Locating Objects Without Bounding Boxes

Javier Ribera, David Güera, Yuhao Chen, Edward J. Delp
Video and Image Processing Laboratory (VIPER), Purdue University

Abstract

Recent advances in convolutional neural networks (CNN) have achieved remarkable results in locating objects in images. In these networks, the training procedure usually requires providing bounding boxes or the maximum number of expected objects. In this paper, we address the task of estimating object locations without annotated bounding boxes which are typically hand-drawn and time consuming to label. We propose a loss function that can be used in any fully convolutional network (FCN) to estimate object locations. This loss function is a modification of the average Hausdorff distance between two unordered sets of points. The proposed method has no notion of bounding boxes, region proposals, or sliding windows. We evaluate our method with three datasets designed to locate people's heads, pupil centers and plant centers. We outperform state-of-the-art generic object detectors and methods fine-tuned for pupil tracking.

1. Introduction

Locating objects in images is an important task in computer vision. A common approach in object detection is to obtain bounding boxes around the objects of interest. In this paper, we are not interested in obtaining bounding boxes. Instead, we define the object localization task as obtaining a single 2D coordinate corresponding to the location of each object. The location of an object can be any key point we are interested in, such as its center. Figure 1 shows an example of localized objects in images. Differently from other keypoint detection problems, we do not know in advance the number of keypoints in the image. To also make the method as generic as possible we do not assume any physical constraint between the points, unlike in cases such as pose estimation. This definition of object localization is more appropriate for applications where objects are very small, or substantially overlap (see the overlapping plants in Figure 1). In these cases, bounding boxes may not be provided by the dataset or they may be infeasible to groundtruth.

Bounding-box annotation is tedious, time-consuming and expensive [37]. For example, annotating ImageNet [43] required 42 seconds per bounding box when crowdsourcing on Amazon's Mechanical Turk using a technique specifically developed for efficient bounding box annotation [50]. In [6], Bell *et al*. introduce a new dataset for material recognition and segmentation. By collecting click location labels in this dataset instead of a full per-pixel segmentation, they reduce the annotation costs an order of magnitude.

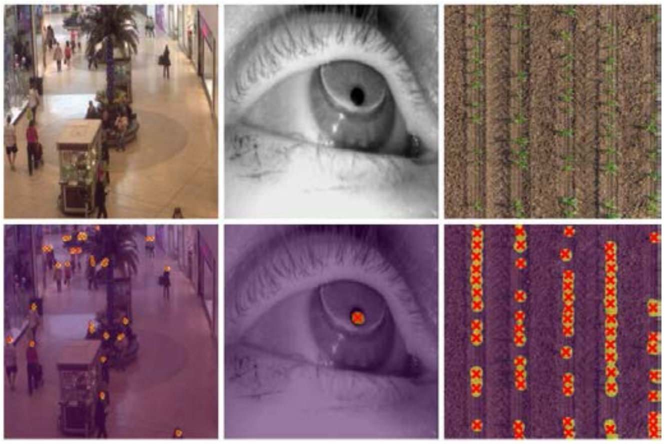

Figure 1. Object localization with human heads, eye pupils and plant centers. (Bottom) Heat map and estimations as crosses.

In this paper, we propose a modification of the average Hausdorff distance as a loss function of a CNN to estimate the location of objects. Our method does not require the use of bounding boxes in the training stage, and does not require to know the maximum number of objects when designing the network architecture. For simplicity, we describe our method only for a single class of objects, although it can trivially be extended to multiple object classes. Our method is object-agnostic, thus the discussion in this paper does not include any information about the object characteristics. Our approach maps input images to a set of coordinates, and we validate it with diverse types of objects. We evaluate our method with three datasets. One dataset contains images acquired from a surveillance camera in a shopping mall, and we locate the heads of people. The second dataset contains images of human eyes, and we locate the center of the pupil. The third dataset contains aerial images of a crop field taken from an Unmanned Aerial Vehicle (UAV), and we locate the centers of highly occluded plants.

Our approach to object localization via keypoint detection is not a universal drop-in replacement for bounding box detection, specially for those tasks that inherently require bounding boxes, such as automated cropping. Also, a limitation of this approach is that bounding box labeling incorporates some sense of scale, while keypoints do not.

The contributions of our work are:

- We propose a loss function for object localization, which we name *weighted Hausdorff distance* (WHD), that overcomes the limitations of pixelwise losses such as L^2 and the Hausdorff distances.
- We develop a method to estimate the location and number of objects in an image, without any notion of bounding boxes or region proposals.
- We formulate the object localization problem as the minimization of distances between points, independently of the model used in the estimation. This allows to use any fully convolutional network architectural design.
- We outperform state-of-the-art generic object detectors and achieve comparable results with crowd counting methods without any domain-specific knowledge, data augmentation, or transfer learning.

2. Related Work

Generic object detectors. Recent advances in deep learning [16, 27] have increased the accuracy of localization tasks such as object or keypoint detection. By generic object detectors, we mean methods that can be trained to detect any object type or types, such as Faster-RCNN [15], Single Shot MultiBox Detector (SSD) [31], or YOLO [40]. In Fast R-CNN, candidate regions or proposals are generated by classical methods such as selective search [59]. Although activations of the network are shared between region proposals, the system cannot be trained end-to-end. Region Proposal Networks (RPNs) in object detectors such as Faster R-CNN [15, 41] allow for end-to-end training of models. Mask R-CNN [18] extends Faster R-CNN by adding a branch for predicting an object mask but it runs in parallel with the existing branch for bounding box recognition. Mask R-CNN can estimate human pose keypoints by generating a segmentation mask with a single class indicating the presence of the keypoint. The loss function in Mask R-CNN is used location by location, making the keypoint detection highly sensitive to alignment of the segmentation mask. SDD provides fixed-sized bounding boxes and scores indicating the presence of an object in the boxes. The described methods either require groundtruthed bounding boxes to train the CNNs or require to set the maximum number of objects in the image being analyzed. In [19], it is observed that generic object detectors such as Faster R-CNN and SSD perform very poorly for small objects.

Counting and locating objects. Counting the number of objects in an image is not a trivial task. In [28], Lempitsky *et al.* estimate a density function whose integral corresponds to the object count. In [47], Shao *et al.* proposed two methods for locating objects. One method first counts and then locates, and the other first locates and then counts.

Locating and counting people is necessary for many applications such as crowd monitoring in surveillance systems, surveys for new businesses, and emergency management [28, 60]. There are multiple studies in the literature, where people in videos of crowds are detected and tracked [2, 7]. These detection methods often use bounding boxes around each human as ground truth. Acquiring bounding boxes for each person in a crowd can be labor intensive and imprecise under conditions where lots of people overlap, such as sports events or rush-hour agglomerations in public transport stations. More modern approaches avoid the need of bounding boxes by estimating a density map whose integral yields the total crowd count. In approaches that involve a density map, the label of the density map is constructed from the labels of the people's heads. This is typically done by centering Gaussian kernels at the location of each head. Zhang *et al.* [62] estimate the density image using a multi-column CNN that learns features at different scales. In [44], Sam *et al.* use multiple independent CNNs to predict the density map at different crowd densities. An additional CNN classifies the density of the crowd scene and relays the input image to the appropriate CNN. Huang *et al.* [20] propose to incorporate information about the body part structure to the conventional density map to reformulate the crowd counting as a multi-task problem. Other works such as Zhang *et al.* [61] use additional information such as the groundtruthed perspective map.

Methods for pupil tracking and precision agriculture are usually domain-specific. In pupil tracking, the center of the pupil must be resolved in images obtained in real-world illumination conditions [13]. A wide range of applications, from commercial applications such as video games [52], driving [48, 17] or microsurgery [14] rely on accurate pupil tracking. In remote precision agriculture, it is critical to locate the center of plants in a crop field. Agronomists use plant traits such as plant spacing to predict future crop yield [56, 51, 57, 12, 8], and plant scientists to breed new plant varieties [3, 35]. In [1], Aich *et al.* count wheat plants by first segmenting plant regions and then counting the number of plants in each segmented patch.

Hausdorff distance. The Hausdorff distance can be used to measure the distance between two sets of points [5]. Modifications of the Hausdorff distance [10] have been used for various multiple tasks, including character recognition [33], face recognition [23] and scene matching [23]. Schutze *et al.* [46] use the average Hausdorff distance to evaluate solutions in multi-objective optimization problems. In [24], Elkhiyari *et al.* compare features extracted by a CNN according to multiple variants of the Hausdorff distance for the task of face recognition. In [11], Fan *et al.* use the Chamfer and Earth Mover's distance, along with a new neural network architecture, for 3D object reconstruction by estimating the location of a fixed number of points. The Hausdorff distance is also a common metric to evaluate the quality of segmentation boundaries in the medical imaging community [54, 63, 30, 55].

3. The Average Hausdorff Distance

Our work is based on the Hausdorff distance which we briefly review in this section. Consider two unordered non-empty sets of points X and Y and a distance metric $d(x, y)$ between two points $x \in X$ and $y \in Y$. The function $d(\cdot, \cdot)$ could be any metric. In our case we use the Euclidean distance. The sets X and Y may have different number of points. Let $\Omega \subset \mathbb{R}^2$ be the space of all possible points. In its general form, the Hausdorff distance between $X \subset \Omega$ and $Y \subset \Omega$ is defined as

$$d_{\mathrm{H}}(X, Y) = \max\left\{\sup_{x\in X} \inf_{y\in Y} d(x, y), \sup_{y\in Y} \inf_{x\in X} d(x, y)\right\}. \tag{1}$$

When considering a discretized and bounded Ω, such as all the possible pixel coordinates in an image, the suprema and infima are achievable and become maxima and minima, respectively. This bounds the Hausdorff distance as

$$d(X, Y) \leq d_{max} = \max_{x\in\Omega, y\in\Omega} d(x, y), \tag{2}$$

which corresponds to the diagonal of the image when using the Euclidean distance. As shown in [5], the Hausdorff distance is a metric. Thus $\forall X, Y, Z \subset \Omega$ we have the following properties:

$$d_H(X, Y) \geq 0 \tag{3a}$$

$$d_H(X, Y) = 0 \iff X = Y \tag{3b}$$

$$d_H(X, Y) = d_H(Y, X) \tag{3c}$$

$$d_H(X, Y) \leq d_H(X, Z) + d_H(Z, Y) \tag{3d}$$

Equation (3b) follows from X and Y being closed, because in our task the pixel coordinate space Ω is discretized. These properties are very desirable when designing a function to measure how similar X and Y are [4].

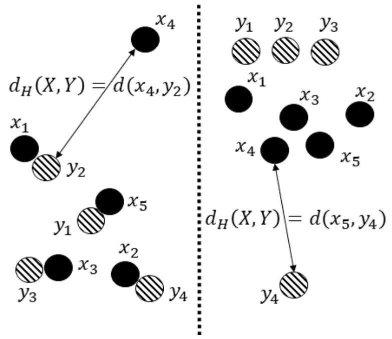

Figure 2. Illustration of two different configurations of point sets $X = \{x_1, ..., x_5\}$ (solid dots) and $Y = \{y_1, ..., y_4\}$ (dashed dots). Despite the clear difference in the distances between points, their Hausdorff distance are equal because the worst outlier is the same.

A shortcoming of the Hausdorff function is its high sensitivity to outliers [46, 54]. Figure 2 shows an example for two finite sets of points with one outlier. To avoid this, the average Hausdorff distance is more commonly used:

$$d_{\mathrm{AH}}(X, Y) = \frac{1}{|X|} \sum_{x\in X} \min_{y\in Y} d(x, y) + \frac{1}{|Y|} \sum_{y\in Y} \min_{x\in X} d(x, y), \tag{4}$$

where $|X|$ and $|Y|$ are the number of points in X and Y, respectively. Note that properties (3a), (3b) and (3c) are still true, but (3d) is not. Also, the average Hausdorff distance is differentiable with respect to any point in X or Y.

Let Y contain the ground truth pixel coordinates, and X be our estimation. Ideally, we would like to use $d_{\mathrm{AH}}(X, Y)$ as the loss function during the training of our convolutional neural network (CNN). We find two limitations when incorporating the average Hausdorff distance as a loss function. First, CNNs with linear layers implicitly determine the estimated number of points $|X|$ as the size of the last layer. This is a drawback because the actual number of points depends on the content of the image itself. Second, FCNs such as U-Net [42] can indicate the presence of an object center with a higher activation in the output layer, but they do not return the pixel coordinates. In order to learn with backpropagation, the loss function must be differentiable with respect to the network output.

4. The Weighted Hausdorff Distance

To overcome these two limitations, we modify the average Hausdorff distance as follows:

$$\begin{aligned} d_{\mathrm{WH}}(p, Y) = &\frac{1}{\mathcal{S} + \epsilon} \sum_{x\in\Omega} p_x \min_{y\in Y} d(x, y) + \\ &\frac{1}{|Y|} \sum_{y\in Y} \underset{x\in\Omega}{M_\alpha} \left[\, p_x d(x, y) + (1 - p_x) d_{max} \,\right], \end{aligned} \tag{5}$$

where

$$\mathcal{S} = \sum_{x\in\Omega} p_x, \tag{6}$$

$$\underset{a\in A}{M_\alpha}\left[f(a)\right] = \left(\frac{1}{|A|}\sum_{a\in A} f^\alpha(a)\right)^{\frac{1}{\alpha}}, \tag{7}$$

is the generalized mean, and ϵ is set to 10^{-6}. We call $d_{\text{WH}}(p, Y)$ the weighted Hausdorff distance (WHD). $p_x \in [0,1]$ is the single-valued output of the network at pixel coordinate x. The last activation of the network can be bounded between zero and one by using a sigmoid non-linearity. Note that p does not need to be normalized, i.e., $\sum_{x\in\Omega} p_x = 1$ is not necessary. Note that the generalized mean $M_\alpha[\cdot]$ corresponds to the minimum function when $\alpha = -\infty$. We justify the modifications applied to Equation (4) to obtain Equation (5) as follows:

1. The ϵ in the denominator of the first term provides numerical stability when $p_x \approx 0 \; \forall x \in \Omega$.

2. When $p_x = \{0,1\}$, $\alpha = -\infty$, and $\epsilon = 0$, the weighted Hausdorff distance becomes the average Hausdorff distance. We can interpret this as the network indicating with complete certainty where the object centers are. As $d_{\text{WH}}(p, Y) \geq 0$, the global minimum ($d_{\text{WH}}(p, Y) = 0$) corresponds to $p_x = 1$ if $x \in Y$ and 0 otherwise.

3. In the first term, we multiply by p_x to penalize high activations in areas of the image where there is no ground truth point y nearby. In other words, the loss function penalizes estimated points that should not be there.

4. In the second term, by using the expression $f(\cdot) := p_x d(x,y) + (1-p_x)d_{max}$ we enforce that

   (a) If $p_{x_0} \approx 1$, then $f(\cdot) \approx d(x_0, y)$. This means the point x_0 will contribute to the loss as in the AHD (Equation (4)).

   (b) If $p_{x_0} \approx 0$, $x_0 \neq y$, then $f(\cdot) \approx d_{max}$. Then, if $\alpha = -\infty$, the point x_0 will not contribute to the loss because the "minimum" $M_{x\in\Omega}[\cdot]$ will ignore x_0. If another point x_1 closer to y with $p_{x_1} > 0$ exists, x_1 will be "selected" instead by $M[\cdot]$. Otherwise $M_{x\in\Omega}[\cdot]$ will be high. This means that low activations around ground truth points will be penalized.

   Note that $f(\cdot)$ is not the only expression that would enforce these two constraints ($f|_{p_x=1} = d(x,y)$ and $f|_{p_x=0} = d_{max}$). We chose a linear function because of its simplicity and numerical stability.

Both terms in the WHD are necessary. If the first term is removed, then the trivial solution is $p_x = 1 \quad \forall x \in \Omega$. If the second term is removed, then the trivial solution is $p_x = 0 \quad \forall x \in \Omega$. These two cases hold for any value of α and the proof can be found in the suplemental material. Ideally, the parameter $\alpha \to -\infty$ so that $M_\alpha(\cdot) = ||\cdot||_{-\infty}$ becomes the minimum operator [26]. However, this would make the second term flat with respect to the output of the network. For a given y, changes in p_{x_0} in a point x_0 that is far from y would be ignored by $M_{-\infty}(\cdot)$, if there is another point x_1 with high activation and closer to y. In practice, this makes training difficult because the minimum is not a smooth function with respect to its inputs. Thus, we approximate the minimum with the generalized mean $M_\alpha(\cdot)$, with $\alpha < 0$. The more negative α is, the more similar to the AHD the WHD becomes, at the expense of becoming less smooth. In our experiments, $\alpha = -1$. There is no need to use $M_\alpha(\cdot)$ in the first term because p_x is not inside the minimum, thus the term is already differentiable with respect to p.

If the input image needs to be resized to be fed into the network, we can normalize the WHD to account for this distortion. Denote the original image size as $(S_o^{(1)}, S_o^{(2)})$ and the resized image size as $(S_r^{(1)}, S_r^{(2)})$. In Equation (5), we compute distances in the original pixel space by replacing $d(x,y)$ with $d(\mathbf{S}x, \mathbf{S}y)$, where $x, y \in \Omega$ and

$$\mathbf{S} = \begin{pmatrix} S_o^{(1)}/S_r^{(1)} & 0 \\ 0 & S_o^{(2)}/S_r^{(2)} \end{pmatrix}. \tag{8}$$

4.1. Advantage Over Pixelwise Losses

A naive alternative is to use a one-hot map as label, defined as $l_x = 1$ for $x \in Y$ and $l_x = 0$ otherwise, and then use a pixelwise loss such as the Mean Squared Error (MSE) or the L^2 norm, where $L^2(l,p) = \sum_{\forall x\in\Omega} |p_x - l_x|^2 \propto \text{MSE}(l,x)$. The issue with pixelwise losses is that they are not informative of how close two points $x \in \Omega$ and $y \in Y$ are unless $x = y$. In other words, it is flat for the vast majority of the pixels, making training unfeasible. This issue is locally mitigated in [58] by using the MSE loss with Gaussians centered at each $x \in Y$. By contrast, the WHD in Equation (5) will decrease the closer x is to y, making the loss function informative outside of the global minimum.

5. CNN Architecture And Location Estimation

In this section, we describe the architecture of the fully convolutional network (FCN) we use, and how we estimate the final object locations. We want to emphasize that the network design is not a meaningful contribution of this work, thus we have not made any attempt to optimize it. Our main contribution is the use of the weighted Hausdorff distance as the loss function. We adopt the U-Net architecture [42] and modify it minimally for this task. Networks similar to U-Net have been proven to be capable of accurately mapping the input image into an output image, when trained in a conditional adversarial network setting [22] or when using a carefully tuned loss function [42]. Figure 3 shows the

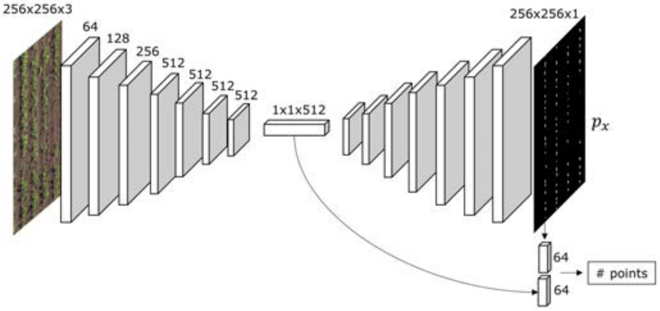

Figure 3. The FCN architecture used for object localization, minimally adapted from the U-Net [42] architecture. We add a small fully-connected layer that combines the deepest features and the estimated probability map to regress the number of points.

hourglass design of U-Net. The residuals connections between each layer in the encoder and its symmetric layer in the decoder are not shown for simplicity.

This FCN has two well differentiated blocks. The first block follows the typical architecture of a CNN. It consists of the repeated application of two 3×3 convolutions (with padding 1), each followed by a batch normalization operation and a Rectified Linear Unit (ReLU). After the ReLU, we apply a 2×2 max pooling operation with stride 2 for downsampling. At each downsampling step we double the number of feature channels, starting with 64 channels and using 512 channels for the last 5 layers.

The second block consists of repeated applications of the following elements: a bilinear upsampling, a concatenation with the feature map from the downsampling block, and two 3×3 convolutions, each followed by a batch normalization and a ReLU. The final layer is a convolution layer that maps to the single-channel output of the network, p.

To estimate the number of objects in the image, we add a branch that combines the information from the deepest level features and also from the estimated probability map. This branch combines both features (the $1 \times 1 \times 512$ feature vector and the 256×256 probability map) into a hidden layer, and uses the 128-dimensional feature vector to output a single number. We then apply a ReLU to ensure the output is positive, and round it to the closest integer to obtain our final estimate of the number of objects, $\hat{C}$.

Although we use this particular network architecture, any other architecture could be used. The only requirement is that the output images of the network must be of the same size as the input image. The choice of a FCN arises from the natural interpretation of its output as the weights (p_x) in the WHD (Equation (5)). In previous works [24, 11], variants of the average Haussdorf distance were successfully used with non-FCN networks that estimate the point set directly. However, in those cases the size of the estimated set is fixed by the size of the last layer. To locate an unknown number of objects, the network must be able to estimate a variable number of object locations. Thus, we could envision the WHD also being used in non-FCN networks as long as the output of the network is used as p in Equation (5).

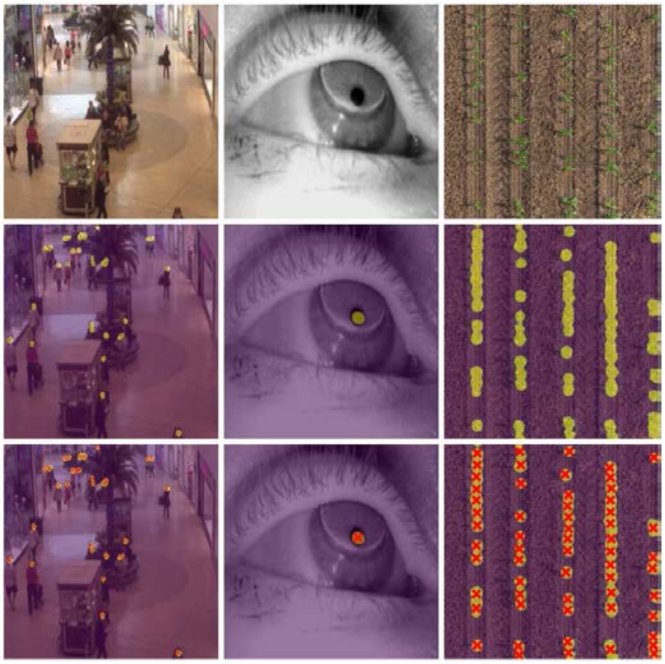

Figure 4. First row: Input image. Second row: Output of the network (p in the text) overlaid onto the input image. This can be considered a saliency map of object locations. Third row: The estimated object locations are marked with a red cross.

The training loss we use to train the network is a combination of Equation (5) and a smooth L_1 loss for the regression of the object count. The final training loss is

$$\mathcal{L}(p, Y) = d_{WH}(p, Y) + \mathcal{L}_{\text{reg}}(C - \hat{C}(p)), \qquad (9)$$

where Y is the set containing the ground truth coordinates of the objects in the image, p is the output of the network, $C = |Y|$, and $\hat{C}(p)$ is the estimated number of objects. $\mathcal{L}_{\text{reg}}(\cdot)$ is the regression term, for which we use the smooth L_1 or Huber loss [21], defined as

$$\mathcal{L}_{\text{reg}}(x) = \begin{cases} 0.5x^2, & \text{for} |x| < 1 \\ |x| - 0.5, & \text{for} |x| \geq 1 \end{cases} \qquad (10)$$

This loss is robust to outliers when the regression error is high, and at the same time is differentiable at the origin.

The network outputs a saliency map p indicating with $p_x \in [0, 1]$ the confidence that there is an object at pixel x. Figure 4 shows p in the second row. During evaluation, our ultimate goal is to obtain $\hat{Y}$, i. e., the estimate of all object locations. In order to convert p to $\hat{Y}$, we threshold p to obtain the pixels $T = \{x \in \Omega \mid p_x > \tau\}$. We can use three different methods to decide which τ to use:

1. Use a constant τ for all images.

2. Use Otsu thresholding [36] to find an adaptive τ different for every image.

3. Use a Beta mixture model-based thresholding (BMM). This method fits a mixture of two Beta distributions to the values of p using the algorithm described in [45], and then takes the mean value of the distribution with highest mean as τ.

Figure 4 shows in the third row an example of the result of thresholding the saliency map p. Then, we fit a Gaussian mixture model to the points T. This is done using the expectation maximization (EM) [34] algorithm and the estimated number of plants $\hat{C}$.

The means of the fitted Gaussians are considered the final estimate $\hat{Y}$. The third row of Figure 4 shows the estimated object locations with red crosses. Note that even if the map produced by the FCN is of good quality, i.e., there is a cluster on each object location, EM may not yield the correct object locations if $|\hat{C} - C| > 0.5$. An example can be observed in the first column of Figure 4, where a single head is erroneously estimated as two heads.

6. Experimental Results

We evaluate our method with three datasets.

The first dataset consists of 2,000 images acquired from a surveillance camera in a shopping mall. It contains annotated locations of the heads of the crowd. This dataset is publicly available at `http://personal.ie.cuhk.edu.hk/~ccloy/downloads_mall_dataset.html` [32]. 80%, 10% and 10% of the images were randomly assinged to the training, validation, and testing datasets, respectively.

The second dataset is presented in [13] with the roman letter V and publicly available at `http://www.ti.uni-tuebingen.de/Pupil-detection.1827.0.html`. It contains 2,135 images with a single eye, and the goal is to detect the center of the pupil. It was also randomly split into training, validation and testing datasets as 80/10/10 %, respectively.

The third dataset consists of aerial images of a crop field taken from a UAV flying at an altitude of 40 m. The images were stitched together to generate a $6,000 \times 12,000$ orthoimage of 0.75 cm/pixel resolution shown in Figure 5. The location of the center of all plants in this image was groundtruthed, resulting in a total of 15,208 unique plant centers. This mosaic image was split, and the left 80% area was used for training, the middle 10% for validation, and the right 10% for testing. Within each region, random image crops were generated. These random crops have a uniformly distributed height and width between 100 and 600 pixels. We extracted 50,000 random image crops in the training region, $5,000$ in the validation region, and $5,000$ in the testing region. Note that some of these crops may highly overlap. We are making the third dataset publicly available at `https://engineering.purdue.edu/~sorghum/dataset-plant-centers-2016`. We believe this dataset will be valuable for the community, as it poses a challenge due to the high occlusion between plants.

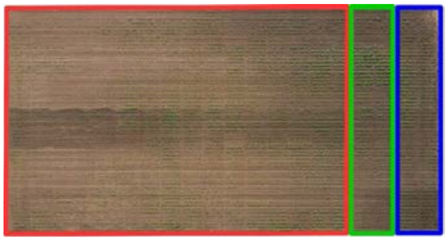

Figure 5. An orthorectified image of a crop field with 15,208 plants. The red region was used for training, the region in green for validation, and the region in blue for testing.

All the images were resized to 256×256 because that is the minimum size our architecture allows. The groundtruthed object locations were also scaled accordingly. As for data augmentation, we only use random horizontal flip. For the plant dataset, we also flipped the images vertically. We set $\alpha = -1$ in Equation (7). We have also experimented with $\alpha = -2$ with no apparent improvement, but we did not attempt to find an optimal value. We retrain the network for every dataset, i.e., we do not use pretrained weights. For the mall and plant dataset, we used a batch size of 32 and Adam optimizer [25, 39] with a learning rate of 10^{-4} and momentum of 0.9. For the pupil dataset, we reduced the size of the network by removing the five central layers, we used a batch size of 64, and stochastic gradient descent with a learning rate of 10^{-3} and momentum of 0.9. At the end of each epoch, we evaluate the average Haussdorf distance (AHD) in Equation (4) over the validation set, and select the epoch with lowest AHD on validation.

As metrics, we report Precision, Recall, F-score, AHD, Mean Absolute Error (MAE), Root Mean Squared Error (RMSE), and Mean Absolute Percent Error (MAPE):

$$\text{MAE} = \frac{1}{N}\sum_{i=1}^{N} |e_i|, \quad \text{RMSE} = \sqrt{\frac{1}{N}\sum_{i=1}^{N} |e_i|^2} \qquad (11)$$

$$\text{MAPE} = 100\frac{1}{N}\sum_{\substack{i=1\\ C_i \neq 0}}^{N} \frac{|e_i|}{C_i} \qquad (12)$$

where $e_i = \hat{C}_i - C_i$, N is the number of images, C_i is the true object count in the i-th image, and $\hat{C}_i$ is our estimate.

A true positive is counted if an estimated location is at most at distance r from a ground truth point. A false positive is counted if an estimated location does not have any ground truth point at a distance at most r. A false negative is counted if a true location does have any estimated location at a distance at most r. Precision is the proportion of our estimated points that are close enough to a true point. Recall is the proportion of the true points that we are able to detect. The F-score is the harmonic mean of precision and recall. Note that one can achieve a precision and recall of 100% even if we estimate more than one object location per ground truth point. This would not be an ideal localization. To take this into account, we also report metrics (MAE, RMSE and MAPE) that indicate if the number of objects is incorrect. The AHD can be interpreted as the average location error in pixels.

Figure 8 shows the F-score as a function of r. Note that r is only an evaluation parameter. It is not needed during training or testing. MAE, RMSE, and MAPE are shown in Table 1. Note that we are using the same architecture for all tasks, except for the pupil dataset, where we removed intermediate layers. Also, in the case of the pupil detection, we know that there is always one object in the image. Thus, regression is not necessary and we can remove the regression term in Equation (9) and fix $\hat{C}_i = C_i = 1 \ \forall i$.

A naive alternative approach to object localization would be to use generic object detectors such as Faster R-CNN [41]. One can train these detectors by constructing bounding boxes with fixed size centered at each labeled point. Then the center of each bounding box can be taken as the estimated location. We used bounding boxes of size 20×20 (the approximate average head and pupil size) and anchor sizes of 16×16 and 32×32. Note that these parameters may be suboptimal even though they were selected to match the type of object. The threshold we used for the softmax scores was 0.5 and for the intersection over union it was 0.4, because they minimize the AHD over the validation set. We used the VGG-16 architecture [49] and trained it using stochastic gradient descent with learning rate of 10^{-3} and momentum of 0.9. For the pupil dataset, we always selected the bounding box with the highest score. We experimentally observed that Faster R-CNN struggles with detecting very small objects that are very close to each other. Tables 2-4 show the results of Faster R-CNN results on the mall, pupil, and plant datasets. Note that the mall and plant datasets, with many small and highly overlapping objects, are the most challenging for Faster R-CNN. This behaviour is consistent with the observations in [19], where, all generic object detectors perform very poorly and Faster R-CNN yields a mean Average Precision (mAP) of 5% in the best case.

We also experimented using mean shift [9] instead of Gaussian mixtures (GM) to detect the local maxima. However, mean shift is prone to detect multiple local maxima, and GMs are more robust against outliers. In our experiments, we observed that precision and recall were substantially worse than using GM. More importantly, using Mean Shift slowed down validation an order of magnitude. The average time for the Mean Shift algorithm to run on one of our images was 12 seconds, while fitting GM using expectation maximization took around 0.5 seconds, when using the scikit-learn implementations [38].

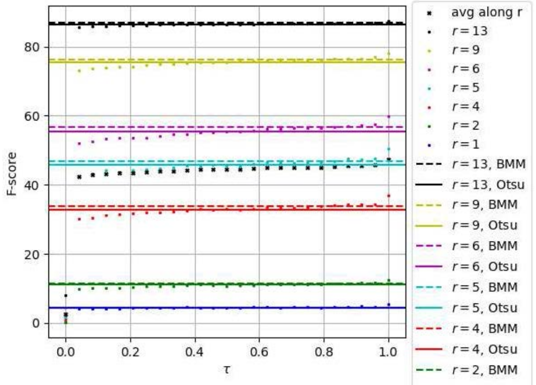

Figure 6. Effect on the F-score of the threshold τ.

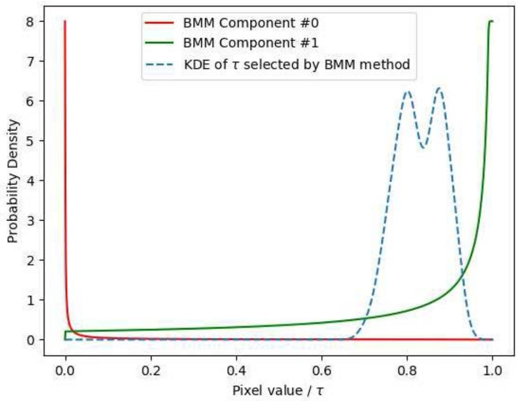

Figure 7. Beta mixture model fitted on the values of p_x, and the thresholds τ used by the BMM method.

We also investigated the effect of the parameter τ, and the three methods to select it presented in Section 5. One may think that this parameter could be a trade-off between some metrics, and that it should be cross-validated. In practice, we observed that τ does not balance precision and recall, thus a precision-recall curve is not meaningful. Instead, we plot the F-score as a function of r in Figure 8. Also, cross-validating τ would imply fixing an "optimal" value for all images. Figure 6 shows that we can do better with adaptive thresholding methods (Otsu or BMM). Note that BMM thresholding (dashed lines) always outperforms Otsu (solid lines), and most of fixed τ. To justify the appropriateness of the BMM method, note that in Figure 4 most of the values in the estimated map are very high or very low. This makes a Beta distribution a better fit than a Normal distribution (as used in Otsu's method) to model p_x. Figure 7 shows the fitted BMM and a kernel density estimation of the values of τ adaptively selected by the BMM method.

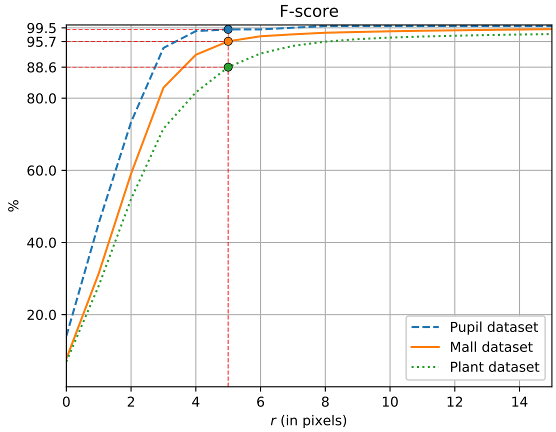

Figure 8. F-score as a function of r, the maximum distance between a true and an estimated object location to consider it correct or incorrect. A higher r makes correctly locating an object easier.

Table 1. Results of our method for object localization, using $r = 5$. Metrics are defined in Equations (4), (11)-(12). Regression metrics for the pupil dataset are not shown because there is always a single pupil ($\hat{C} = C = 1$). Figure 8 shows the F-score for other r values.

Metric	Mall dataset	Pupil dataset	Plant dataset	Average
Precision	95.2%	99.5%	88.1%	94.4%
Recall	96.2%	99.5%	89.2%	95.0%
F-score	95.7%	99.5%	88.6%	94.6%
AHD	4.5 px	2.5 px	7.1 px	4.7 px
MAE	1.4	-	1.9	1.7
RMSE	1.8	-	2.7	2.3
MAPE	4.4%	-	4.2%	4.3 %

Lastly, as our method locates and counts objects simultaneously, it could be used as a counting technique. We also evaluated our technique in the task of crowd counting using the ShanghaiTech Part B dataset presented in [62], and achieve a MAE of 19.9. Even though we do not outperform state of the art methods that are specifically fine-tuned for crowd counting [29], we can achieve comparable results with our generic method. We expect future improvements such as architectural changes or using transfer learning to further increase the performance.

A PyTorch implementation of the weighted Hausdorff distance loss and trained models are available at `https://github.com/javiribera/locating-objects-without-bboxes`.

Table 2. Head location results using the mall dataset, using $r = 5$.

Metric	Faster-RCNN	Ours
Precision	81.1%	**95.2 %**
Recall	76.7%	**96.2 %**
F-score	78.8 %	**95.7 %**
AHD	7.6 px	**4.5 px**
MAE	4.7	**1.4**
RMSE	5.6	**1.8**
MAPE	14.8%	**4.4 %**

Table 3. Pupil detection results, using $r = 5$. Precision and recall are equal because there is only one estimated and one true object.

Method	Precision	Recall	AHD
Swirski [53]	77 %	77 %	-
ExCuSe [13]	77 %	77 %	-
Faster-RCNN	99.5 %	99.5 %	2.7 px
Ours	**99.5 %**	**99.5 %**	**2.5 px**

Table 4. Plant location results using the plant dataset, using $r = 5$.

Metric	Faster-RCNN	Ours
Precision	86.6 %	**88.1 %**
Recall	78.3 %	**89.2 %**
F-score	82.2 %	**88.6 %**
AHD	9.0 px	**7.1 px**
MAE	9.4	**1.9**
RMSE	13.4	**2.7**
MAPE	17.7 %	**4.2 %**

7. Conclusion

We have presented a loss function for the task of locating objects in images that does not need bounding boxes. This loss function is a modification of the average Hausdorff distance (AHD), which measures the similarity between two unordered sets of points. To make the AHD differentiable with respect to the network output, we have considered the certainty of the network when estimating an object location. The output of the network is a saliency map of object locations and the estimated number of objects. Our method is not restricted to a maximum number of objects in the image, does not require bounding boxes, and does not use region proposals or sliding windows. This approach can be used in tasks where bounding boxes are not available, or the small size of objects makes the labeling of bounding boxes impractical. We have evaluated our approach with three different datasets, and outperform generic object detectors and task-specific techniques. Future work will include developing a multi-class object location estimator in a single network, and evaluating more modern CNN architectures.

Acknowledgements: This work was funded by the Advanced Research Projects Agency-Energy (ARPA-E), U.S. Department of Energy, under Award Number DE-AR0000593. The views and opinions of the authors expressed herein do not necessarily reflect those of the U.S. Government or any agency thereof. We thank Professor Ayman Habib for the orthophotos used in this paper. Contact information: Edward J. Delp, `ace@ecn.purdue.edu`

References

[1] S. Aich, I. Ahmed, I. Obsyannikov, I. Stavness, A. Josuttes, K. Strueby, H. Duddu, C. Pozniak, and S. Shirtliffe. Deepwheat: Estimating phenotypic traits from crop images with deep learning. *Proceedings of the IEEE Winter Conference on Applications of Computer Vision*, March 2018. Stateline, NV.

[2] M. Andriluka, S. Roth, and B. Schiele. People-tracking-by-detection and people-detection-by-tracking. *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, June 2008. Anchorage, AK.

[3] J. L. Araus and J. E. Cairns. Field high-throughput phenotyping: the new crop breeding frontier. *Trends in Plant Science*, 19(1):52–61, January 2014.

[4] E. M. Arkin, L. P. Chew, D. P. Huttenlocher, K. Kedem, and J. S. Mitchell. An efficiently computable metric for comparing polygonal shapes. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 13(3), March 1991.

[5] H. Attouch, R. Lucchetti, and R. J. B. Wets. The topology of the ρ-Hausdorff distance. *Annali di Matematica Pura ed Applicata*, 160(1):303–320, December 1991.

[6] S. Bell, P. Upchurch, N. Snavely, and K. Bala. Material recognition in the wild with the materials in context database (supplemental material). *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, June 2015. Boston, MA.

[7] M. D. Breitenstein, F. Reichlin, B. Leibe, E. Koller-Meier, and L. V. Gool. Online multiperson tracking-by-detection from a single, uncalibrated camera. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 33(9):1820–1833, 2011.

[8] B. S. Chauhan and D. E. Johnson. Row spacing and weed control timing affect yield of aerobic rice. *Field Crops Research*, 121(2):226–231, March 2001.

[9] D. Comaniciu and P. Meer. Mean shift: A robust approach toward feature space analysis. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 24(5):603–619, 2002.

[10] M.-P. Dubuisson and A. K. Jain. A modified Hausdorff distance for object matching. *Pattern Recognition*, pages 566–568, October 1994.

[11] H. Fan, H. Su, and L. Guibas. A point set generation network for 3D object reconstruction from a single image. *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 2463–2471, July 2017. Honolulu, HI.

[12] D. E. Farnham. Row spacing, plant density, and hybrid effects on corn grain yield and moisture. *Agronomy Journal*, 93:1049–1053, September 2001.

[13] W. Fuhl, T. Kübler, K. Sippel, W. Rosenstiel, and E. Kasneci. ExCuSe: Robust pupil detection in real-world scenarios. *Proceedings of the International Conference on Computer Analysis of Images and Patterns*, pages 39–51, September 2015. Valletta, Malta.

[14] W. Fuhl, T. Santini, C. Reichert, D. Claus, A. Herkommer, H. Bahmani, K. Rifai, S. Wahl, and E. Kasneci. Non-intrusive practitioner pupil detection for unmodified microscope oculars. *Computers in Biology and Medicine*, 79:36–44, December 2016.

[15] R. Girshick. Fast R-CNN. *Proceedings of the IEEE International Conference on Computer Vision*, pages 1440–1448, December 2015.

[16] I. Goodfellow, Y. Bengio, and A. Courville. *Deep Learning*. MIT Press, November 2016.

[17] J. Gu, X. Yang, S. De Mello, and J. Kautz. Dynamic facial analysis: From bayesian filtering to recurrent neural network. *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 1548–1557, July 2017. Honolulu, HI.

[18] K. He, G. Gkioxari, P. Dollár, and R. Girshick. Mask R-CNN. *arXiv:1703.06870*, April 2017.

[19] J. Huang, V. Rathod, C. Sun, M. Zhu, A. Korattikara, A. Fathi, I. Fischer, Z. Wojna, Y. Song, S. Guadarrama, and K. Murphy. Speed/accuracy trade-offs for modern convolutional object detectors. *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, July 2017. Honolulu, HI.

[20] S. Huang, X. Li, Z. Zhang, F. Wu, S. Gao, R. Ji, and J. Han. Body structure aware deep crowd counting. *IEEE Transactions on Image Processing*, 27(3):1049–1059, March 2018.

[21] P. J. Huber. Robust estimation of a location parameter. *The Annals of Mathematical Statistics*, pages 73–101, 1964.

[22] P. Isola, J.-Y. Zhu, T. Zhou, and A. A. Efros. Image-to-image translation with conditional adversarial networks. *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, July 2017. Honolulu, HI.

[23] K. L. K. Lin and W. Siu. Spatially eigen-weighted Hausdorff distances for human face recognition. *Pattern Recognition*, 36(8):1827–1834, August 2003.

[24] H. E. Khiyari and H. Wechsler. Age invariant face recognition using convolutional neural networks and set distances. *Journal of Information Security*, 8(3):174–185, July 2017.

[25] D. P. Kingma and J. Ba. Adam: A method for stochastic optimization. *Proceedings of the International Conference for Learning Representations*, abs/1412.6980, April 2015. San Diego, CA.

[26] C. S. Kubrusly. Banach spaces L^p. In *Essentials of Measure Theory*, page 83. Springer, Cham, 2005.

[27] Y. LeCun, Y. Bengio, and G. Hinton. Deep learning. *Nature*, 521:436–444, May 2015.

[28] V. Lempitsky and A. Zisserman. Learning to count objects in images. *Proceedings of the Advances in Neural Information Processing Systems*, pages 1324–1332, December 2010. Vancouver, Canada.

[29] Y. Li, X. Zhang, and D. Chen. CSRNet: Dilated convolutional neural networks for understanding the highly congested scenes. *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 1091–1100, June 2018.

[30] S. Liao, Y. Gao, A. Oto, and D. Shen. Representation learning: A unified deep learning framework for automatic prostate mr segmentation. *Proceedings of the Medical Image Computing and Computer-Assisted Intervention*, pages 254–261, September 2013. Nagoya, Japan.

[31] W. Liu, D. Anguelov, D. Erhan, C. Szegedy, S. Reed, C. Fu, and A. C. Berg. SSD: Single shot multibox detector. *Proceedings of the European Conference on Computer Vision*, pages 21–37, October 2016. Amsterdam, The Netherlands.

[32] C. C. Loy, K. Chen, S. Gong, and T. Xiang. Crowd counting and profiling: Methodology and evaluation. In *Modeling, Simulation and Visual Analysis of Crowds*, pages 347–382. Springer, October 2013.

[33] Y. Lu, C. L. Tan, W. Huang, and L. Fan. An approach to word image matching based on weighted Hausdorff distance. *Proceedings of International Conference on Document Analysis and Recognition*, pages 921–925, September 2001.

[34] T. K. Moon. The expectation-maximization algorithm. *IEEE Signal Processing Magazine*, 13(6):47–60, November 1996.

[35] E. H. Neilson, A. M. Edwards, C. K. Blomstedt, B. Berger, B. L. Mller, and R. M. Gleadow. Utilization of a high-throughput shoot imaging system to examine the dynamic phenotypic responses of a C_4 cereal crop plant to nitrogen and water deficiency over time. *Journal of Experimental Botany*, 66(7):1817–1832, 2015.

[36] N. Otsu. A threshold selection method from gray-level histograms. *IEEE Transactions on Systems, Man, and Cybernetics*, 9(1):62–66, January 1979.

[37] D. P. Papadopoulos, J. R. R. Uijlings, F. Keller, and V. Ferrari. We don't need no bounding-boxes: Training object class detectors using only human verification. *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 854–863, June 2016. Las Vegas, NV.

[38] F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, V. Dubourg, J. Vanderplas, A. Passos, D. Cournapeau, M. Brucher, M. Perrot, and E. Duchesnay. Scikit-learn: Machine learning in Python. *Journal of Machine Learning Research*, 12:2825–2830, 2011.

[39] S. J. Reddi, S. Kale, and S. Kumar. On the convergence of adam and beyond. *Proceedings of the International Conference on Learning Representations*, April 2018. Vancouver, Canada.

[40] J. Redmon, S. Divvala, R. Girshick, and A. Farhadi. You only look once: Unified, real-time object detection. *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 779–788, June 2016. Las Vegas, NV.

[41] S. Ren, K. He, R. Girshick, and J. Sun. Faster R-CNN: Towards real-time object detection with region proposal networks. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 36(6):1137–1149, June 2017.

[42] O. Ronneberger, P. Fischer, and T. Brox. U-Net: Convolutional networks for biomedical image segmentation. *Proceedings of the International Conference on Medical Image Computing and Computer-Assisted Intervention*, pages 234–241, October 2015. Munich, Germany.

[43] O. Russakovsky, J. Deng, H. Su, J. Krause, S. Satheesh, S. Ma, Z. Huang, A. Karpathy, A. Khosla, M. Bernstein, A. C. Berg, and L. Fei-Fei. Imagenet large scale visual recognition challenge. *International Journal of Computer Vision*, 11(3):211–252, December 2015.

[44] D. B. Sam, S. Surya, and R. V. Babu. Switching convolutional neural network for crowd counting. *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 4031–4039, July 2017.

[45] C. Schröder. A hybrid parameter estimation algorithm for beta mixtures and applications to methylation state classification. *Algorithms for Molecular Biology*, 12(21):62–66, August 2017.

[46] O. Schutze, X. Esquivel, A. Lara, and C. A. C. Coello. Using the averaged Hausdorff distance as a performance measure in evolutionary multiobjective opti-

mization. *IEEE Transactions on Evolutionary Computation*, 16(4):504–522, August 2012.

[47] J. Shao, D. Wang, X. Xue, and Z. Zhang. Learning to point and count. *arXiv:1512.02326*, December 2015.

[48] A. Shrivastava, T. Pfister, O. Tuzel, J. Susskind, W. Wang, and R. Webb. Learning from simulated and unsupervised images through adversarial training. *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 2107–2116, July 2017. Honolulu, HI.

[49] K. Simonyan and A. Zisserman. Very deep convolutional networks for large-scale image recognition. *Proceedings of the International Conference on Learning Representations*, May 2015. San Diego, CA.

[50] H. Su, J. Deng, and L. Fei-Fei. Crowdsourcing annotations for visual object detection. *Proceedings of the Association for the Advancement of Artificial Intelligence Human Computation Workshop*, WS-12-08:40–46, July 2012. Toronto, Canada.

[51] R. Sui, B. E. Hartley, J. M. Gibson, C. Yang, J. A. Thomasson, and S. W. Searcy. High-biomass sorghum yield estimate with aerial imagery. *Journal of Applied Remote Sensing*, 5(1):053523, January 2011.

[52] V. Sundstedt. *Gazing at Games: An Introduction to Eye Tracking Control*, volume 5. Morgan & Claypool Publishers, San Rafael, CA, 2012.

[53] L. Świrski, A. Bulling, and N. Dodgson. Robust real-time pupil tracking in highly off-axis images. *Proceedings of the Symposium on Eye Tracking Research and Applications*, pages 173–176, March 2012. Santa Barbara, CA.

[54] A. A. Taha and A. Hanbury. Metrics for evaluating 3D medical image segmentation: Analysis, selection, and tool. *BMC Medical Imaging*, 15(1):29, August 2015.

[55] P. Teikari, M. Santos, C. Poon, and K. Hynynen. Deep learning convolutional networks for multiphoton microscopy vasculature segmentation. *arXiv:1606.02382*, June 2016.

[56] J. H. M. Thornley. Crop yield and planting density. *Annals of Botany*, 52(2):257–259, August 1983.

[57] I. Tokatlidis and S. D. Koutroubas. A review of maize hybrids' dependence on high plant populations and its implications for crop yield stability. *Field Crops Research*, 88(2):103–114, August 2004.

[58] J. Tompson, R. Goroshin, A. Jain, Y. LeCun, and C. Bregler. Efficient object localization using convolutional networks. *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 648–656, June 2015. Boston, MA.

[59] J. R. R. Uijlings, , K. E. A. Van De Sande, T. Gevers, and A. W. M. Smeulders. Selective search for object recognition. *International Journal of Computer Vision*, 104(2):154–171, September 2013.

[60] F. Xiong, X. Shi, and D. Yeung. Spatiotemporal modeling for crowd counting in videos. *Proceedings of the IEEE International Conference on Computer Vision*, pages 5151–5159, October 2017. Venice, Italy.

[61] C. Zhang, H. Li, X. Wang, and X. Yang. Cross-scene crowd counting via deep convolutional neural networks. *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 833–841, June 2015. Boston, MA.

[62] Y. Zhang, D. Zhou, S. Chen, S. Gao, and Y. Ma. Single-image crowd counting via multi-column convolutional neural network. *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 589–597, June 2016. Las Vegas, NV.

[63] S. K. Zhou, H. Greenspan, and D. Shen. *Deep Learning for Medical Image Analysis*. Academic Press, London, United Kingdom, 2017.

CVPR2019 Paper Translation

姓名： 张舵

学号： 2017303074

班号： 10011705

定位没有边界框的目标

Javier Ribera, David Güera, Yuhao Chen, Edward J. Delp
视频与图像处理实验室 (VIPER), 普渡大学

摘要

最近卷积神经网络研究进展（CNN）在图像中的目标定位方面取得了显著的效果。在这些网络中，训练过程通常需要提供边界框或者预期的目标的最大数量。在这篇论文中，我们解决了预估目标位置的任务并且没有使用标注的边界框，其中这些边界框通常需要手工绘制并且消耗时间去进行标注。我们提出了一个损失函数，可在任何全卷积网络（FCN）使用来预估对象位置。这个损失函数是两个无序点集之间的平均Hausdorff距离的改进。提出的方法没有边界框、候选区域或者滑动窗口的概念。我们使用三个数据集来评估我们的方法，这些数据集旨在定位人的头部、瞳孔中心和植物中心。我们的性能优于最先进的通用物体探测器和针对瞳孔跟踪进行微调的方法。

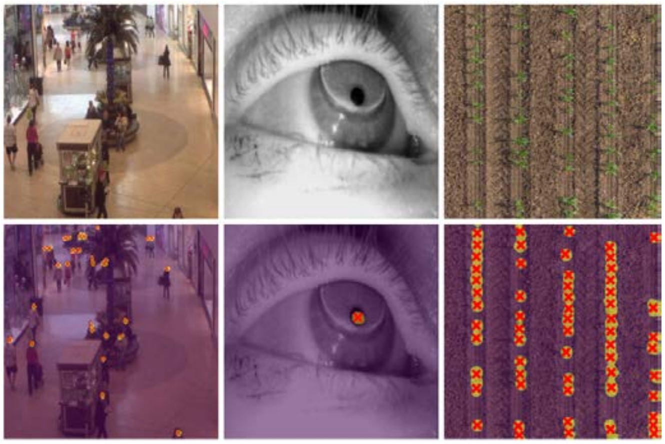

图 1. 人的头部、瞳孔、植物中心的目标定位。（下面一排）热度图和交叉估计

1. 介绍

定位物体在图像中的位置在计算机视觉中是一件非常重要的任务。在目标检测中一种普遍的方法是获取感兴趣对象周围的边界框。在本文中，我们对获取边界框不敢兴趣。相反，我们将目标检测任务定义为获取对应于每个对象位置的单个二维坐标。目标的位置可以是我们感兴趣的任何关键点，比如它的中心。图片 1 展示了一个在图像中定位目标的例子。与其它关键点检测问题不同，我们事先不知道在图像中关键点的数目。为了使这个方法尽可能的通用，我们不假定点之间存在任何物理约束，这与姿态估计等情况不同。这种对目标检测的定义更适合于对象非常小或者基本上重叠的应用（比如图片1中重叠的植物）。在这些情况下，数据集可能不会提供边界框或者它们不可能得到真实的边界框（groundtruth）。

边界框的标注冗长、耗时且昂贵 [37]。例如，在Amazon的Mechanical Turk使用专门为边界框开发的高效标注技术进行众包 [50]，标注ImageNet [43]中的每一个边界框需要42秒。在 [6]中，Bell等人介绍了一个用于材料分类和分割的数据集。通过在此数据集中收集单击位置的标签，而不是像素级的分割，将标注成本降低了一个数量级。

在本文中，我们提出了一种修正的平均hausdorff距离作为CNN的损失函数来预测目标的位置。我们的方法不要求在训练阶段使用边界框，并且在设计网络结构的时候也不需要知道目标的最大数量。简单起见，我们只用一类目标来描述我们的方法，尽管这方法可以简单地扩展到多类目标。我们的方法是目标不可知的，因此本文的讨论不包括任何关于目标特性的信息。我们的方法将输入图像映射到一组坐标，并用不同的对象对其进行验证。我们用三个数据集来验证我们的方法。第一个数据集包含了从购物中心监控摄像头获取的图像，我们在其中定位了人头。第二个数据集包含了人眼睛的图片，我们在其中定位了瞳孔的中心。

第三个数据集包含了由无人机航拍得到的谷物地的图像，我们定位了高度封闭（highly occluded）植物中心。

我们通过关键点检测实现目标定位的方法并不能取代边界框检测，尤其是对那些天生需要边界框的任务，比如自动裁剪。此外，这种方法的一个局限性是，边界框标签包含了某种意义上的尺度，而关键点没有。

我们工作的贡献是：

- 我们提出了一个用于目标定位的损失函数，我们称之为加权*hausdoeff*距离（WHD），它克服了像素级的损失函数，如L2和hausdorff距离，的局限性。。
- 我们发展了一种用于估计图像中物体的位置和数量的方法，并且这种方法不使用任何边界框或者候选局域。
- 我们把目标定位问题表述为点之间最小距离的问题，与预测中使用的模型无关。这允许使用任何全卷积的网络架构设计。
- 我们的性能优于目前最先进的通用目标检测器，并且在不需要任何领域特定知识、数据增强或迁移学习的情况下，在人群计数场景下获得了不错的结果。

2. 相关工作

通用目标检测器。 深度学习 [16, 27]的最新进展提高了定位任务（如目标或关键点检测）的准确性。对于通用的目标检测器，我们指的的是可以经过训练来检测任何一种或多种目标类型的方法，例如Faster-RCNN [15]，SSD [31]或者YOLO [40]。在Fast R-CNN中，候选区域或者建议区域是通过诸如选择性搜索 [59]之类的经典方法生成的。虽然网络的激活在不同的候选区域是共享的，但是系统不能端到端地进行训练。目标检测器中的区域建议网络（RPNs）例如Faster R-CNN [15, 41]允许端到端的模型训练。 Mask R-CNN [18]通过增添一个用于预测物体掩码的分支来扩展Faster R-CNN，但是它和已经存在的用于边界框识别的分支并行。 Mask r-cnn可以通过生成一个单独的类来表示关键点的存在来估计人体姿态关键点。 Mask R-CNN中使用的损失函数是逐点的，使得关键点检测对分割掩模的对齐高度敏感。 SDD提供固定大小的边界框和分数，指示框中是否存在目标。刚刚描述的方法要么在训练CNN时需要边界框要么要求知道图像中的最大目标数量。在 [19]中，观察到一些诸如Faster R-CNN或者SSD的通用目标检测器在检测小物体时表现很差。

计数和定位目标。 计算一个图片中目标的数目并不是一件简单的任务。在 [28]中，Lempitsky等人估计一个密度函数，它的积分相当于目标的数量。在 [47],Shao等人提出了两种定位目标物体的方法。一种方法是首先计数然后定位，另一种方法是首先定位然后计数。

定位和计数人群的数量在很多应用中都是必要的，例如监控系统中的人群监控，新业务的调查，以及应急事件管理 [28, 60]. 文献对此有多种研究，人群在时评中被检测和跟踪。这些检测方法通常使用包围的人边界框作为基础。在很多人互相重叠的情况下，获取一个人群中每一个人边界框是劳动密集且不准确的工作，比如运动会或者高峰时段的交通车站。很多现代的方法通过估计一个密度图然后对其进行积分得到总人数来避免对边界框的需要。在使用密度图的方法中，密度图的标签由人脑袋的标签构建。这通常是通过将高斯核集中在每个头部的位置来完成的。张等人 [62]用多栏CNN估计密度图像，这个CNN可以在不同的尺度上学习特征。在[44]中，Sam等人使用多个独立的CNN来预测不同人群密度下的密度图。另外一个CNN对人群场景的密度进行分类，并将输入图像转发给相应的CNN。黄等人 [20]建议将有关身体部位结构的信息合并到传统的密度图中，将人群计数重新定义为一个多任务问题。其它的工作比如张等人 [61]使用额外的信息比如被视为基本真理的映射图。

瞳孔跟踪和精确农业的方法通常是特定领域的。在瞳孔跟踪中，必须在实际光照条件下对瞳孔中心进行解析。广泛的应用中，从商业应用比如视频游戏 [52],驾驶 [48, 17]或者微手术 [14]都依赖于瞳孔追踪。在远程精准农业中，定位农田中的植物中心是至关重要的。农学家用这些植物特性比如植物间距来预测未来的农田产量 [56, 51, 57, 12, 8]，而植物学家则利用这些特性来培育新的植物品种 [3, 35]。在 [1]中，Aich等人通过首先将植物分割成若干区域然后计数每个区域的植物数量来计算小麦的数量。

豪斯多夫（Hausdorff）距离。 豪斯多夫距离可以用来衡量两个点集之间的距离 [5]。豪斯多夫距离的改进版 [10]被用到了很多任务之中，包括字符识别，[33]，人脸识别 [23]以及场景匹配 [23]。 Schutze等人[46]使用平均的豪斯多夫距离来衡量多目标优化的解决方案。在 [24]中，Elkhiyari等人比较CNN根据多个豪斯多夫距离的变体提取的特征，以完成人脸识别任务。在 [11]中，Fan等人使用Chamfer和Mover’s距离以及一个新的神经网络结构，通过估计一些固定点的位置来进行三维重建。豪斯多夫距离也是评价医学影像界分割边界质量的一个常用指标。

3. 平均豪斯多夫（Hausdorff）距离

我们的工作是建立在豪斯多夫距离基础上的，我们将在本节中简洁地回顾它。考虑两个非空的无序点集 X 和 Y 以及一个距离度量 $d(x,y)$,其中 $x \in X$ 并且 $y \in Y$。函数 $d(\cdot,\cdot)$可以是任何度量标准。在我们的例子中，我们使用欧几里得距离。点集 X 和 Y 可能拥有不同数量的点。让 $\Omega \subset \mathbb{R}^2$表示这个空间所有的点。一般来说，$X \subset \Omega$ 和 $Y \subset \Omega$之间的距离被定义为

$$d_{\mathrm{H}}(X,Y) = \max\left\{\sup_{x\in X}\inf_{y\in Y} d(x,y), \sup_{y\in Y}\inf_{x\in X} d(x,y)\right\}. \tag{1}$$

当考虑一个离散的有界的 Ω，比如一张图片中的所有像素点，最小上界和最大下界都是可实现的，并且分别成为最大值和最小值。这将豪斯多夫距离限制为

$$d(X,Y) \le d_{max} = \max_{x\in\Omega, y\in\Omega} d(x,y), \tag{2}$$

当使用欧式距离时它相当于一张图像的对角线。如果[5]所展示的，豪斯多夫距离是一个衡量标准。因此，对于$\forall X,Y,Z \subset \Omega$，我们有以下的性质：

$$d_H(X,Y) \ge 0 \tag{3a}$$

$$d_H(X,Y) = 0 \iff X = Y \tag{3b}$$

$$d_H(X,Y) = d_H(Y,X) \tag{3c}$$

$$d_H(X,Y) \le d_H(X,Z) + d_H(Z,Y) \tag{3d}$$

方程 (3b)由X和Y是有界的导出，因为在我们的任务中像素的坐标空间 Ω是离散的。当要设计一个函数去衡量X和Y是多么相似的时候，这些特性是非常有用的。

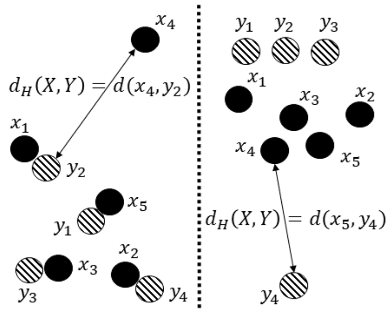

图 2. 点集$X = \{x_1,...,x_5\}$（实心点）和$Y = \{y_1,...,y_4\}$（虚线点）两种不同的布局的展示。尽管这两个点集之间有着明显的区别，但它们之间的豪斯多夫距离却因为最糟糕的离群点是一样的而相等。

豪斯多夫函数的一个缺点是它对离群点的高度敏感性 [46, 54]。图 2展示一个关于两个都有一个离群点的点集的例子。为了避免这种情况，平均豪斯多夫距离被广泛地使用：

$$d_{\mathrm{AH}}(X,Y) = \frac{1}{|X|}\sum_{x\in X}\min_{y\in Y} d(x,y) + \frac{1}{|Y|}\sum_{y\in Y}\min_{x\in X} d(x,y), \tag{4}$$

其中$|X|$和$|Y|$分别是X和Y中点的数量。

注意性质 (3a)， (3b)和 (3c)仍然是真的，但是(3d)不再成立。同时，平均豪斯多夫距离关于任何在X或者Y中的点是可微分的。

让Y包含正确的像素坐标，X是我们的估计。理想情况下，我们将在训练卷积神经网络（CNN）时使用$d_{\mathrm{AH}}(X,Y)$作为我们的损失函数。当把平均豪斯多夫距离作为损失函数时我们发现两个限制。首先，具有线性层的CNN隐式地决定了要估计$|X|$中点的数目，即最后一层的神经元个数。这是一个缺陷，因为真正点的数目由图像自己的内容决定。第二，全卷积网络（FCNs）比如U-Net [42]可以指示在输出层中存在具有更高激活度的对象中心，但它们不返回像素坐标。为了在学习过程中使用反向传播，损失函数对网络的输出必须是可微的。

4. 带权重的豪斯多夫（Hausdorff）距离

为了克服这两种限制，我们对平均豪斯多夫距离的改进如下所示：

$$d_{\mathrm{WH}}(p,Y) = \frac{1}{\mathcal{S}+\epsilon}\sum_{x\in\Omega} p_x \min_{y\in Y} d(x,y) + \frac{1}{|Y|}\sum_{y\in Y} \underset{x\in\Omega}{M_\alpha}\left[\, p_x d(x,y) + (1-p_x) d_{max}\,\right], \tag{5}$$

其中

$$\mathcal{S} = \sum_{x\in\Omega} p_x, \tag{6}$$

$$\underset{a\in A}{M_\alpha}\left[f(a)\right] = \left(\frac{1}{|A|}\sum_{a\in A} f^{\alpha}(a)\right)^{\frac{1}{\alpha}}, \tag{7}$$

是由平均值产生，ϵ被设定为10^{-6}。我们称$d_{\mathrm{WH}}(p,Y)$为加权豪斯多夫距离（WHD）。$p_x \in [0,1]$是网络在坐标点x处输出的一个单值。最后一层网络的激活值可以使用sigmoid非线性激活函数将其限制在0和1之间。注意p不需要标准化，即$\sum_{x\in\Omega} p_x = 1$是不必要的。注意广义上讲，平均$M_\alpha\left[\cdot\right]$对应于当$\alpha = -\infty$时的最小函数。我们证明将改进应用于 (4)可以得到 (5)：

1. 第一个项分母中的ϵ在$p_x \approx 0\ \forall x \in \Omega$时提供了数值的稳定性。

2. 当$p_x = \{0,1\}, \alpha = -\infty$，并且$\epsilon = 0$时，加权的豪斯多夫距离变成了平均的豪斯多夫距离。我们可以将其解释为网络可以指示出完全确定的物体坐标。当$d_{\mathrm{WH}}(p,Y) \geq 0$时，如果$x \in Y$，则全局最小值($d_{\mathrm{WH}}(p,Y) = 0$)对应于$p_x = 1$，否则对应于0。

3. 在第一项中，我们乘以p_x 来惩罚在图像中附近没有真正点y的高激活值。换句话说，这个损失函数惩罚了不应该出现在这地方的估计点。

4. 在第二项中，通过使用表达式 $f(\cdot) := p_x d(x,y) + (1-p_x) d_{max}$，我们迫使

   (a) 如果$p_{x_0} \approx 1$，则$f(\cdot) \approx d(x_0,y)$。这意味着点$x_0$对损失的贡献和其在AHD（Equation (4)）中的贡献相同。

   (b) 如果$p_{x_0} \approx 0$，$x_0 \neq y$，则$f(\cdot) \approx d_{max}$。然后，如果$\alpha = -\infty$，这个点x_0将不会对损失值产生贡献，因为这个“最小值”$M_{x\in\Omega}[\cdot]$会忽视x_0。如果存在一个其它点x_1和y更近且$p_{x_1} > 0$，x_1将被“选择”，而不是$M[\cdot]$。否则，$M_{x\in\Omega}[\cdot]$的值将会很高。这意味着真实目标点周围的低激活值将被惩罚。

   注意$f(\cdot)$并不是唯一实施这两个约束($f|_{p_x=1} = d(x,y)$ and $f|_{p_x=0} = d_{max}$)的表达式。我们选择一个线性的函数，原因是它们的简单性和数值稳定性。

WHD中的两项都是必要的。如果第一项被移除，则这个平凡解是$p_x = 1 \quad \forall x \in \Omega$。如果第二项被移除，则这个平凡解是$p_x = 0 \quad \forall x \in \Omega$。这两种情况对$\alpha$取任何值都适用，其证明在附录中。理想情况下，这个参数$\alpha \to -\infty$，因此$M_\alpha(\cdot) = ||\cdot||_{-\infty}$变成了取最小的操作符[26]。然而，这将使第二项与网络的输出持平。对于一个给定的y，如果有另一个点x_1有更高的激活值且更靠近y，点x_0中p_{x_0}中的变化即远离y的点将被$M_{-\infty}(\cdot)$忽视。在实践中，这是训练变得困难因为这个最小值对于输入来说并不是一个平滑的函数。因此，我们用广义的平均$M_\alpha(\cdot)$来近似最小值，且$\alpha < 0$。α越负，AHD和WHD越相似，以变的不那么光滑为代价。在我们的实验中，$\alpha = -1$。在第一项中没必要使用$M_\alpha(\cdot)$ 因为p_x并不在最小值中，因此这一项已经对p可微。

如果在将图像喂给网络之前需要对其进行改变，我们可以规范化WHD来应对这种失真。将原始图像记为$(S_o^{(1)}, S_o^{(2)})$同时将改变过后的图像记为$(S_r^{(1)}, S_r^{(2)})$。In Equation (5), we compute distances in the original pixel space by replacing $d(x,y)$ with $d(\mathbf{S}x, \mathbf{S}y)$, where $x,y \in \Omega$ and 在方程 (5)，我们用$d(\mathbf{S}x,\mathbf{S}y)$来替代$d(x,y)$来在原始的像素空间中计算距离，其中$x,y \in \Omega$，并且

$$\mathbf{S} = \begin{pmatrix} S_o^{(1)}/S_r^{(1)} & 0 \\ 0 & S_o^{(2)}/S_r^{(2)} \end{pmatrix}. \tag{8}$$

4.1. 相对于像素级损失函数的优势

一种幼稚的替代方法是使用单热编码的图作为标签，定义$x \in Y$时$l_x = 1$，否则$l_x = 0$，然后使用像素级的损失函数，比如均方误差（MSE）或者L^2标准，其中$L^2(l,p) = \sum_{\forall x\in\Omega} |p_x - l_x|^2 \propto \mathrm{MSE}(l,x)$。像素级损失函数的问题是除非$x = y$，否则它们不能说明两个点$x \in \Omega$和$y \in Y$的距离有多斤。换言之，对于绝大

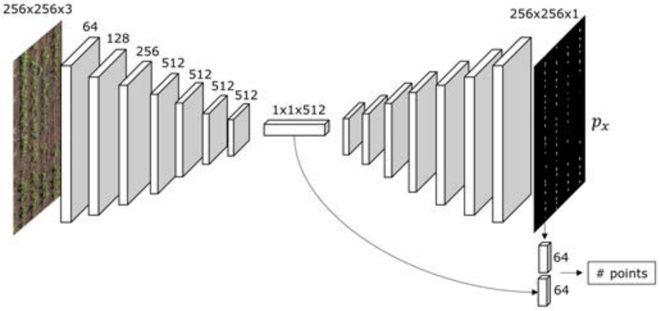

图 3. 用于目标定位的FCN架构，最小限度的改编自U-Net [42]架构。 W我们添加了一个小的全连接层，它将最深层的特征和估计出的概率密度图结合起来去回归得到点的数量。

多数像素来说，它是平坦的，这使得训练变得不可行。这个问题通过使用以$x \in Y$为中心高斯分布的均方误差在局部得到缓解[58]。对比之下，方程 (5)中的WHD将会随着x靠近y而减小，使得损失函数可以全局最小值之外提供训练所需的信息。

5. CNN架构以及位置估计

在这个部分，我们将描述我们使用的全卷积网络，以及我们我们怎么估计最终的位置。我们想强调这个网络结构的设计在我们的工作中并不是一个有意义的贡献。因此我们没有尝试去优化它。我们的主要贡献是使用加权的豪斯多夫距离作为我们的损失函数。我们采用了U-Net的网络结构[42]并对为了这个任务做了微小的修改。类似于U-Net的网络已经被证明，在训练时采用条件对抗生成网络的设置或 [22]者使用一个精心调试的损失函数的情况下，它有能力精准地将输入图片映射到输出图片上。图 3展示了U-Net的沙漏设计。为了简单起见，没有显示出编码器中每一层与解码器中之对称的层之间的残差连接。

这个FCN有两个区别很大的块。第一个块遵循了CNN典型的结构。它由两个3×3卷积（带1填充）重复应用组成，其中每一层紧跟了一个批量归一化和一个整流线性单元（ReLU）。在ReLU之后，我们使用了不长为2的2×2的最大池化操作来进行降采样。在下采样的步骤中，我么将特征的通道数加倍，开始为64通道，在最后5层使用512通道。

第二个块由以下元素重复应用组成：一个双线性上采样，与一个来自下采样的特征图拼接，以及两个3×3的卷积，其中每一个跟随一个批量归一化和一个RELU激活层。最后一层是一个将网络输出映射到一个单通道的的卷积层，p。

为了估计目标的图片中目标的数量，我们添加了一个将最深层的特征和估计的概率图结合起来的分支。这个分支将两个特征(一个$1 \times 1 \times 512$特征向量和256×256概率图)结合到一个隐藏层中，然后使用一个128维的特征向量输出一个值。然后我们使用看一个ReLU激活函数来保证输出时正的，同时将其舍入到最近的整数来获得我们最终对目标数量的估计，$\hat{C}$。

虽然我们使用了特定的网络结构，其它任何结构也可以被使用。唯一的要求是网络的输出图像必须和输入图像的尺寸相同。选择FCN源自于其输出作为WHD（方程 (5)）中的权重(p_x)的自然解释。在之前的工作中[24, 11]，平均豪斯多夫距离的变体已经被成功地应用在直接估计点集的非FCN网络中。但是，在这些案例中，被估计集合的大小已经被最后一层的尺寸给确定了。为了定位未知数量的物体，网络必须有能力来估计数量可变的物体位置。因此，我们可以设想，只要网络的输出使用方程(5)中的p，WHD也可以应用到非FCN网络当中。

我们用于训练网络的的损失函数是一个方程(5)和一个用于对目标数量进行回归的平滑的L_1损失函数的组合。最终的损失函数是

$$\mathcal{L}(p, Y) = d_{WH}(p, Y) + \mathcal{L}_{\text{reg}}(C - \hat{C}(p)), \tag{9}$$

其中Y是包含了图像中目标的真正坐标的集合，p是网络的输出，$C = |Y|$，以及$\hat{C}(p)$是对目标数量的估计。 $\mathcal{L}_{\text{reg}}(\cdot)$是回归项，这一项我们使用平滑的$L_1$或者Huber损失[21]，定义为

$$\mathcal{L}_{\text{reg}}(x) = \begin{cases} 0.5x^2, & \text{for} |x| < 1 \\ |x| - 0.5, & \text{for} |x| \geq 1 \end{cases} \tag{10}$$

当回归误差较大时，这种损失对离群点是鲁棒的，同时在原点是可微的。

网络输出一个显著图p，用$p_x \in [0, 1]$指示在像素点x有目标物体的置信度。图4在第二行展示了p。在评估过程中，我们的最终目标是获得$\hat{Y}$，例如，对所有的目标位置的估计。为了将p转化为$\hat{Y}$，我们对p进行了阈值来获得像素$T = \{x \in \Omega \mid p_x > \tau\}$。我们可以使用三种方法来决定使用哪一个$\tau$。

1. 对于所有的图像使用一个常数τ。

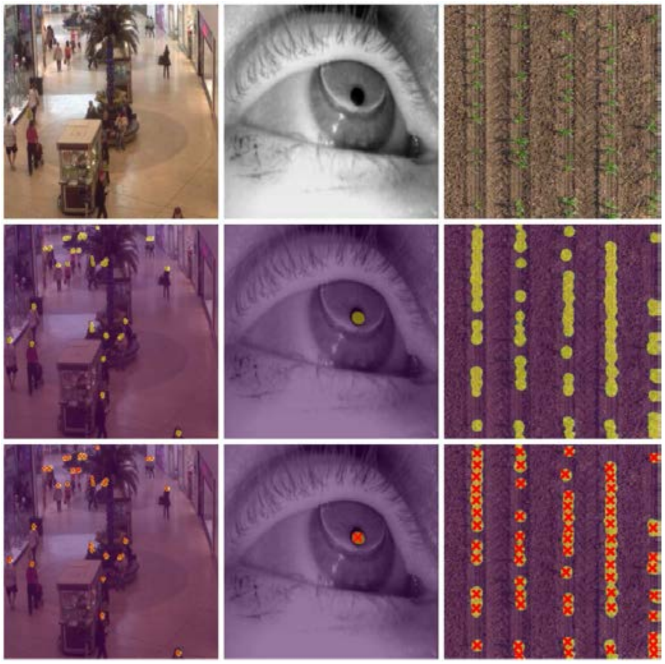

图 4. 第一行，输入图像第二行：网络的输出（文本中的p）覆盖在输入图像上。这个可以被认为是一个目标定位的显著图。第三行：估计目标的位置使用了红色十字标记。

2. 使用Otsu阈值[36]来为每一幅图片寻找一个合适的τ。

3. 使用Beta混合模型的阈值（BMM）。该方法使用[45]中描述的算法将两个Beta分布的混合拟合为p值，然后取平均值最高的分布的平均值作为τ。

图4第三行展示了一个对显著图p阈值化后得到的结果的例子。然后，我们将高斯混合模型拟合都T点。这是使用期望最大化（EM）[34]算法和估计的植物数量$\hat{C}$来完成的。

拟合的高斯平均数被认为最终的估计值$\hat{Y}$。图4的第三行展示标记了红色十字的最终估计物体。需要注意的是，即使FCN结果良好，即每个目标的位置都一个集群，如果$|\hat{C} - C| > 0.5$，EM也有可能不能产生正确的目标位置。在图4的第一列可以看到一个例子，一个人头被错误地估计为两个人头。

6. 实验结果

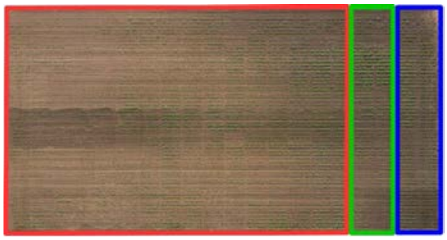

图 5. 一个拥有15208张图片的拼接图像。红色区域用于训练，绿色区域用于验证，蓝色区域用于测试。

我们用三个数据集来评价我们地方法。第一个数据集包含来自一个商场监控摄像头拍摄的2000张照片。它包含已经标注的人群中人头的位置。这个数据集是可以在http://personal.ie.cuhk.edu.hk/~ccloy/downloads_mall_dataset.html [32]公开访问的。 80%, 10% and 10% 的随机打乱的图片分别被作为训练集验证集和测试集。

第二个数据集在[13]中以罗马字母出现，并且 在http://www.ti.uni-tuebingen.de/Pupil-detection.1827.0.html公开提供。它包含2135张单只眼睛的图片，目标是检测瞳孔的中央。它也是随意按照80/10/10 %的比例被划分为训练集，验证集和测试集。

第三个数据集包含一个农田的航拍图像，这些图像是由一个无人机在40m的海拔拍摄的。这些图像被拼接在一起，生成了一个$6,000 \times 12,000$的0.75cm/每像素分辨率的正射图像，如图5所示。图像中所有植物的中心都是真实的，共计15208个独一无二的植物中心。这个马赛克图像被分割，左边百分之80用于训练，中间的百分之10用于验证，右边百分之10用于测试。在每个区域内，随机生成图像作物。这些随机作物的高度和宽度在100到600像素之间均匀分布。我们在训练区域随机抽取50000张图片，验证区域和测试区域各随机抽取$5,000$张图片。注意这些图片可能会高度重叠。我们使第三个数据集可 以 在https://engineering.purdue.edu/~sorghum/dataset-plant-centers-2016被 公开访问。我们相信这个数据集对社区是有用的，因为由于植物的高度封闭性，这是一个极大的挑战。

所有的图片都被调整为256×256，因为这是我们网络结构允许的最小大小。真实目标的位置也被相应的缩放。对于数据增强，我们使用了随机水平翻转。对于植物的数据集，我们也使用了竖直的翻转。我们

设置方程 (7)中的$\alpha = -1$。我们也实验了$\alpha = -2$，但没有明显的提升，但是我们没有尝试去寻找一个最佳值。我们为每个数据集重新训练网络，即我们不使用预先训练的权重。对于商场和植物的数据集，我们设置批大小（batch size），使用Adam优化器 [25, 39]以及将学习率设置为10^{-4}，动量设置为0.9。对于眼瞳的数据集，我们通过移除5各个中间层来降低网络的大小，使用批大小（batch size）为64，随机梯度下降法的学习率为10^{-3}以及动量设置为0.9。在每一轮训练的结束，我们在验证集上衡量方程(4)中的平均豪斯多夫距离（AHD），同时选择在验证集上AHD最小的那一轮。

关于衡量标准，我们报告了准确率，召回率，F得分，AHD，平均绝对误差(MAE)，根均方误差（RMSE）以及平均绝对百分比误差（MAPE）。

$$\mathrm{MAE} = \frac{1}{N}\sum_{i=1}^{N}|e_i|, \quad \mathrm{RMSE} = \sqrt{\frac{1}{N}\sum_{i=1}^{N}|e_i|^2} \tag{11}$$

$$\mathrm{MAPE} = 100\frac{1}{N}\sum_{\substack{i=1\\C_i\neq 0}}^{N}\frac{|e_i|}{C_i} \tag{12}$$

其中$e_i = \hat{C}_i - C_i$，以及N是图片的数量，C_i是第i张图片中正确的目标个数，$\hat{C}_i$是我们估计的目标个数。

如果估计位置与真值点的距离不超过r，则计为真阳性。如果估计位置在距离r处没有任何真值点，则计为假阳性。如果一个真值点再距离r内没有估计点，则记为假阴性。准确率是我们估计的点中，真阳性的比例。召回率是真实位置中，我们检测出来的比率。F分数是准确率和召回率的调和平均数。注意，即使我们估计每个真值点附近有多个目标位置，也可以达到100%的精度和召回率。这将不是一个理想的定位。将这个考虑在内，我们同时报告了可以指示出目标数量是否正确的度量标准（MAE，RMSE和MAPE）。AHD可以被解释为以像素为单位的平均误差。

图 8展示了F分数作为r的函数。注意，r是只是一个的评价参数。在训练和测试时并不需要它。 MAE，RMSE以及MAPE在表1中展示了出来。注意，除了在眼瞳的数据集中去掉中间层，我们对所有任务都使用了相同的结构。并且，在瞳孔检测中，我们知道图片中只有一个目标。因此，回归是不必要的，同时我们可以移除方程 (9)中的回归项以及固定$\hat{C}_i = C_i = 1\ \forall i$。

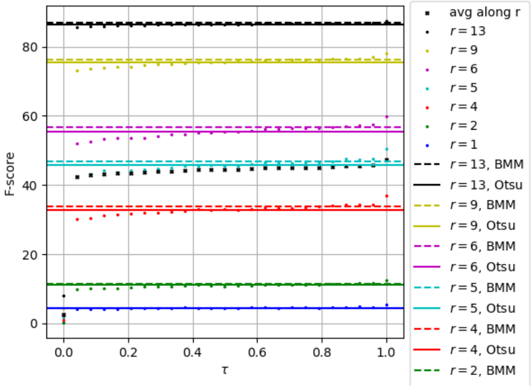

图 6. τ阈值对F分数的影响。

一种简单的目标定位方法是使用通用目标检测器，比如Faster R-CNN[41]。我们可以通过构造以每个标记点为中心的固定大小的边界框来训练这些检测器。然后，每个边界框的中心可以被认为是估计的位置。我们使用20×20大小的包围盒（近似平均的头部和瞳孔大小）和 16×16和32×32的锚大小。请注意，这些参数可能是次优的，即使它们是为匹配对象类型而选择的。我们用于SoftMax分数的阈值为0.5，用于联合交叉口的阈值为0.4，因为他们将验证集上的AHD最小化。我们使用VGG-16结构 [49]并用学习率为10^{-3}，动量为0.9的随机梯度下降法对其进行训练。对于瞳孔数据集，我们总选择得分最高的边界框。我们实验观察到，Faste R-CNN难以检测到非常小的且相互非常近的目标。表格 2-4展示了Faster R-CNN在商场，瞳孔以及植物数据集上的效果。注意商场和植物数据集有很多很小且相互重合的目标，这是对Faster R-CNN最大的挑战。这个行为和[19]中观察的一致，其中，所有的通用目标检测器表现得很差，Faster R-CNN在最好的情况下只有5%的平均正确率。

我们还使用均值漂移[9]代替高斯混合（GM）来检测局部极大值。但是，均值漂移容易检测出多个局部最大值，而GM对离群点的鲁棒性更强。在我们的实验中，我们观察到它的准确率和召回率要比使用GM差得多。更重要得是，使用均值漂移使验证慢了一个数量级。在使用scikit learn实现时，mean-shift算法在其中一幅图像上运行的平均时间是12秒，而使用期望最大化拟合gm大约需要0.5秒。

我们还研究了参数τ的影响，使用第 5节中的展示的三种方法。有人可能会认为，这个参数可能是某些

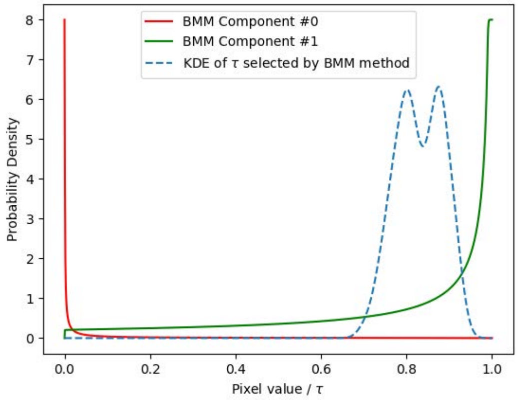

图 7. Beta混合模型拟合p_x值，阈值τ采用BMM方法。

表 1. 使用$r = 5$时，我们方法目标定位的结果。度量标准在方程 (4)， (11)-(12)中进行了定义。在瞳孔数据集中，因为总是只有一个瞳孔($\hat{C} = C = 1$)，故没有展示回归的度量标准。图8展示了其它值得F分数。

衡量标准	商场	瞳孔	植物	平均值
	数据集	数据集	数据集	
准确率	95.2%	99.5%	88.1%	94.4%
召回率	96.2%	99.5%	89.2%	95.0%
F分数	95.7%	99.5%	88.6%	94.6%
AHD	4.5 px	2.5 px	7.1 px	4.7 px
MAE	1.4	-	1.9	1.7
RMSE	1.8	-	2.7	2.3
MAPE	4.4%	-	4.2%	4.3 %

指标之间的一种折衷，并且应该进行交叉验证。在实际应用中，我们发现τ不平衡准确度和召回率，因此精确召回曲线没有意义。相反，我们在图 8画出了F值作为r的函数。除此之外，交叉验证tan意味着找到所有图像的“最佳”值。图 reffig:tau展示出使用适应性的阈值化方法效果更好（Otus或者BMM）。注意BMM阈值化方法（虚线）总是比Otus（实线）和大多数固定的τ效果好。为了证明BMM方法的适当性，请注意，在图 4中，估计图中的大多数值都非常高或非常低。这使得Beta分布比正态分布（如otsu方法中使用的）更适合对p_x建模。图 7显示了用BMM方法自适应选择的τ值的拟合BMM和核密度估计。

最后，因为我们的方法同时定位和计数目标，所以它可以被用为一个计数技术。我们还使用[62]中提供的上海理工大学B部分数据集对我们的人群计数技术进

表 2. 商场数据集得头部定位结果，使用$r = 5$。

衡量标准	Faster-RCNN	我们的
准确率	81.1%	**95.2 %**
召回率	76.7%	**96.2 %**
F分数	78.8 %	**95.7 %**
AHD	7.6 px	**4.5 px**
MAE	4.7	**1.4**
RMSE	5.6	**1.8**
MAPE	14.8%	**4.4 %**

表 3. 瞳孔检测结果，使用$r = 5$。因为只有一个估计和一个真实目标，故准确率和召回率相同

方法	准确率	召回率	AHD
Swirski [53]	77 %	77 %	-
ExCuSe [13]	77 %	77 %	-
Faster-RCNN	99.5 %	99.5 %	2.7 px
Ours	**99.5 %**	**99.5 %**	**2.5 px**

表 4. 使用植物数据集进行植物定位的结果，使用$r = 5$。

衡量标准	Faster-RCNN	我们的
准确率	86.6 %	**88.1 %**
召回率	78.3 %	**89.2 %**
F分数	82.2 %	**88.6 %**
AHD	9.0 px	**7.1 px**
MAE	9.4	**1.9**
RMSE	13.4	**2.7**
MAPE	17.7 %	**4.2 %**

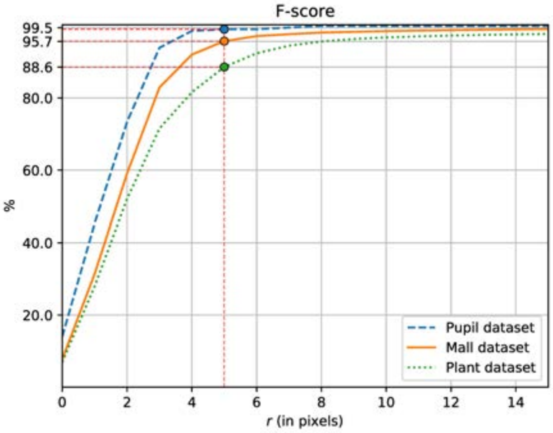

图 8. F分数作为r的函数，正确与估计的目标位置之间的最大距离，以判断其正确与否。一个较高的r使正确地地位更加简单。

行了评估，并获得了19.9的MAE。尽管我们没有超过专门针对人群计数进行微调的最新方法[29]，但我们可以使用我们的通用方法获得可比的结果。我们期望将来的改进，比如架构的改变或者使用迁移学习来进一步提高性能。

使用PyTorch实现的使用加权豪斯多夫距离作为损失函数的代码和训练的模型可以在`https://github.com/javiribera/locating-objects-without-bboxes`上获得。

7. 结论

我们提出了一个损失函数，用于在不需要包围盒边界框的图像中的目标定位。这个损失函数时平均豪斯多夫距离（AHD）的改进，用来衡量来两个无序点集之间的距离。为了使AHD相对于网络输出可微，我们在估计目标位置时考虑了网络的确定性。网络的输出是一个用来定位的显著性图和一个估计目标的数量的数字。我们的方法不受目标最大数量的限制，不需要边界框，不需要候选区域或者滑动窗。这个方法可以应用在边界框难以获得或者目标尺寸太小以至于边界框的标定不实际的任务。我们在三个数据集上衡量了我们的数据集，并且效果优于通用目标检测器和特定任务技术。未来的工作将包括在开发出在单一网络中的多目标定位器，以及使用更多现代的CNN架构。

致谢: 这项工作由高级研究项目局能源部（ARPA-E）资助。美国能源部，授权编号DE-AR0000593。本文作者的观点和意见并不一定反映美国政府或其任何机构的观点和意见。因为本文中使用的正射图像，我们感谢Ayman Habib教授。联系信息：Edward J. Delp, `ace@ecn.purdue.edu`

参考文献

[1] S. Aich, I. Ahmed, I. Obsyannikov, I. Stavness, A. Josuttes, K. Strueby, H. Duddu, C. Pozniak, and S. Shirtliffe. Deepwheat: Estimating phenotypic traits from crop images with deep learning. *Proceedings of the IEEE Winter Conference on Applications of Computer Vision*, March 2018. Stateline, NV.

[2] M. Andriluka, S. Roth, and B. Schiele. People-tracking-by-detection and people-detection-by-tracking. *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, June 2008. Anchorage, AK.

[3] J. L. Araus and J. E. Cairns. Field high-throughput phenotyping: the new crop breeding frontier. *Trends in Plant Science*, 19(1):52–61, January 2014.

[4] E. M. Arkin, L. P. Chew, D. P. Huttenlocher, K. Kedem, and J. S. Mitchell. An efficiently computable metric for comparing polygonal shapes. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 13(3), March 1991.

[5] H. Attouch, R. Lucchetti, and R. J. B. Wets. The topology of the ρ-Hausdorff distance. *Annali di Matematica Pura ed Applicata*, 160(1):303–320, December 1991.

[6] S. Bell, P. Upchurch, N. Snavely, and K. Bala. Material recognition in the wild with the materials in context database (supplemental material). *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, June 2015. Boston, MA.

[7] M. D. Breitenstein, F. Reichlin, B. Leibe, E. Koller-Meier, and L. V. Gool. Online multiperson tracking-by-detection from a single, uncalibrated camera. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 33(9):1820–1833, 2011.

[8] B. S. Chauhan and D. E. Johnson. Row spacing and weed control timing affect yield of aerobic rice. *Field Crops Research*, 121(2):226–231, March 2001.

[9] D. Comaniciu and P. Meer. Mean shift: A robust approach toward feature space analysis. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 24(5):603–619, 2002.

[10] M.-P. Dubuisson and A. K. Jain. A modified Hausdorff distance for object matching. *Pattern Recognition*, pages 566–568, October 1994.

[11] H. Fan, H. Su, and L. Guibas. A point set generation network for 3D object reconstruction from a single image. *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 2463–2471, July 2017. Honolulu, HI.

[12] D. E. Farnham. Row spacing, plant density, and hybrid effects on corn grain yield and moisture. *Agronomy Journal*, 93:1049–1053, September 2001.

[13] W. Fuhl, T. Kübler, K. Sippel, W. Rosenstiel, and E. Kasneci. ExCuSe: Robust pupil detection in real-world scenarios. *Proceedings of the International Conference on Computer Analysis of Images and Patterns*, pages 39–51, September 2015. Valletta, Malta.

[14] W. Fuhl, T. Santini, C. Reichert, D. Claus, A. Herkommer, H. Bahmani, K. Rifai, S. Wahl, and E. Kasneci. Non-intrusive practitioner pupil detection for unmodified microscope oculars. *Computers in Biology and Medicine*, 79:36–44, December 2016.

[15] R. Girshick. Fast R-CNN. *Proceedings of the IEEE International Conference on Computer Vision*, pages 1440–1448, December 2015.

[16] I. Goodfellow, Y. Bengio, and A. Courville. *Deep Learning*. MIT Press, November 2016.

[17] J. Gu, X. Yang, S. De Mello, and J. Kautz. Dynamic facial analysis: From bayesian filtering to recurrent neural network. *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 1548–1557, July 2017. Honolulu, HI.

[18] K. He, G. Gkioxari, P. Dollár, and R. Girshick. Mask R-CNN. *arXiv:1703.06870*, April 2017.

[19] J. Huang, V. Rathod, C. Sun, M. Zhu, A. Korattikara, A. Fathi, I. Fischer, Z. Wojna, Y. Song, S. Guadarrama, and K. Murphy. Speed/accuracy trade-offs for modern convolutional object detectors. *Proceedings of*

the IEEE Conference on Computer Vision and Pattern Recognition, July 2017. Honolulu, HI.

[20] S. Huang, X. Li, Z. Zhang, F. Wu, S. Gao, R. Ji, and J. Han. Body structure aware deep crowd counting. *IEEE Transactions on Image Processing*, 27(3):1049–1059, March 2018.

[21] P. J. Huber. Robust estimation of a location parameter. *The Annals of Mathematical Statistics*, pages 73–101, 1964.

[22] P. Isola, J.-Y. Zhu, T. Zhou, and A. A. Efros. Image-to-image translation with conditional adversarial networks. *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, July 2017. Honolulu, HI.

[23] K. L. K. Lin and W. Siu. Spatially eigen-weighted Hausdorff distances for human face recognition. *Pattern Recognition*, 36(8):1827–1834, August 2003.

[24] H. E. Khiyari and H. Wechsler. Age invariant face recognition using convolutional neural networks and set distances. *Journal of Information Security*, 8(3):174–185, July 2017.

[25] D. P. Kingma and J. Ba. Adam: A method for stochastic optimization. *Proceedings of the International Conference for Learning Representations*, abs/1412.6980, April 2015. San Diego, CA.

[26] C. S. Kubrusly. Banach spaces L^p. In *Essentials of Measure Theory*, page 83. Springer, Cham, 2005.

[27] Y. LeCun, Y. Bengio, and G. Hinton. Deep learning. *Nature*, 521:436–444, May 2015.

[28] V. Lempitsky and A. Zisserman. Learning to count objects in images. *Proceedings of the Advances in Neural Information Processing Systems*, pages 1324–1332, December 2010. Vancouver, Canada.

[29] Y. Li, X. Zhang, and D. Chen. CSRNet: Dilated convolutional neural networks for understanding the highly congested scenes. *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 1091–1100, June 2018.

[30] S. Liao, Y. Gao, A. Oto, and D. Shen. Representation learning: A unified deep learning framework for automatic prostate mr segmentation. *Proceedings of the Medical Image Computing and Computer-Assisted Intervention*, pages 254–261, September 2013. Nagoya, Japan.

[31] W. Liu, D. Anguelov, D. Erhan, C. Szegedy, S. Reed, C. Fu, and A. C. Berg. SSD: Single shot multibox detector. *Proceedings of the European Conference on Computer Vision*, pages 21–37, October 2016. Amsterdam, The Netherlands.

[32] C. C. Loy, K. Chen, S. Gong, and T. Xiang. Crowd counting and profiling: Methodology and evaluation. In *Modeling, Simulation and Visual Analysis of Crowds*, pages 347–382. Springer, October 2013.

[33] Y. Lu, C. L. Tan, W. Huang, and L. Fan. An approach to word image matching based on weighted Hausdorff distance. *Proceedings of International Conference on Document Analysis and Recognition*, pages 921–925, September 2001.

[34] T. K. Moon. The expectation-maximization algorithm. *IEEE Signal Processing Magazine*, 13(6):47–60, November 1996.

[35] E. H. Neilson, A. M. Edwards, C. K. Blomstedt, B. Berger, B. L. Møller, and R. M. Gleadow. Utilization of a high-throughput shoot imaging system to examine the dynamic phenotypic responses of a C_4 cereal crop plant to nitrogen and water deficiency over time. *Journal of Experimental Botany*, 66(7):1817–1832, 2015.

[36] N. Otsu. A threshold selection method from gray-level histograms. *IEEE Transactions on Systems, Man, and Cybernetics*, 9(1):62–66, January 1979.

[37] D. P. Papadopoulos, J. R. R. Uijlings, F. Keller, and V. Ferrari. We don't need no bounding-boxes: Training object class detectors using only human verification. *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 854–863, June 2016. Las Vegas, NV.

[38] F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R.

Weiss, V. Dubourg, J. Vanderplas, A. Passos, D. Cournapeau, M. Brucher, M. Perrot, and E. Duchesnay. Scikit-learn: Machine learning in Python. *Journal of Machine Learning Research*, 12:2825–2830, 2011.

[39] S. J. Reddi, S. Kale, and S. Kumar. On the convergence of adam and beyond. *Proceedings of the International Conference on Learning Representations*, April 2018. Vancouver, Canada.

[40] J. Redmon, S. Divvala, R. Girshick, and A. Farhadi. You only look once: Unified, real-time object detection. *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 779–788, June 2016. Las Vegas, NV.

[41] S. Ren, K. He, R. Girshick, and J. Sun. Faster R-CNN: Towards real-time object detection with region proposal networks. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 36(6):1137–1149, June 2017.

[42] O. Ronneberger, P. Fischer, and T. Brox. U-Net: Convolutional networks for biomedical image segmentation. *Proceedings of the International Conference on Medical Image Computing and Computer-Assisted Intervention*, pages 234–241, October 2015. Munich, Germany.

[43] O. Russakovsky, J. Deng, H. Su, J. Krause, S. Satheesh, S. Ma, Z. Huang, A. Karpathy, A. Khosla, M. Bernstein, A. C. Berg, and L. Fei-Fei. Imagenet large scale visual recognition challenge. *International Journal of Computer Vision*, 11(3):211–252, December 2015.

[44] D. B. Sam, S. Surya, and R. V. Babu. Switching convolutional neural network for crowd counting. *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 4031–4039, July 2017.

[45] C. Schröder. A hybrid parameter estimation algorithm for beta mixtures and applications to methylation state classification. *Algorithms for Molecular Biology*, 12(21):62–66, August 2017.

[46] O. Schutze, X. Esquivel, A. Lara, and C. A. C. Coello. Using the averaged Hausdorff distance as a performance measure in evolutionary multiobjective optimization. *IEEE Transactions on Evolutionary Computation*, 16(4):504–522, August 2012.

[47] J. Shao, D. Wang, X. Xue, and Z. Zhang. Learning to point and count. *arXiv:1512.02326*, December 2015.

[48] A. Shrivastava, T. Pfister, O. Tuzel, J. Susskind, W. Wang, and R. Webb. Learning from simulated and unsupervised images through adversarial training. *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 2107–2116, July 2017. Honolulu, HI.

[49] K. Simonyan and A. Zisserman. Very deep convolutional networks for large-scale image recognition. *Proceedings of the International Conference on Learning Representations*, May 2015. San Diego, CA.

[50] H. Su, J. Deng, and L. Fei-Fei. Crowdsourcing annotations for visual object detection. *Proceedings of the Association for the Advancement of Artificial Intelligence Human Computation Workshop*, WS-12-08:40–46, July 2012. Toronto, Canada.

[51] R. Sui, B. E. Hartley, J. M. Gibson, C. Yang, J. A. Thomasson, and S. W. Searcy. High-biomass sorghum yield estimate with aerial imagery. *Journal of Applied Remote Sensing*, 5(1):053523, January 2011.

[52] V. Sundstedt. *Gazing at Games: An Introduction to Eye Tracking Control*, volume 5. Morgan & Claypool Publishers, San Rafael, CA, 2012.

[53] L. Świrski, A. Bulling, and N. Dodgson. Robust real-time pupil tracking in highly off-axis images. *Proceedings of the Symposium on Eye Tracking Research and Applications*, pages 173–176, March 2012. Santa Barbara, CA.

[54] A. A. Taha and A. Hanbury. Metrics for evaluating 3D medical image segmentation: Analysis, selection, and tool. *BMC Medical Imaging*, 15(1):29, August 2015.

[55] P. Teikari, M. Santos, C. Poon, and K. Hynynen. Deep learning convolutional networks for multiphoton microscopy vasculature segmentation. *arXiv:1606.02382*, June 2016.

[56] J. H. M. Thornley. Crop yield and planting density. *Annals of Botany*, 52(2):257–259, August 1983.

[57] I. Tokatlidis and S. D. Koutroubas. A review of maize hybrids' dependence on high plant populations and its implications for crop yield stability. *Field Crops Research*, 88(2):103–114, August 2004.

[58] J. Tompson, R. Goroshin, A. Jain, Y. LeCun, and C. Bregler. Efficient object localization using convolutional networks. *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 648–656, June 2015. Boston, MA.

[59] J. R. R. Uijlings, , K. E. A. Van De Sande, T. Gevers, and A. W. M. Smeulders. Selective search for object recognition. *International Journal of Computer Vision*, 104(2):154–171, September 2013.

[60] F. Xiong, X. Shi, and D. Yeung. Spatiotemporal modeling for crowd counting in videos. *Proceedings of the IEEE International Conference on Computer Vision*, pages 5151–5159, October 2017. Venice, Italy.

[61] C. Zhang, H. Li, X. Wang, and X. Yang. Cross-scene crowd counting via deep convolutional neural networks. *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 833–841, June 2015. Boston, MA.

[62] Y. Zhang, D. Zhou, S. Chen, S. Gao, and Y. Ma. Single-image crowd counting via multi-column convolutional neural network. *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 589–597, June 2016. Las Vegas, NV.

[63] S. K. Zhou, H. Greenspan, and D. Shen. *Deep Learning for Medical Image Analysis*. Academic Press, London, United Kingdom, 2017.