I}_a} - N_{\mathbf{I}_a}^{est}|^2}, \tag{4.2}$$

where $N_{\mathbf{I}_a}$ is the number of object in image $\mathbf{I}_a$, and A is the total number of images. The term $N_{\mathbf{I}_a}^{est}$ in equation (4.1) and equation (4.2) is the estimated number of object in image $\mathbf{I}_a$, which is calculated by integral over the whole density map:

$$N_{\mathbf{I}_a}^{est} = \sum_{j=1}^{J}\mathbf{D}_{\mathbf{I}_a}^{est}(\mathbf{x}_j). \tag{4.3}$$

4.2.2 Evaluate the uncertainty estimation result

Given a trained network with stochastic attention modules, we estimation the uncertainty of the model by sampling multiple times and and calculate the variances of the predictions. We denote our estimated variances as Var^{est}. We evaluate our uncertainty estimation result on our test set using the percentage of the ground-truth within a band centered at the prediction N^{est} and with a bandwidth six standard deviation $\Pr(N^{est} - 3\sqrt{\text{Var}^{est}} < N < N^{est} + 3\sqrt{\text{Var}^{est}})$.

4.3 Experiments

We evaluate our method on two plant counting benchmarks: Global Wheat Dataset [1] and our collected field experiment data. To thoroughly evaluate how different loss function, data augmentation and attention module may help to improve the counting accuracy, we do ablation studies on

Global Wheat Dataset and report the counting result on our split test set. To evaluate the effect of parameters of stochastic attention module and search for the best modeling approach, we also do ablation studies by changing the shape parameter k and modeling different attention modules to be stochastic.

4.3.1 Ablation studies

We do ablation studies on the Global Wheat Dataset. We adopt the ResNet18 backbone [**?**] as the baseline architecture. We remove the last two residual blocks and fully connected layers of ResNet-18, and change the stride of the 5th block of ResNet18 to 1, following [37]. The decoder network contains two 1×1 convolution layers [37] to incorporate image features. The output is upsample by 8 to match the size of input images. The network is implemented on PyTorch based on [37]. We use the Adam optimizer [38] and set the learning rate to be 1e-5.We set the batch size to be 25.

Our experiment design can be split into three parts. We first evaluate different loss functions and data augmentation methods. Then we add attention modules. Finally, we evaluate the effect of Weibull shape parameter k for stochastic attention modules on three different settings. We will explain the detail of each experiment below.

Loss function and data augmentation: We experimentally compare the counting performance of Bayesian loss to pixelwise Euclidean loss and evaluate the effect of data augmentation. For Bayesian loss, we set σ to 20. For Pixel-wise Euclidean loss, we set σ to be 10. To help the neural networks trained using pixelwise Euclidean loss to converge correctly, we magnify the ground truth density map by 10.

To speed up the training procedure, we resize all the training images and test images to 512×512. To augment the training data, we randomly select 50% training images and crop them to 512×512, and resize the remaining training images to 512×512. We also flip the training images horizontally and vertically.

Experiment results can be summarized in Table 4.1. We further plot the ground truth $N_{\mathbf{I}}$ with respect to the absolute errors ($|N_{\mathbf{I}} - N_{\mathbf{I}}^{est}|$) and relative errors ($\frac{|N_{\mathbf{I}} - N_{\mathbf{I}}^{est}|}{N_{\mathbf{I}}}$) of each test images in

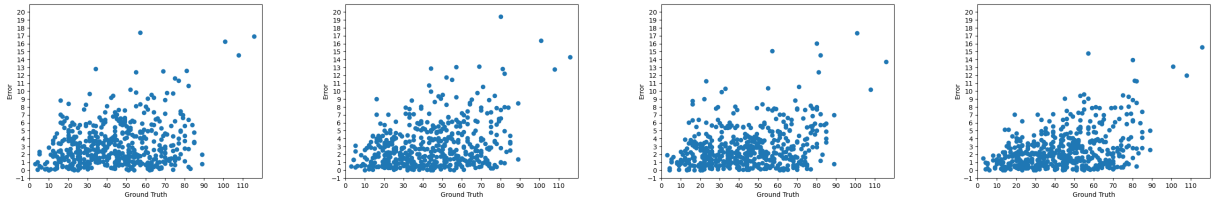
Description	# of Params (M)	RMSE	MAE
ResNet18 + pixelwise Euclidean loss (w/o augmentation)	2.80	4.34	3.26
ResNet18 + Bayesian loss (w/o augmentation)	2.80	3.87	2.88
ResNet18 + pixelwise Euclidean Loss (Resize + crop + flip)	2.80	4.20	3.13
ResNet18 + Bayesian loss (Resize + crop + flip)	2.80	**3.57**	**2.59**

Table 4.1: Effect of training loss functions and data augmentation. Counting accuracy is measured by root-mean-square error (RMSE) and mean-absolute error (MAE); The numbers of model parameters are in millions (M).

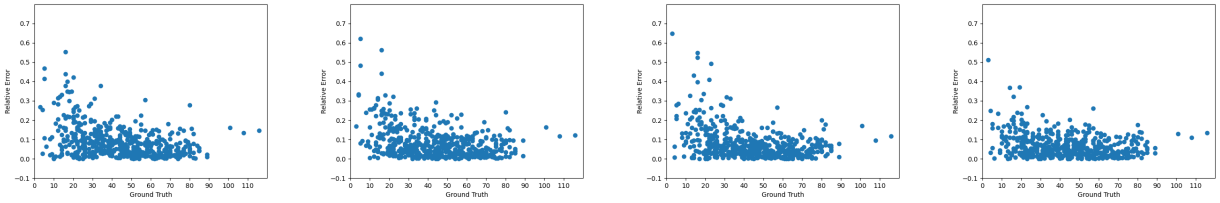
Figure 4.3. We can see that the absolute errors are higher when the test images contain highly dense wheat heads, while the relative errors are higher when wheat heads in test image are sparse. The network trained using Bayesian loss can produce more accurate count prediction than pixelwise Euclidean loss. Data augmentation can improve the counting accuracy on both pixelwise Euclidean loss and Bayesian loss. We observe that with data augmentation, the network will make better count prediction especially on images which the networks trained without data augmentation make large errors. This suggest that on highly variant images, the data augmentation is a critical part for training an accurate and robust counting model. As a brief conclusion, we will train our counting network with Bayesian loss and data augmentation.

Attention module and data augmentation: Although we have achieved better counting accuracy on by integrating with Bayesian loss and data augmentation, the absolute errors on highly dense images and relative errors on sparse images are still high. In this experiment, we evaluate the effectiveness of attention module for counting results. We compare the counting accuracy of network without attention modules to network with attention modules on two augmentation setups. We insert CBAM attention modules between the 5th and 6th residual blocks, and between 6th residual block and decoder network. We do experiment to see how attention module will affect the counting errors of images of different wheat head density.

Similarly, we report our experiment result in Table 4.2, and plot the ground-truth $N_{\mathbf{I}}$ with re-



(a) Absolute error with pixelwise loss (w/o aug.) (b) Absolute error with pixelwise loss (with aug.) (c) Absolute error with Bayesian loss (w/o aug.) (d) Absolute error with Bayesian loss (with aug.)



(e) Relative error with pixelwise loss (w/o aug) (f) Relative error with pixelwise loss (with aug.) (g) Relative error with Bayesian loss (w/o aug.) (h) Relative error with Bayesian loss (with aug.)

Figure 4.3: Comparison of counting errors with respect to the ground-truth counts with different loss functions and augmentation (aug.) setups.

Discription	# of Params (M)	RMSE	MAE
ResNet18 + Bayesian loss (w/o aug.)	2.80	3.87	2.88
ResNet18 + Bayesian loss + Attention (CBAM) (w/o aug.)	2.82	3.91	2.85
ResNet18 + Bayesian loss (with aug.)	2.80	3.57	2.59
ResNet18 + Bayesian loss + Attention (CBAM) (with aug.)	2.82	**3.19**	**2.33**

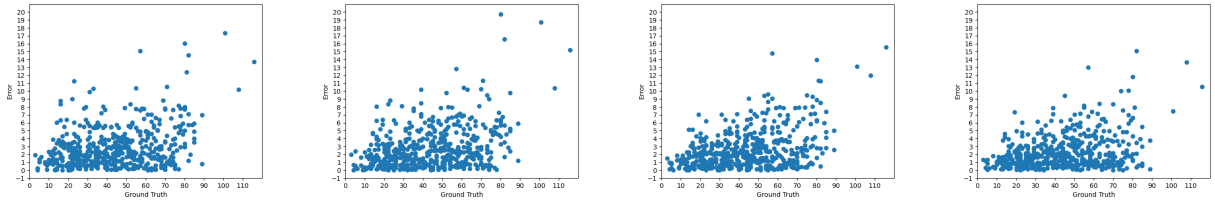
Table 4.2: Ablation studies with attention modules.

spect to the absolute errors ($|N_{\mathbf{I}} - N_{\mathbf{I}}^{est}|$) and relative errors ($\frac{|N_{\mathbf{I}} - N_{\mathbf{I}}^{est}|}{N_{\mathbf{I}}}$) of each test images in Figure 4.4. We observe that the performance are almost the same when the networks are trained without augmented data, however, the counting accuracy improves dramatically by applying attention modules on augmented dataset. In addition, from the Figure 4.4, we can find that the counting errors reduced on both highly dense images and sparse images. In a brief conclusion, although the attention modules is a powerful tool which has improve the performance on many other computer vision tasks, we still need to carefully design and trained the network to make use of them.

Throughout these two ablation studies, we study the effect of loss functions, data augmentation and attention modules on counting accuracy. Our final network architecture design is shown as in Figure 3.4. We train our Bayesian counting network using Bayesian loss with the aforementioned data augmentation.

Shape parameter k and different stochastic attention module setups: In last part of our ablation studies, we study the effect of different stochastic attention modules and Weibull shape parameter k. We model the CBAM attention module between 6th residual block and decoder network to be stochastic. In first and second setting, we model channel attention weights and spatial attention weights as random variables, respectively. In our third setting, we model both channel attention weights and spatial attention weights as random variables.

We report our experiment result in Table 4.3. As we can see, modeling the attention module to be stochastic will only slightly degrade the counting performance. The counting accuracy degrades the least when we use stochastic channel attention and set $k = 1$, while degrades the most when

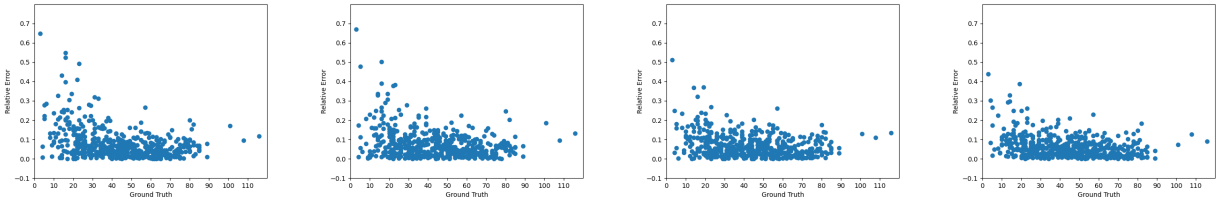


(a) Absolute error w/o attention (w/o aug.)

(b) Absolute error with attention (w/o aug.)

(c) Absolute error w/o attention (with aug.)

(d) Absolute error with attention (with aug.)



(e) Relative error w/o attention (w/o aug.)

(f) Relative error with attention (w/o aug.)

(g) Relative error w/o attention (with aug.)

(h) Relative error with attention (with aug.)

Figure 4.4: Comparison of counting errors with respect to the ground-truth counts with different augmentation setups and with/without (w/o) attention.

Stochastic attention type	k	RMSE	MAE	$\pm 3\sqrt{\text{Var}^{est}}$	$\pm 2\sqrt{\text{Var}^{est}}$	$\pm 1\sqrt{\text{Var}^{est}}$
Channel	0.99	3.24	2.36	77.6%	59.1%	33.4%
Channel	1	3.23	2.33	92.5%	82.2%	53.6%
Channel	5	3.30	2.38	32.0%	22.7%	12.1%
Spatial	0.99	3.44	2.49	31.8%	15.0%	5.9%
Spatial	1	3.26	2.38	13.0%	9.4%	5.7%
Spatial	5	3.39	2.45	8.3%	6.1%	3.1%
Channel+Spatial	0.99	3.32	2.44	83.6%	67.4%	39.7%
Channel+Spatial	1	3.30	2.42	85.4%	70.8%	42.5%
Channel+Spatial	5	3.67	2.73	56.3%	39.7%	23.5%

Table 4.3: Effect of the hyperparameter k with different stochastic attention module setups. We measure the uncertainty estimation result using the percentage of the ground-truth in a band centered at the prediction N^{est} with a bandwidth of six standard deviation Pr($N \in N^{est} \pm 3\sqrt{\text{Var}^{est}}$), four standard deviation Pr($N \in N^{est} \pm 2\sqrt{\text{Var}^{est}}$), and two standard deviation Pr($N \in N^{est} \pm 1\sqrt{\text{Var}^{est}}$).

we model both channel attention weights and spatial attention weights as random variables and set $k = 5$.

We also evaluate our uncertainty estimation using the portion of ground-truth in bands centered at prediction N^{est} with a bandwidth of $6\sqrt{\text{Var}^{est}}$, $4\sqrt{\text{Var}^{est}}$, and $2\sqrt{\text{Var}^{est}}$. For now, the uncertainty estimation result is unsatisfactory and may not be able to reflect the distribution of our data. We still need to carefully fine-tune and calibrate our stochastic attention modules. The result is relatively better when we only model the channel attention weights as random variables and set $k = 1$.

4.3.2 Experimental results

We have evaluated our Bayesian counting method together with other baseline methods on both the Global Wheat Dataset [1] and our own images from the field experiment of flying a drone in the selected fields in TreeTownUSA. The results are reported in Tables 4.4 and 4.5, respectively. By incorporating the Bayesian loss, attention modules, and appropriate data augmentation, our method outperforms the baseline methods on both of these two datasets. On the Global Wheat

Description	# of Params (M)	RMSE	MAE
ResNet18 + pixelwise Euclidean loss (baseline)	2.80	4.34	3.26
ResNet18 + Bayesian loss + Attention (CBAM)	2.82	3.19	2.33
ResNet18 + Bayesian loss + Stochastic Attention	2.82	3.23	2.33

Table 4.4: Results on Global Wheat Dataset.

Description	# of Params(M)	RMSE	MAE
ResNet18 + pixelwise Euclidean loss (baseline)	2.80	1.92	2.60
ResNet18 + Bayesian loss + Attention	2.82	1.23	1.54
ResNet18 + Bayesian loss + Stochastic Attention	2.82	1.59	2.31

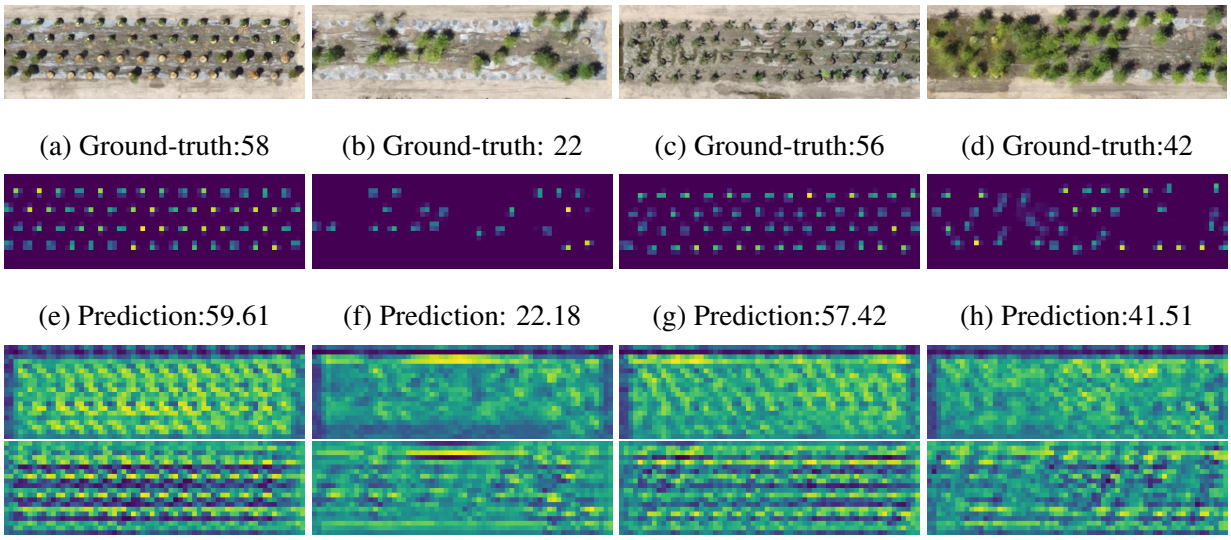
Table 4.5: Results on our own field experiment data from TreeTownUSA.

Dataset [1], by introducing stochastic attention weights, we can enable the uncertainty quantification capability of the counting model without significant degradation on counting accuracy. On our field experiment data, however, introducing the stochastic attention weights will result in a drop of counting performance. We still need to carefully fine-tune and calibrate both of two stochastic attention modules to improve the counting accuracy and uncertainty estimation results, which will be our future research direction.

4.3.3 Visualization and discussion

In this section, we provide several examples to help visualization of the derived predictions by our Bayesian counting method on both the Global Wheat Dataset [1] and our field experiment data. Figures 4.5 and 4.6 show our examples of the test images (first row), density map predictions (second row) and attention maps (third and forth rows). In the second, third, and forth rows, warmer colors denote higher values while cooler colors denote lower values.

As we can see in Figure 4.5, our method provides accurate count predictions together with accurate location information on our field experiment data when the boundary of plants is clear.



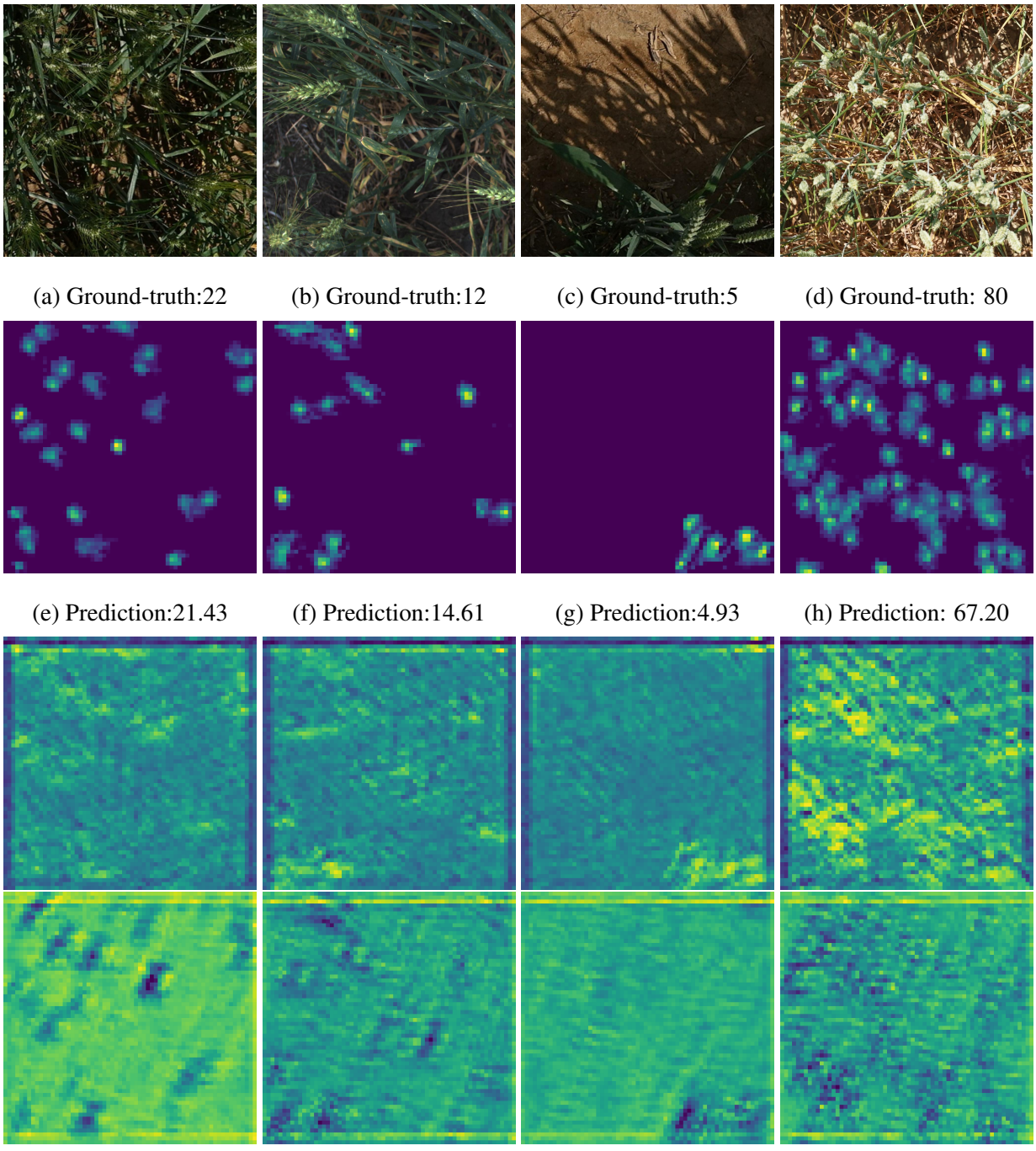
(a) Ground-truth:58 (b) Ground-truth: 22 (c) Ground-truth:56 (d) Ground-truth:42

(e) Prediction:59.61 (f) Prediction: 22.18 (g) Prediction:57.42 (h) Prediction:41.51

Figure 4.5: Visulization of field experiment images(first row), prediction(second row), attention map(third and forth row).

When the boundary becomes blurred, though we may not be able to distinguish each plant from the others in the derived density map, we are still able to estimate the counts accurately.

In Figure 4.6, we can observe that on the Global Wheat Dataset [1], the counting errors are low, even when the background appearance or illumination is complex. In highly dense images, however, the counting accuracy drops a lot. Although the model captures the locations of the most of wheat heads correctly, the model can not give an accurate estimation of the density. Further improvement of the counting accuracy in highly dense plant images will be another future research direction.



(a) Ground-truth:22 (b) Ground-truth:12 (c) Ground-truth:5 (d) Ground-truth: 80

(e) Prediction:21.43 (f) Prediction:14.61 (g) Prediction:4.93 (h) Prediction: 67.20

Figure 4.6: Visualization of test images(first row), prediction(second row), attention map(third and forth row) in Global Wheat Dataset. Figure 4.6a, 4.6b, 4.6c and 4.6d are reprint from [1]

.

5. CONCLUSIONS AND FUTURE RESEARCH

In this thesis, we study the topic of object counting in agriculture applications. To improve the performance and tackle the uncertainty issue in object counting, we have introduced attention modules and model the attention weights statistically to enable uncertainty quantification capability in counting. In partical, we use Weibull random variables to model attention weights so that we may derive a distribution of predicted counts instead of only providing point estimates as in the existing object counting models. We evaluate our Bayesian counting model on the Global Wheat Dataset and perform ablation studies to understand the effects of different model setups on count estimation accuracy with uncertainty quantification. Our experimental results demonstrate that adding attention modules can improve the the accuracy of count estimates, especially when images have varying quality and appearance. More importantly, introducing the randomness to the attention weights enable the first counting model with uncertainty quantification to the best of our knowledge, without harming the counting accuracy.

Here we note that our Bayesian counting method can be readily applied to the other object counting tasks such as crowd counting, vehicle counting, or environment survey. With the ability to estimate the uncertainty of prediction, we can make a more robust counting estimation and further facilitate automatic object counting. Often it is resource demanding to have high-quality label annotations given images. With our Bayesian counting model with uncertainty quantification capability, semi-supervised or active learning can be developed to achieve more efficient label annotations. Since our attention module is a plug-in module, we can easily adopt it to some other computer vision tasks. This will be another goal of our future research.

REFERENCES

[1] E. David, S. Madec, P. Sadeghi-Tehran, H. Aasen, B. Zheng, S. Liu, N. Kirchgessner, G. Ishikawa, K. Nagasawa, M. A. Badhon, C. Pozniak, B. de Solan, A. Hund, S. C. Chapman, F. Baret, I. Stavness, and W. Guo, "Global wheat head detection (gwhd) dataset: a large and diverse dataset of high resolution rgb labelled images to develop and benchmark wheat head detection methods," 2020.

[2] J. Liu, C. Gao, D. Meng, and A. G. Hauptmann, "Decidenet: Counting varying density crowds through attention guided detection and density estimation," *CoRR*, vol. abs/1712.06679, 2017.

[3] M. A. Hossain, M. Hosseinzadeh, O. Chanda, and Y. Wang, "Crowd counting using scale-aware attention networks," *CoRR*, vol. abs/1903.02025, 2019.

[4] D. Guo, K. Li, Z.-J. Zha, and M. Wang, "Dadnet: Dilated-attention-deformable convnet for crowd counting," in *Proceedings of the 27th ACM International Conference on Multimedia*, pp. 1823–1832, 2019.

[5] Z. Shi, P. Mettes, and C. G. M. Snoek, "Counting with focus for free," *CoRR*, vol. abs/1903.12206, 2019.

[6] X. Jiang, L. Zhang, M. Xu, T. Zhang, P. Lv, B. Zhou, X. Yang, and Y. Pang, "Attention scaling for crowd counting," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 4706–4715, 2020.

[7] Z. Ma, X. Wei, X. Hong, and Y. Gong, "Bayesian loss for crowd count estimation with point supervision," *CoRR*, vol. abs/1908.03684, 2019.

[8] X. Cao, Z. Wang, Y. Zhao, and F. Su, "Scale aggregation network for accurate and efficient crowd counting," in *Proceedings of the European Conference on Computer Vision (ECCV)*, pp. 734–750, 2018.

[9] Z. Qiu, L. Liu, G. Li, Q. Wang, N. Xiao, and L. Lin, "Crowd counting via multi-view scale aggregation networks," in *2019 IEEE International Conference on Multimedia and Expo (ICME)*, pp. 1498–1503, 2019.

[10] Sheng-Fuu Lin, Jaw-Yeh Chen, and Hung-Xin Chao, "Estimation of number of people in crowded scenes using perspective transformation," *IEEE Transactions on Systems, Man, and Cybernetics - Part A: Systems and Humans*, vol. 31, no. 6, pp. 645–654, 2001.

[11] M. Li, Z. Zhang, K. Huang, and T. Tan, "Estimating the number of people in crowded scenes by mid based foreground segmentation and head-shoulder detection," in *2008 19th International Conference on Pattern Recognition*, pp. 1–4, 2008.

[12] A. B. Chan, Zhang-Sheng John Liang, and N. Vasconcelos, "Privacy preserving crowd monitoring: Counting people without people models or tracking," in *2008 IEEE Conference on Computer Vision and Pattern Recognition*, pp. 1–7, 2008.

[13] X. Liu, J. Van De Weijer, and A. D. Bagdanov, "Leveraging unlabeled data for crowd counting by learning to rank," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 7661–7669, 2018.

[14] X. Liu, J. Van De Weijer, and A. D. Bagdanov, "Exploiting unlabeled data in cnns by self-supervised learning to rank," *IEEE transactions on pattern analysis and machine intelligence*, vol. 41, no. 8, pp. 1862–1878, 2019.

[15] Y. Liu, M. Shi, Q. Zhao, and X. Wang, "Point in, box out: Beyond counting persons in crowds," 2019.

[16] D. B. Sam, N. N. Sajjan, H. Maurya, and R. V. Babu, "Almost unsupervised learning for dense crowd counting," in *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 33, pp. 8868–8875, 2019.

[17] R. Guerrero-Gómez-Olmedo, B. Torre-Jiménez, R. López-Sastre, S. Maldonado-Bascón, and D. Onoro-Rubio, "Extremely overlapping vehicle counting," in *Iberian Conference on Pattern Recognition and Image Analysis*, pp. 423–431, Springer, 2015.

[18] D. Onoro-Rubio and R. J. López-Sastre, "Towards perspective-free object counting with deep learning," in *European conference on computer vision*, pp. 615–629, Springer, 2016.

[19] E. Walach and L. Wolf, "Learning to count with cnn boosting," in *European conference on computer vision*, pp. 660–676, Springer, 2016.

[20] V. Lempitsky and A. Zisserman, "Learning to count objects in images," *Advances in neural information processing systems*, vol. 23, pp. 1324–1332, 2010.

[21] F. Gnädinger and U. Schmidhalter, "Digital counts of maize plants by unmanned aerial vehicles (uavs)," *Remote sensing*, vol. 9, no. 6, p. 544, 2017.

[22] W. Guo, B. Zheng, A. B. Potgieter, J. Diot, K. Watanabe, K. Noshita, D. R. Jordan, X. Wang, J. Watson, S. Ninomiya, *et al.*, "Aerial imagery analysis–quantifying appearance and number of sorghum heads for applications in breeding and agronomy," *Frontiers in plant science*, vol. 9, p. 1544, 2018.

[23] X. Jin, S. Liu, F. Baret, M. Hemerlé, and A. Comar, "Estimates of plant density of wheat crops at emergence from very low altitude uav imagery," *Remote Sensing of Environment*, vol. 198, pp. 105–114, 2017.

[24] Y. Gal, *Uncertainty in Deep Learning*. PhD thesis, University of Cambridge, 2016.

[25] D. Ryan, S. Denman, C. Fookes, and S. Sridharan, "Crowd counting using multiple local features," in *2009 Digital Image Computing: Techniques and Applications*, pp. 81–88, 2009.

[26] K. Chen, S. Gong, T. Xiang, and C. C. Loy, "Cumulative attribute space for age and crowd density estimation," in *2013 IEEE Conference on Computer Vision and Pattern Recognition*, pp. 2467–2474, 2013.

[27] M. Fu, P. Xu, X. Li, Q. Liu, M. Ye, and C. Zhu, "Fast crowd density estimation with convolutional neural networks," *Engineering Applications of Artificial Intelligence*, vol. 43, pp. 81–88, 2015.

[28] Y. Zhang, D. Zhou, S. Chen, S. Gao, and Y. Ma, "Single-image crowd counting via multi-column convolutional neural network," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 589–597, 2016.

[29] Y. Yang, G. Li, D. Du, Q. Huang, and N. Sebe, "Embedding perspective analysis into multi-column convolutional neural network for crowd counting," *IEEE Transactions on Image Processing*, vol. 30, pp. 1395–1407, 2021.

[30] V. A. Sindagi and V. M. Patel, "Ha-ccn: Hierarchical attention-based crowd counting network," *IEEE Transactions on Image Processing*, vol. 29, pp. 323–335, 2019.

[31] G. Gao, Q. Liu, and Y. Wang, "Counting dense objects in remote sensing images," 2020.

[32] F. Wang, M. Jiang, C. Qian, S. Yang, C. Li, H. Zhang, X. Wang, and X. Tang, "Residual attention network for image classification," 2017.

[33] J. Hu, L. Shen, S. Albanie, G. Sun, and E. Wu, "Squeeze-and-excitation networks," 2019.

[34] S. Woo, J. Park, J.-Y. Lee, and I. S. Kweon, "Cbam: Convolutional block attention module," in *Proceedings of the European conference on computer vision (ECCV)*, pp. 3–19, 2018.

[35] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," 2015.

[36] X. Fan, S. Zhang, B. Chen, and M. Zhou, "Bayesian attention modules," *arXiv preprint arXiv:2010.10604*, 2020.

[37] J. Gao, W. Lin, B. Zhao, D. Wang, C. Gao, and J. Wen, "C^3 framework: An open-source pytorch code for crowd counting," *arXiv preprint arXiv:1907.02724*, 2019.

[38] D. P. Kingma and J. Ba, "Adam: A method for stochastic optimization," *arXiv preprint arXiv:1412.6980*, 2014.