ACCURATE SMALL OBJECT DETECTION VIA DENSITY MAP AIDED SALIENCY ESTIMATION

X. Q. Zhou, Y. X. Zou, Y. Wang*

ADSPLAB/ELIP, School of ECE, Peking University, Shenzhen 518055, China

ABSTRACT

Small object detection (SOD) in crowded scenes is a challenging task since objects are densely distributed and partially overlapped. In this paper, we propose a novel SOD method by fully exploring the information provided by the image and its estimated density map. Our proposed SOD method consists of two main stages. Initial object locations are firstly computed based on object spatial distribution information obtained from the estimated density maps. Inspired by the human visual attention mechanism, a saliency map which offers object boundaries is then employed to accurately estimate the bounding boxes with the support of the estimated initial object locations. Experimental results on three public small object datasets and a self-built snipe dataset demonstrate the effectiveness of our proposed SOD method, especially under small training set condition. It is encouraged to see that our SOD method only requires the dotted annotation training datasets and is able to estimate the bounding boxes fitting the shape of the objects accurately.

Index Terms**—**small object detection, density map, saliency map estimation, superpixel, dotted annotation training set

1. INTRODUCTION

Small object detection (SOD) with large numbers of objects is an object detection task for particular crowded macro-scene (such as flocks of birds) and microcosm (such as cells under a microscope). Object detection has two specific tasks: to estimate the object locations as well as the bounding boxes for further application such as object recognition. Therefore, the accuracy of the bounding box estimation (how accurate the bounding box fits the object in its overall shape) is equally important as the precision of object location in object detection tasks [1].

Typically, most individual-centric object detection methods use a sliding window or a region proposal method to generate proposals and train a classifier using the supervised algorithm [2-6]. For these object detection methods, positive and negative samples with bounding box annotations are needed. However, for SOD in crowded scenes, as the difficulty of manual annotation increases, usually only dotted annotations of objects are available for the positive samples in training sets and no negative sample is provided. Besides, the limited size of each object in crowded scenes leads to the insufficiency of visual information (e.g. textural or edge features, etc.), such as those in Fig.1. Moreover, valid features are partially hidden due to the frequent occlusion and overlap between objects. All the factors discussed above are challenges for using the traditional individual-centric detection methods [2-6, 14] to solve the SOD problems in crowded scenes.

Recently, object counting methods based on density map estimation [7,15,16] achieve promising performance in crowded scenes. Their research shows that the density map essentially offers good object distribution information which can be used for SOD in crowded scenes as well. Local maximum (LM) and integer programming (IP), as two different methods, were respectively employed to estimate the spatial locations of objects from a density map [8]. However, careful analysis shows that, for the LM method, the moving step of each ROI is hard to determine since the detector may fire on the same object twice by small steps while large steps may cause detection omission. Research outcomes show that outstanding detection performance is achieved by IP [8]. Unfortunately, as a typical NP-hard problem, IP is generally computational expensive. Moreover, since LM and IP methods compute the object locations based on density maps, the object boundary information is not available. As a result, such density estimation based SOD methods are not able to provide accurate shape fitting bounding boxes.

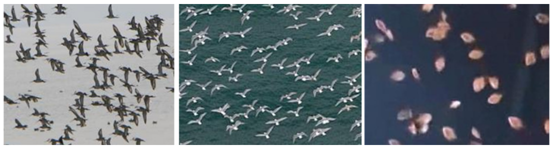


Fig. 1. Examples of small object detection

In this study, to supplement the object boundary details in density map, we propose an accurate SOD method for crowded scenes with dotted annotation training sets by exploring the information from both the image and its estimated density map. Specifically, an effective density map reconstruction method is presented. Benefitting from our previous research on visual object counting problem [15, 16], we observe that training images and their corresponding density maps share similar local geometry in patch level. We investigate this similarity to reconstruct the density maps of the testing images and pre-localize the objects by computing the local maximums over the density map. After that, we transfer to explore in the image domain. As superpixel based detection has been proved to be more flexible than proposal based detection [17] in terms of overlapped objects, we design a saliency map estimation method in superpixel level for crowded scenes. With the initial object localization results aided, object bounding boxes are further computed based on estimated saliency maps. As the superpixel based saliency map respect the boundaries of objects, our predicted bounding boxes fit the shape of object accurately.

In the rest of this paper, Sec. 2 presents the derivation and details of our proposed SOD method. Sec.3 shows the experimental results and performance analysis. Finally, Sec.4 gives the conclusion.

2. METHOD FORMULATION

In this section, we specifically present our small object detection (SOD) method, which is formulated in four parts as shown in Fig.2.

978-1-5090-2175-8/17/$31.00 ©2017 IEEE

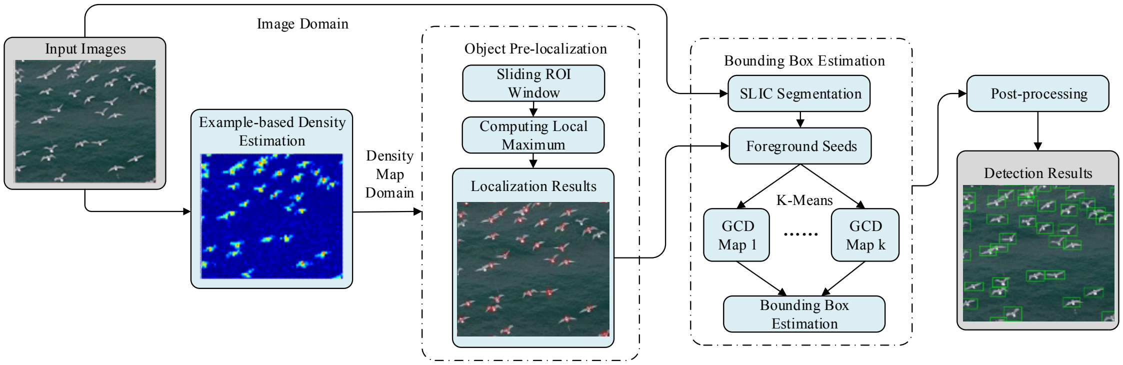


Fig. 2.The pipeline of our proposed small object detection method.

The example-based density map is estimated from the input image using exampled-based density map estimation (E-DME) method [16]. Then, object pre-localization is realized by computing local maximum in each sliding window over the estimated density map. On the other hand, bounding box estimation is realized in the image domain where the input image is segmented into N superpixels by SLIC [12]. The foreground seeds (superpixels) are selected by the pre-localization results and they are divided into K clusters by CIE-LAB features. K global color distinction (GCD) maps, regarded as sub-maps of saliency, are constructed based on K clusters. Hence, the superpixels with their saliency value approximated to the seeds are selected and the bounding boxes are estimated by the boundaries of the superpixels. Finally, Post-processing is conducted to optimize the detection result.

2.1. Example-based density map estimation

Example-based density map estimation (E-DME) method is our previous work for object counting [16]. In this paper, we adopt this method to estimate the density map, which takes advantage of the underlying local geometrical similarity between image patches and their density map patches. Thus, the estimated density map can better embody the spatial distribution information of objects and it's more beneficial to further object localization. To make the presentation completeness, the key principle of E-DME is presented. To estimate the density map, a set of N training images $\{\boldsymbol{I}_1, \boldsymbol{I}_2, \dots, \boldsymbol{I}_N\}$ is required. For each image $\boldsymbol{I}_i$ ($1 \le i \le N$), the ground-truth object locations are annotated manually with a set of 2D dots $\boldsymbol{P}_i$ at the center of each object. With the ground-truth locations, the ground truth density map for each annotated pixel location of a training image $p \in \boldsymbol{I}_i$ is defined in [7] as

$$F_i^o(p) = \sum\nolimits_{P \in \boldsymbol{P}_i} \mathcal{N}(p; P, \delta^2), \forall p \in \boldsymbol{I}_i \tag{1}$$

where $\mathcal{N}(p; P, \delta^2)$ is a 2D Gaussian kernel centered at P and δ is a smoothness parameter.

For E-DME method, image patch form is desired in the density estimation stage. Consequently, a set of image patches $\boldsymbol{Y} = \{\boldsymbol{y}_1, \boldsymbol{y}_2, \dots, \boldsymbol{y}_M\}$ ($\boldsymbol{y}_i \in \mathbb{R}^{n\times 1}$) is extracted from the training images $\boldsymbol{I}_i$, $i \in 1,2,\dots,N$, and the density map set $\boldsymbol{Y}^d = \{\boldsymbol{y}_1^d, \boldsymbol{y}_2^d, \dots, \boldsymbol{y}_M^d\}$ of corresponding patches is extracted from $\boldsymbol{I}_i^d$, $i \in 1,2,\dots,N$.

Based on local linear embedding (LLE), local geometry on manifolds characterize that an input image patch $\boldsymbol{x}$ can be linearly reconstructed by its neighbors $\widetilde{\boldsymbol{Y}}$ in a small spatial neighborhood [9,10].For a given test image patch $\boldsymbol{x}$ with unknown density, we compute its reconstruction weights $\boldsymbol{w}$ by minimizing the reconstruction error.

$$\boldsymbol{w}^* = \arg\min_{\boldsymbol{w}} ||\boldsymbol{x} - \widetilde{\boldsymbol{Y}}\boldsymbol{w}||_2^2 \tag{2}$$

As assumed in [16] that two manifolds, which are formed by image patches and their counterpart density maps respectively, share the similar local geometry. Eqn. (2) obtains such geometry from image patches. Thus, the density map $\boldsymbol{x}^d$ will be predicted using the reconstruction weights and the neighboring patches $\widetilde{\boldsymbol{Y}}^d$ on density maps.

$$\boldsymbol{x}^d \cong \widetilde{\boldsymbol{Y}}^d \boldsymbol{w}^* \tag{3}$$

2.2. Object pre-localization via density map

From the definition of density map given in Eqn. (1), we can see that it depicts object density in each image pixel and shows the spatial information distribution of objects. It is clear that the rough location of each object can be obtained by finding every local maximum on the given region of density map. Specifically, given the estimated density map $\boldsymbol{x}^d$ for an image $\boldsymbol{I}$, a set of sliding windows $\boldsymbol{S}_1, \boldsymbol{S}_2, \dots, \boldsymbol{S}_M$ is defined to locate the local maximums on the estimated density map. The size of sliding windows is set as the average object size. The sliding windows move vertically or horizontally at a fixed step. The density patch in each window $\boldsymbol{S}_i (1 \le i \le M)$ is represented as $\boldsymbol{z}_i^d$, $\boldsymbol{z}_i^d(p) = \boldsymbol{x}^d(p)(p \in \boldsymbol{S}_i)$. The number of objects n_i in each window $\boldsymbol{S}_i$ can be computed by integrating over $\boldsymbol{z}_i^d$.

$$n_i = \sum\nolimits_{p \in \boldsymbol{z}_i^d} \boldsymbol{z}_i^d(p) \tag{4}$$

If n_i exceeds the density threshold c_b, it is considered that there is more than one object in window $\boldsymbol{S}_i$. Then the local maximum can be computed within $\boldsymbol{z}_i^d$. Each pixel of the estimated local maximum is formulated as

$$p_i^* = \arg\max_{p \in \boldsymbol{z}_i^d} \boldsymbol{z}_i^d(p), \text{ s.t. } n_i > c_b \tag{5}$$

The coordinates of p_i^* are regarded as the estimated object location. Typical value of c_b is set from 0.7 to 1, resulting in a looser or tighter constraint of the localization process.

2.3. Bounding box estimation by saliency map

Density map gives the object locations, but it lacks the detailed information of objects from image domain, especially the object boundary which is significant for object detection. Inspired by the human visual attention mechanism, saliency map estimation which provides object boundaries is introduced to estimate the accurate bounding box with the support of pre-localization results.

The process of bounding box estimation is realized by superpixel level, which refers to perceptually meaningful patches formed by adjacent pixels. Compact and highly uniform superpixels that respect image boundaries generated by the simple linear iterative clustering (SLIC) algorithm [12] are desirable for edge extraction of the small objects. Hence, we use SLIC to segment a test image into *N* small superpixels.

As the target objects are densely distributed in crowded scenes, typical background-based saliency estimation methods [11] fail to handle the task. Thus, we adjust the method from [11] to a saliency estimation method based on incomplete foreground localized in Sec.2.2. Given the object location estimated by density map, we employ the superpixels which contain the pixels $p_i^*(1 \leq i \leq M)$ of estimated object location as foreground seeds. The set of all the foreground seeds is represented as $\boldsymbol{E}$. To adapt the diversity of foreground seeds, K-means is employed to divide the foreground seeds into *K* clusters based on CIE-LAB color features, where *K* is set to 3 empirically in this paper. From these clusters, *K* different global color distinction (GCD) maps are computed as *K* sub-maps of saliency. The *k*-th GCD map measures the similarity of each superpixel *t* in image $\boldsymbol{I}$ and the foreground seeds in the *k*-th cluster (*k* = 1, 2, · · ·, *K*). The element $s_{k,t}$ in the GCD matrix is defined as the saliency value of each superpixel *t* in the *k*-th GCD map and is computed as:

$$s_{k,t} = \frac{1}{p^k}\sum_{j=1}^{p^k} 1/\, e^{-\frac{\|c_t, c_j\|}{2\sigma_1^2}} + \beta \qquad (6)$$

where p^k represents the number of foreground seeds belonging to cluster k and $\|c_t, c_j\|$ is the Euclidean Distance between the superpixel *t* and *j* in CIE-LAB color space. We set the balance weight σ_1= 0.2 and β= 10 as [11].

Once the GCD maps are computed, the superpixels with their saliency value approximated to the seeds are selected, then the bounding boxes can be estimated by the boundaries of the selected superpixels. Specifically, based on the *k*-th GCD map, we first compute the mean m_k and standard deviation σ_k of the saliency values from all the foreground seeds $s_{k,t}(t \in \boldsymbol{E})$.Then, traverse the *k*-th GCD map to seek for a set of superpixels $\boldsymbol{q}^k$ with each saliency value$s_{k,q_n^k} \in [m_k - \sigma_k, m_k + \sigma_k]$, where *n* indicates the index of each superpixel in $\boldsymbol{q}^k$. After selecting the target superpixels $\boldsymbol{q}^k$in *K* maps respectively, the superpixels $\boldsymbol{q}^k$ from *K* GCD maps are combined together without duplicate, the final set of the selected superpixels is represented as $\boldsymbol{q}$ .The selected superpixels are considered to be the target objects and the bounding box of each target object is designed as $\boldsymbol{B}$=(x_1, y_1, x_2, y_2) =(min($\boldsymbol{x}_q$),min($\boldsymbol{y}_{\mathrm{q}}$), max($\boldsymbol{x}_q$),max($\boldsymbol{y}_{\mathrm{q}}$)),where $(\boldsymbol{x}_q, \boldsymbol{y}_q)$ represents the coordinates of all pixels in the superpixel q, (x_1, y_1, x_2, y_2) represents the coordinates of the predicted rectangle bounding box.

The process of bounding box estimation is shown in Fig.3. In the third and fourth column, darker superpixels are more similar with the seeds in their corresponding foreground cluster, so they are also more likely to be selected as the final target objects. By traversing the sub-maps of saliency, our bounding box estimation method supplements the incomplete localization result in Sec.2.1, making our proposed small object detection method more independent on the localization result.

2.4. Post-processing

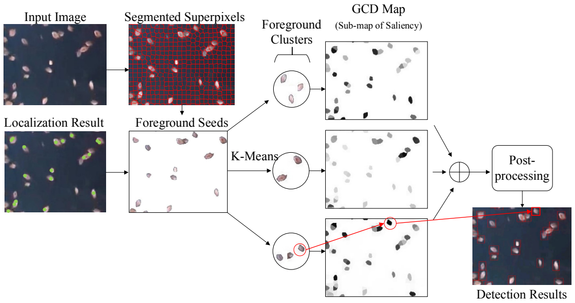


Fig. 3. The illustration of bounding box estimation

Considering that bounding box estimation of our method is based on segmented superpixels, one undesirable condition may exist that some objects are segmented into more than one superpixels. In such condition, one complete object may be partially detected with more than one bounding boxes. To get one complete bounding box for each whole object, the superpixels with incomplete objects are meant to be correctly grouped to their corresponding objects.

As the integral of density over ROI indicates the number of objects in it, we solve the above-mentioned problem via integrating over the bounding boxes in estimated density map $\boldsymbol{x}^d$. For each bounding box B_i, we go through its spatial neighbors within distance d from centroids. If the bounding box and its neighboring box B_j satisfy the condition $\sum_{p \in B_i \cup B_j} \boldsymbol{x}^d(p) < d_b$, the two bounding boxes B_i and B_j are considered to be the parts of a complete object, where counting threshold $d_b = 1$ in general. Then we merge each pair of the selected bounding boxes (e.g. B_i and B_j) into one complete bounding box to optimize the partially detection result.

3. EXPERIMENTS

In order to demonstrate the effectiveness of our proposed small object detection (SOD) method, we conduct several experiments on three public small object datasets [8] and a self-built snipe dataset. The performance is evaluated by precision P, recall R, and F_1 scores, where P is the fraction of detections that are matched with ground-truth; R is the fraction of ground-truth that are paired with detections and F_1 is defined as $\frac{2PR}{P+R}$. Detections are judged to be true/false positives by measuring bounding box overlap ratio that are defined in [13]. A detection result is considered to be true positive if the overlap ratio is larger than 0.5, and vice-versa.

3.1. Public small object datasets

Public small object datasets include fly, fish and seagull datasets with low/medium dense distribution and slightly overlap [8]. Following the settings in [8], 32 images are used for training and 64/50 images are used for testing. For seagull dataset, one high-resolution image (624×964) is used for training and another one is used for testing. The number of superpixels *N* is respectively set as 1000, 1100, 6200 for fly, fish, seagull datasets.

3.2. Self-built snipe dataset

To evaluate the generalization performance of our proposed SOD method, we collect bird flock images from the nature reserve in Shenzhen and build a snipe dataset. It is observed that this snipe dataset includes some severe overlapped objects. Specifically, the snipe dataset contains 4 high-resolution images (608×912) with an average of 265 ± 6 snipes. In order to evaluate the adaptation of different detection methods when the training set is small, we use 2 images for training and another 2 images for testing in our experiments.

Table 1 Detection results on small object datasets and snipe dataset

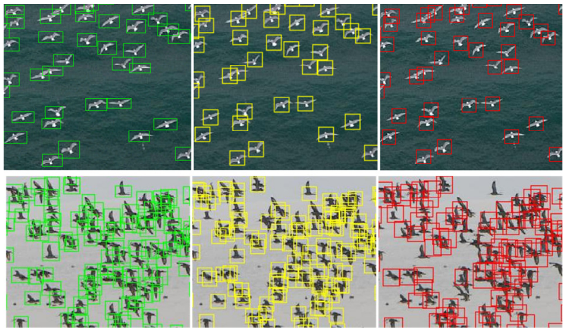
Method	Fly			Fish			Seagull			Snipe		
	R%	P%	F_1%	R%	P%	F_1%	R%	P%	F_1%	R%	P%	F_1%
HOG+SVM[2]	-	-	-	71.40	19.95	31.18	53.15	33.77	41.30	59.42	27.94	38.01
RPN+ZF[6]/300 proposals	-	-	-	61.97	64.23	63.08	22.05	70.82	33.63	15.04	36.32	21.27
RPN+ZF[6]/3000 proposals	-	-	-	62.44	63.23	62.83	57.61	75.40	65.31	31.70	36.50	33.93
SW[8]/our density	47.10	31.34	37.64	52.29	33.94	41.47	55.38	27.81	37.02	36.59	19.71	25.62
LM[8]/our density	63.96	64.43	64.19	71.88	71.85	71.87	74.80	88.20	80.59	56.52	58.63	57.56
Ours	70.39	69.63	**70.01**	82.81	84.46	**83.63**	82.81	84.27	**83.53**	77.72	79.53	**78.61**

Three different types of object detection methods have been taken to evaluate the performance over different datasets. Specifically, the comparison methods include the general sliding window based detection approach (e.g. HOG+SVM [2]), the region proposal based detection approach (e.g. Faster RCNN [6]) and the small instance detection baselines simply based on density maps (e.g. SW,LM) [8]. It is noted that all the training sets in our experiments are only annotated with dots in the center of each object. However, HOG+SVM and Faster RCNN generally require the bounding box annotation. In this study, we generate the bounding box annotation by locating a square bounding box of average object size around the labeled dot. Moreover, for Faster RCNN method, the ImageNet pre-trained ZF net [18] that has 5 conv layers and 3 fc layers is used. The training sets for Faster RCNN are split into two groups, three quarters of the training images are used for training and one quarter of them are for validation. For density based small object detection methods [8], the same density map estimated by [16] is used for localizing objects.

The detection results comparison are presented in Table 1. On all four datasets, the overall performance is favorable to our method. For the snipe dataset, under the condition of small training set scale as well as severe overlap between the objects, the F_1 score of our method significantly outperforms that of other methods by at least 21.05%. This indicates that our method is less affected by the overlapped objects since it benefits from superpixel segmentation by using global image information. Moreover, the experimental results of snipe dataset indicate that our method is less affected by the training set scale. To further validate the effect of training set scale, we use smaller training sets for fly and fish datasets which include only 5 images respectively to resume the experiments. F_1 scores decrease by only 2.21%/0.23% for the fly/fish small training datasets in our SOD method, whereas F_1 scores decrease by 11.32%/7.64% in LM for the same settings. Therefore, it is clear to see that our SOD method only asks for small training datasets which is a good candidate for small scale datasets.

It has been proved in recent research that Faster RCNN achieves excellent performance in general detection conditions [6]. However, for SOD problems in our experiments, the performance declines when the number of small size objects increases. It can be observed from Table 1 that, for seagull and snipe datasets where about 200 objects present in each image, the detection results of Faster RCNN are related to the number of proposals. Insufficient proposals lead to low recall.

Furthermore, as shown in Fig. 4, the bounding box of our SOD method is shape-related which fits the objects much more accurately compared to other methods. In the following, we make efforts to quantitatively evaluate the bounding box fitting accuracy of our SOD method. It is noted that the ground-truth object boundaries are unavailable but the ground-truth bounding boxes are provided in the testing datasets. Therefore, we take the ground-truth bounding boxes as standard. We use stricter evaluation criterion by increasing the bounding box overlap ratio from 0.5 to 0.6 and resume the experiments. Larger overlap ratio means a detection is judged to be true only when the detection is much closer to the ground-truth bounding box in terms of object location as well as the shape fitting accuracy. With the overlap ratio set to 0.6, F_1 scores decrease by 20.98% / 11.36% / 20.94% / 22.77% for fly / fish / seagull / snipe respectively in our SOD method, whereas F_1 scores decrease by 27.29%/18.79%/ 26.52%/35.34% in LM. As the evaluation criterion becomes stricter, F_1 scores of our method have a smaller decline, which indicates that our predicted bounding boxes are more accurate in fitting the ground-truth bounding boxes.



(a) Ours (b) LM [8] (c) Faster RCNN [6]

Fig. 4. Detection results comparison on (row 1) seagulls and (row 2) snipes.

4. CONCLUSION

In this paper, we propose an accurate small object detection method by exploring from both the image and its estimated density map. Our method takes advantage of the object spatial distribution information in density map but avoids its drawback of obscure object boundary. Although our method is trained by dotted annotation datasets, the estimated bounding box fits the object accurately due to the sufficient boundary information provided by saliency map. Experimental results validate the effectiveness of our small object detection method over three public small object datasets and a self-built snipe dataset which include limited scale of the training datasets and overlapped objects. The accurate detection results are obviously conducive to further application, such as object recognition.

ACKNOWLEDGMENT

This work was partially supported by the Shenzhen Science & Technology Fundamental Research Program (No: JCYJ2016 0330095814461&No: JCYJ20150430162332418).

REFERENCE

[1] A. Monroy and B. Ommer, "Beyond Bounding-Boxes: Learning Object Shape by Model-Driven Grouping," in *European Conference on Computer Vision,* vol. 7574, pp. 580-593, 2012.

[2] N. Dalal and B. Triggs, "Histograms of Oriented Gradients for Human Detection," in *IEEE Conference on Computer Vision & Pattern Recognition*, pp. 886-893, 2013.

[3] P. Viola and M. Jones, "Rapid Object Detection using a Boosted Cascade of Simple Features," *Proc Cvpr,* vol. 1, p. 511, 2001.

[4] R. Girshick, J. Donahue, T. Darrell, and J. Malik, "Rich Feature Hierarchies for Accurate Object Detection and Semantic Segmentation," *Computer Science,* pp. 580-587, 2013.

[5] R. Girshick, "Fast R-CNN," *Computer Science,* 2015.

[6] 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 & Machine Intelligence,* pp. 1-1, 2015.

[7] V. S. Lempitsky and A. Zisserman, "Learning To Count Objects in Images," in *Advances in Neural Information Processing Systems,* pp. 1591-1591*,* 2010.

[8] Z. Ma, L. Yu, and A. B. Chan, "Small Instance Detection by Integer Programming on Object Density Maps," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 3689-3697, 2015.

[9] S. T. Roweis and L. K. Saul, "Nonlinear Dimensionality Reduction by Locally Linear Embedding," *Science,* vol. 290, pp. 2323-6, 2000.

[10] H. Chang, D. Y. Yeung, and Y. Xiong, "Super-resolution through neighbor embedding," in *IEEE Computer Society Conference on Computer Vision & Pattern Recognition*, pp. 275-282, 2004.

[11] Y. Qin, H. Lu, Y. Xu, and H. Wang, "Saliency detection via Cellular Automata," in *IEEE Conference on Computer Vision & Pattern Recognition*, pp. 110-119, 2015.

[12] R. Achanta, A. Shaji, K. Smith, A. Lucchi, P. Fua, and S. Süsstrunk, "SLIC superpixels," *Epfl,* 2010.

[13] M. Everingham, L. V. Gool, C. K. I. Williams, J. Winn, and A. Zisserman, "The Pascal Visual Object Classes (VOC) Challenge," *International Journal of Computer Vision,* vol. 88, pp. 303-338, 2010.

[14] P. Felzenszwalb, D. Mcallester, and D. Ramanan, "A discriminatively trained, multiscale, deformable part model," in *IEEE Conference on Computer Vision & Pattern Recognition*, pp. 1-8, 2008.

[15] Y. Wang, Y. Zou, J. Chen, X. Huang, C. Cai, "Example-based visual object counting with a sparsity constraint," in *IEEE International Conference on Multimedia and Expo*, 2016.

[16] Y. Wang, Y. Zou, " Fast visual object counting via example-based density estimation," in *IEEE International Conference on Image Processing*, pp. 3653–3657, 2016.

[17] J. Yan,Y. Yu，X. Zhu, Z. Lei, S. Li. "Object detection by labeling superpixels." in *IEEE Conference on Computer Vision & Pattern Recognition*, pp. 5107-5116, 2015.

[18] M. D. Zeiler, R. Fergus. "Visualizing and understanding convolutional networks," in *European conference on computer vision* , pp. 818-833,2014.