Displays 76 (2023) 102282

Contents lists available at ScienceDirect

Displays

journal homepage: www.elsevier.com/locate/displa

Deep active learning with Weighting filter for object detection ☆

Wei Huang [a,*], Shuzhou Sun [b,c], Xiao Lin [b,c,*], Dawei Zhang [a], Lizhuang Ma [d]

[a] School of Optical-Electrical and Computer Engineering, University of Shanghai for Science and Technology, Shanghai 200093, China
[b] College of Information, Mechanical and Electrical Engineering, Shanghai Normal University, Shanghai 201418, China
[c] Shanghai Engineering Research Center of Intelligent Education and Bigdata, Shanghai Normal University, Shanghai 200234, China
[d] Department of Computer and Engineering, Shanghai Jiao Tong University, Shanghai 200240, China

ARTICLE INFO

MSC:
41A05
41A10
65D05
65D17

Keywords:
Active learning
Object detection
Neural network

ABSTRACT

Active learning has been demonstrated to be effective in reducing labeling costs by selecting the most valuable data from the unlabeled pool. However, the training data of the first epoch in almost all active learning methods is randomly selected, which will cause an instability learning process. Additionally, current active learning, especially uncertainty-based active learning methods, is prone to the problem of data bias because model learning inevitably prefers partial data. For the above issues, we propose Weighting filter (W-filter) tailored for object detection in this paper, which is an image filtering algorithm that can calculate the contribution of a single image to the neural network training as well as remove similar ones in the entire selected data to optimize the sampling results. We first use W-filter to select the training data of the first epoch, which can guarantee better performance and a more stable learning process. Then, we propose to resample the uncertain data from the perspective of the frequency domain to alleviate the problem of data bias. Finally, we redesign several classical uncertainty methods specifically for classification to make them more suitable for the task of object detection. We do rigorous experiments on standard benchmark datasets to validate our work. Several classical detectors such as Faster R-CNN, SSD, R-FCN, CenterNet, EfficientDet, and effective networks including ResNet, DarkNet, MobileNet are used in experiments, which shows our framework is detector-agnostic and network-agnostic and thus can meet any detection scenario.

1. Introduction

Deep neural network has achieved remarkable success in object detection, and several classical detectors [1–6] have obtained superior performance on standard benchmarks such as PASCAL VOC [7] and COCO [8]. But those detectors often rely on large-scale labeled training data, this requires very expensive labeling costs, especially in areas that need professional knowledge. To this problem, active learning selects a batch of the most valuable data from a large amount of unlabeled pool, so as to improve the quality of training data as much as possible under the limited labeling budget [9]. The above process can be repeated many epochs until the labeling budget is exhausted or the model reaches the expected performance. From the perspective of application scenarios, the existing active learning approaches are mainly divided into membership query synthesis [10,11], stream-based [12,13] selective sampling and pool-based methods [14,15]. The first method means that model can generate a specific distribution of data for subsequent learning. And the second refers to the selection model that can only obtain part of the unlabeled data at a time while the last one can get all. In this era of flooding of data, obviously, the last one has more practical value.

In this paper, we mainly focus on improving pool-based active learning for object detection. Given a pool of unlabeled data, pool-based methods can be further divided into the following three forms according to the different query criteria: (i) uncertainty-based approach [14,16]. This type of technology design function can calculate data uncertainty. But static functions often prefer partial data and cause sampling bias, that is, selected data cannot represent the entire unlabeled data. Meanwhile, most of the existing uncertainty approaches focus on image classification and that for object detection is rarely studied, (ii) diversity-based approach [17,18]. Diversity approaches preferentially select the batch of data with the most dispersed feature distance. Obviously, for more complex tasks, extra feature engineering will be required to ensure performance, (iii) expected model change approach [19,20]. This kind of approach usually takes processed unlabeled data (e.g., adding noise) as inputs to observe the changes in the model outputs. It is very time-consuming and laborious in the face

☆ This paper was recommended for publication by Prof. G. Guangtao Zhai.

* Corresponding authors.
E-mail address: 191380039@st.usst.edu.cn (W. Huang).

https://doi.org/10.1016/j.displa.2022.102282
Received 29 April 2022; Received in revised form 2 August 2022; Accepted 5 August 2022
Available online 17 September 2022

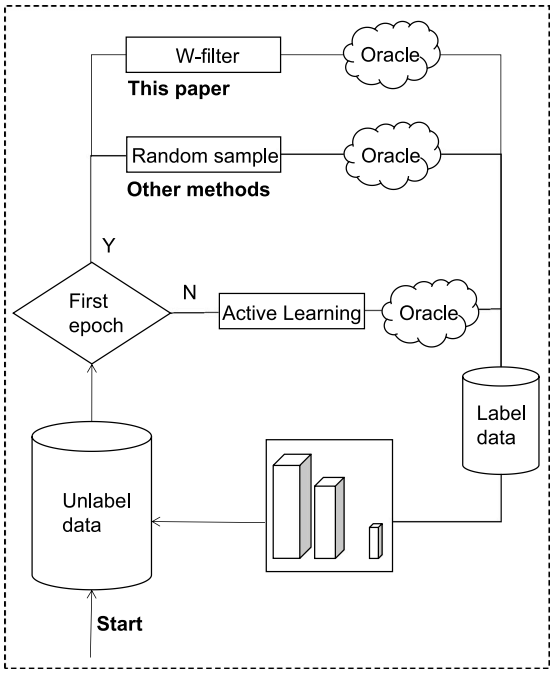

Fig. 1. The comparison of our proposed and other methods in the first epoch. The training data of the first epoch of other methods is randomly selected, while that in our proposed is selected by the active learning methods.

of large-scale unlabeled data or deep neural networks. Apart from the aforementioned problems, the training data of the first epoch in almost all existing active learning approaches are randomly selected. Obviously, this type of approach is difficult to obtain data that can represent the entire sample. This will not only cause an unstable learning process but also lead to poor performance of these active learning methods because most of them rely on the learning results of the first epoch to continue.

Lots of literature has observed that the deep neural networks are more sensitive to the edges of objects in images to be learned [21–24]. Inspired by those excellent works, we propose to calculate the value of images from the perspective of the frequency domain, because the edges of objects often corresponds to the high frequency. To achieve this goal, we propose Weighting filter (W-filter), which is an image filtering algorithm that can calculate the frequency domain information of images and remove similar ones in selected data. Note that the result of the W-filter is weighting frequency domain information rather than absolute information because we argue that those with more scattered edge information among data with the same level of frequency domain information are more valuable. Therefore we consider the frequency domain density in the design. That is, the selection result of the W-filter is not only rich in edge information but also more scattered. W-filter can calculate the value of data without relying on any model results, so we can first use it to select the training data of the first epoch. Compared with those methods that randomly select training data in the first epoch, our method obviously can obtain a more stable learning process and better performance. In Fig. 1, we show the comparison of our proposed method and other methods in the first epoch.

Additionally, many active learning methods, especially uncertainty-based methods, are prone to the problem of data bias. This is mainly due to the different proportion and difficulty of different classes in the training data, and the model after learning inevitably tends to go overboard on part classifications. When the data bias is severe, it can even cause the result to be weaker than random sampling. In this paper, we propose to use W-filter to alleviate the problem of data bias. Specifically, from the second epoch, we use W-filter to resample the selected uncertain data to remove similar ones. Our method removes part of the data with higher similarity from the perspective of the frequency domain, so the structure of selected uncertainty data can be optimized.

Finally, many classic uncertainty methods are specifically designed for classification. When faced with more complex data used in object detection, these methods cannot well balance the uncertainty among multiple objects. To this problem, we also redesign several classical uncertainty methods in this paper, including Least Confidence uncertainty approaches [16,25], Margin uncertainty approaches [26] and Entropy uncertainty approaches [27,28], to make them more suitable for the task of target detection uncertainty. In addition to the redesigned classic uncertainty methods, we will also compare the state-of-the-art uncertainty approaches in recent years, see Experimental Results for more. The main contributions of our work can be summarized as follows:

- We propose an image filter algorithm named Weighting filter (W-filter). Compared with those methods that randomly select training data in the first epoch, W-filter based active learning model can select the data that contributes the most to network learning for training and thus can obtain a more stable learning process and better performance. And to the best of our knowledge, this paper is the first work to focus on this issue.
- We propose a new uncertainty calculation mechanism, that is, resampling original uncertain data from the perspective of frequency domain information, which can remove similar data and thus can alleviate the problem of data bias in other active learning methods, especially uncertainty-based approaches.
- We improve several uncertainty methods originally designed for classification tasks such that these uncertainty estimation manners can be adapted for handing complex images in the object detection task. The reason behind is that our uncertainty computation method considers all objects in the image to calculate their uncertainty, thereby providing a more reliable uncertainty estimation than original uncertainty methods.

2. Related work

Pool-based methods for active learning mainly can be divided into uncertainty-based approaches [14,16], diversity-based approaches [17,18], and expected model change approaches [19,20] according to the query criteria. In this paper, we mainly focus on uncertainty-based approaches, Least confidence [16,25], Margin [26], Entropy [27,28] are both the classic methods in this type of approach. Least confidence calculates the uncertainty of the unlabeled data through the maximum predicted probability, while Margin considering the first two probabilities. When there are many classes in the unlabeled dataset, the above two methods will obviously lose a lot of important probability information. For this, Entropy takes account of all output probability while calculating the uncertainty.

Although the above uncertainty-based approaches have been proven effective in the face of the simple classification task, there are obviously the following problems when it is directly used for the task of large scale detection: (1) those uncertainty-based approaches only consider the output confidence, which will inevitably cause data bias while facing large scale unlabeled data because of the preference of the network learning process [29,30], (2) uncertainty approaches are based on the trained model to calculate uncertainty, which means that the first epoch of training data for active learning cannot be generated by these methods. So the first epoch training data in almost all existing active learning technology is randomly selected, which will undoubtedly cause unstable training process and worse performance, (3) different from the data in classification, the images in detection often belong to different classes and have different learning difficulties [27]. The above three classic uncertainty approaches are designed for classification, so

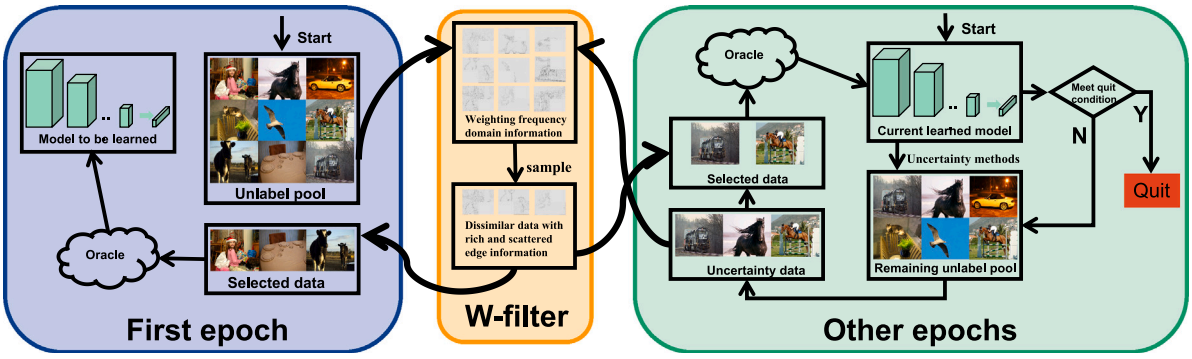

Fig. 2. Deep active learning structure based on W-filter and redesigned uncertainty approaches. Our framework uses W-filter for data sampling in the first epoch while W-filter and redesigned uncertainty approaches are jointly used in the other epochs. And the quit condition is the label budget is exhausted or the expected performance of the model is reached.

using them directly in detection will obviously ignore the relationship and difference between objects in the image, which is obviously unreasonable.

Apart from the above three classical uncertainty approaches, some recent efforts seek improvement from multiple perspectives. Loss prediction module [25] is jointly trained with the target model, and it can be used to predict the target loss of unlabeled inputs. But this method will increase the cost of network training. Localization-Aware Active Learning [31] proposed localization tightness and localization stability to calculate uncertainty, but it needs the network to provide intermediate prediction results (e.g., predictions by Region Proposal Network (RPN) in Faster R-CNN [1]), which means that this method cannot be used in model without intermediate prediction (e.g., one-stage object detectors [5,6]). Ensemble-based method [14] uses five committee networks to calculate uncertainty, which is obviously impractical in the face of large-scale unlabeled data and deep neural networks.

In this paper, we propose a novel active learning approach for object detection. In the first epoch, we use W-filter for sampling to ensure a stable training process and better performance. While for the latter epochs, we combine W-filter and redesigned uncertainty approaches to calculate the uncertainty, which can significantly alleviate the problem of data bias.

3. Method

This section presents implementation methods and details of our proposed method. First, we will introduce W-filter, which can select a batch of training data most suitable for the deep neural network in the first epoch and cooperate with uncertainty methods to complete active learning in subsequent epochs. Then, we will redesign several classic uncertainty approaches, including Least Confidence uncertainty approaches, Margin uncertainty approaches, Entropy uncertainty approaches, to make them more suitable for the task of object detection. Lastly, we will introduce how to combine W-filter and redesigned approaches to calculating the uncertainty of the unlabeled data. Our proposed deep active learning structure see Fig. 2.

3.1. W-filter

Different from other active learning techniques that randomly select training data in the first epoch, we use W-filter to sample the training data suitable for the deep neural network. For an unlabeled image $I_{(h,w)}$, where h and w are the height and width of this image. We first use Fourier Transform (FT) to calculate its frequency domain information $F(u,v)$ and remove the low frequency, the results we denote $F(u,v)'$. And we use inverse Fourier Transform (iFT) to restore $F(u,v)'$ to image $I_{(H,W)}$. The above process can be calculated as follows:

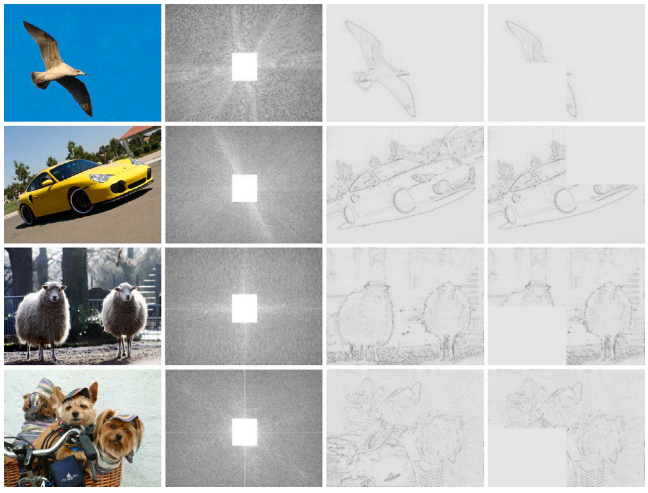

Fig. 3. The processing result of W-filter in the frequency domain. The first column is the original images $I_{(h,w)}$, and the second is their frequency domain $F(u,v)'$. The third and fourth columns are restored image $I_{(H,W)}$ and the results of W-filter $I_{(\mathcal{H},\mathcal{W})}$, respectively.

$$F(u,v) = \sum_{x=0}^{h-1}\sum_{y=0}^{w-1} I_{(h,w)}^{(x,y)} e^{-2j\Pi\left(\frac{ux}{h}+\frac{vy}{w}\right)} \tag{1}$$

$$F(u,v)' = \begin{cases} F(u,v), F(u,v) \geq f \\ 0, F(u,v) < f \end{cases} \tag{2}$$

$$I_{(H,W)} = \sum_{U=0}^{h-1}\sum_{V=0}^{w-1} F(U,V) e^{2j\Pi\left(\frac{UH}{h}+\frac{VW}{w}\right)} \tag{3}$$

Where $I_{(h,w)}^{(x,y)}$ is the frequency values of coordinate (x,y) in $I_{(h,w)}$. f is the threshold for removing low frequency.

Then, we divide $I_{(H,W)}$ into n regions evenly according to the area and remove the one with the largest frequency values. The reason we perform this, specifically, is to select images with more scattered edge information, i.e., those images should still have higher pixel values after removing the area with the largest pixel values. Removed region $R_{(h_n,w_n)}$ can be calculated as follows:

$$R_{(h_n,w_n)} = \arg\max \sum_{i=1}^{n}\sum_{x=0}^{h_n}\sum_{y=0}^{w_n} I_{(h_n,w_n)}^{(x,y)} \tag{4}$$

Where h_n and w_n are the height and width of divided region n respectively. And we denote the removed results $I_{(\mathcal{H},\mathcal{W})}$.

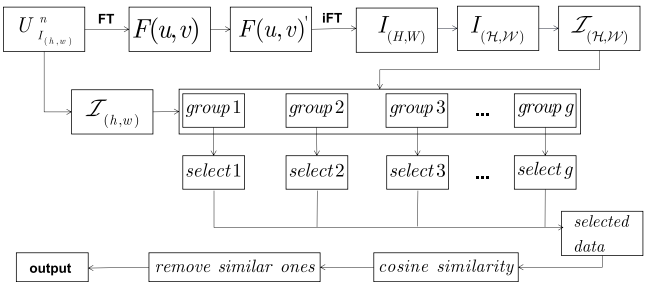

Fig. 4. The flow chat of W-filter. This module mainly completes data sampling in the first epoch, and cooperates with redesigned uncertainty approaches to calculate the image uncertainty in the later epochs.

Next, for each image $I_{(h,w)}$ and $I_{(\mathcal{H},\mathcal{W})}$ in unlabeled data $U^n_{I_{(h,w)}}$, we calculate the sum of frequency values and denote them as $\mathcal{I}_{(h,w)}$ and $\mathcal{I}_{(\mathcal{H},\mathcal{W})}$ respectively. We divide $U^n_{I_{(h,w)}}$ into g groups according to the $\mathcal{I}_{(h,w)}$, and select the data according to the $\mathcal{I}_{(\mathcal{H},\mathcal{W})}$ in each group. However, considering that the average frequency values of different group are not equal, it is unreasonable to take the same amount of samples from each group. For this, we define the sampling rate r_i here, it can be calculated as:

$$r_i = \frac{\frac{1}{s_i}\sum_{j=1}^{s_i}\mathcal{I}_{(h^j,w^j)}}{\frac{1}{\sum_{m=1}^{g}S_m}\sum_{n=1}^{g}\sum_{j=1}^{s_n}\mathcal{I}_{(h^j,w^j)}} \tag{5}$$

Where r_i represents the sampling rate of ith group, and S_i denotes the number of samples in ith group.

Finally, we remove the similar ones in the selected data to maximize the diversity of the samples. We use cosine similarity to calculate the similarity of weighting frequency domain information of images. For two weighting frequency domain information $f_x = [x_1, x_2, \dots, x_N]$ and $f_y = [y_1, y_2, \dots, y_N]$, its similarity can be calculated as:

$$S_{(f_x,f_y)} = \frac{\sum_{i=1}^{N} x_i \times y_i}{\sqrt{\sum_{i=1}^{N}(x_i)^2}\sqrt{\sum_{i=1}^{N}(y_i)^2}} \tag{6}$$

Here we show the processing results of W-filer in the frequency domain and its complete implementation process, see Fig. 3 and Fig. 4 respectively. Considering that the high frequency of images often corresponds to the edges of objects to be learned, W-filter prefers images with more and scatted edges, which obviously is in line with the expectation of the deep neural network for training data. To further prove this point, we select two images with the largest W-filter result and the other two with the smallest and obtain different channel feature maps based on the pre-trained model. According to the previous literature [21–24], the channel with large variance retains the most information. Therefore, for each image, we choose the five channels with the largest variance, the results are shown in Fig. 5. We can intuitively see that the images with large W-filter results include more features for deep neural network learning.

3.2. Redesigned uncertainty approaches

Most existing uncertainty approaches are tailored for classification. Although there has been lots of work in recent years to improve these approaches to fit other tasks, such as object detection, those improved methods are too expensive to be practical when faced with large-scale unlabeled data or deep neural networks. In this paper, we redesigned three classic uncertainty approaches used in classification, including Least Confidence uncertainty approaches, Margin uncertainty approaches and Entropy uncertainty approaches. The cooperation of W-filter and redesigned uncertainty approaches can help the task of object detection to complete active learning at a very low labeling

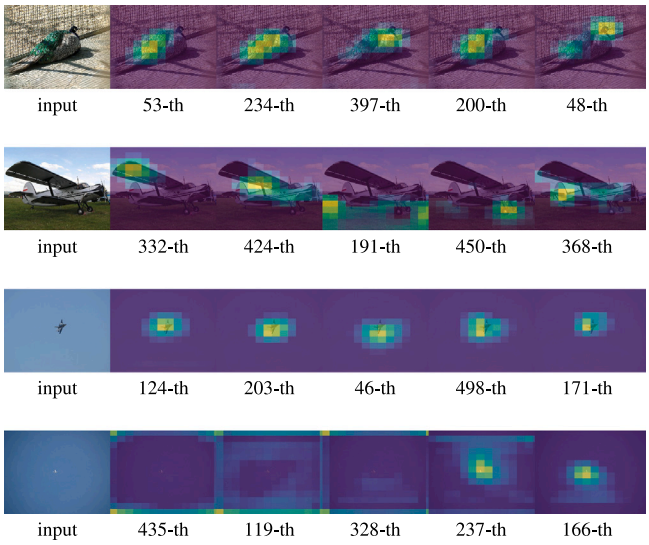

Fig. 5. Feature map of two images with the largest W-filter result (the top two lines) and the other two with the smallest (the bottom two lines). The pretrained model we use is VGG16 trained by ImageNet, the selected feature map is block5_conv3, and the five channels with the largest variance are showed. The dataset we use here is PASCAL VOC 2007.

cost, i.e., we can achieve high performance with less labeled data. For an image $I_{(h,w)}$ in unlabeled pool $U^n_{I_{(h,w)}}$ with c classes, $C = \{C_i\}_{i=1}^{n_I}$ represents the confidence of the prediction boxes obtained by the model inference, where n_I is the number of objects in the image $I_{(h,w)}$. Based on this definition, the uncertainty of image $I_{(h,w)}$ can be calculated.

Redesigned Least Confidence (RLC) uncertainty approaches. Least Confidence is a commonly used sampling method for classification. However, for image used in object detection, it often contains more than one object to be detected. Here, we let the sum of the uncertainties of all prediction boxes to represent the uncertainty of entire image. For image $I_{(h,w)}$, its RLC uncertainty $I^{RLC}_{(h,w)}$ can be calculated as:

$$\begin{aligned} I^{RLC}_{(h,w)} &= \textstyle\sum_{i=1}^{n_I}\left(1 - C_{\mathrm{i}}\right) \\ s.t. \quad C_{\mathrm{i}} &= \underset{j \in [1,\dots,c]}{\operatorname{argmax}}\, p_i^j \end{aligned} \tag{7}$$

Where p_i^j represents the confidence that the ith prediction box in image is jth class.

Redesigned Margin (RM) uncertainty approaches. Least confidence only calculates the maximum confidence of the prediction boxes, however, Margin considers the first two. For image $I_{(h,w)}$, its RM uncertainty $I^{RM}_{(h,w)}$ can be calculated as:

$$\begin{aligned} I^{RM}_{(h,w)} &= \textstyle\sum_{i=1}^{n_I}\left(C_{\mathrm{i}} - C'_{\mathrm{i}}\right) \\ \text{s.t.} \quad C_{\mathrm{i}} &= \underset{\mathrm{j} \in [1,\dots,c]}{\operatorname{argmax}}\, p_i^j \\ C'_{\mathrm{i}} &= \underset{\mathrm{j} \in [1,\dots,c]\backslash C_{\mathrm{i}}}{\operatorname{argmax}}\, p_i^j \end{aligned} \tag{8}$$

Where C'_{i} represents the second highest confidence of the ith prediction box in image $I_{(h,w)}$.

Redesigned Entropy (RE) uncertainty approaches. Although Margin considers the first two maximum confidence, this method will still overview lots of classes when there are many classes to be learned in unlabeled data. Entropy calculates the confidence of all classes for each prediction box. For image $I_{(h,w)}$, its uncertainty $I^{RE}_{(h,w)}$ can be calculated as:

$$I^{RE}_{(h,w)} = -\sum_{i=1}^{n_I}\sum_{j=1}^{C} p_i^j \log\left(p_i^j\right) \tag{9}$$

Table 1
The performance of original Least Confidence of the first two epoch. The model and dataset we used here are EfficientDet with backbone of EfficientDet-D0 and PASCAL VOC 2007. Considering the influence of random selection of training data in the first epoch, we repeat the experiment four times here.

	Person	Chair	Car	Bike	Bus	Bottle	Cow	Sofa	Bird	Mbike	Plant	Dog	Table	Horse	Sheep	Boat	Areo	Train	Cat	Tv	mAP
epoch 1	39.1	48.4	14.9	4.6	44.6	31.8	67.2	56.3	44.6	31.1	8.0	57.6	62.8	38.7	73.6	16.0	20.9	18.4	56.2	33.2	38.41
epoch 2	61.1	29.7	30.0	17.2	47.7	58.2	76.3	74.8	31.9	21.8	56.2	65.7	69.1	54.9	69.8	18.2	13.1	61.3	66.3	43.5	48.35
epoch 1	39.8	33.7	25.6	4.1	14.5	40.2	63.9	48.2	26.7	19.3	11.9	36.0	51.0	44.9	58.0	16.8	19.0	34.0	54.6	41.5	34.19
epoch 2	59.5	32.1	38.0	14.6	31.5	64.1	60.9	68.7	26.6	19.4	59.1	62.4	64.6	49.4	68.2	16.8	12.0	52.3	58.4	44.7	45.17
epoch 1	23.4	41.7	12.3	6.1	14.3	32.4	62.4	56.4	40.0	26.3	10.3	47.2	54.2	34.6	66.6	15.0	19.9	16.5	31.3	37.0	32.39
epoch 2	45.1	59.9	22.9	15.5	46.6	35.1	70.0	69.4	43.0	33.0	14.7	57.1	60.9	48.6	75.9	24.4	27.8	29.5	63.4	39.5	44.12
epoch 1	19.4	51.8	18.0	5.2	35.5	32.0	65.9	65.0	44.0	32.5	5.4	55.0	59.0	38.9	75.4	14.2	22.3	21.5	54.2	30.8	37.30
epoch 2	58.5	29.6	37.7	16.6	45.1	62.2	78.6	68.8	32.0	19.3	53.0	57.4	51.4	53.6	60.4	11.1	16.0	56.9	68.0	45.5	46.06

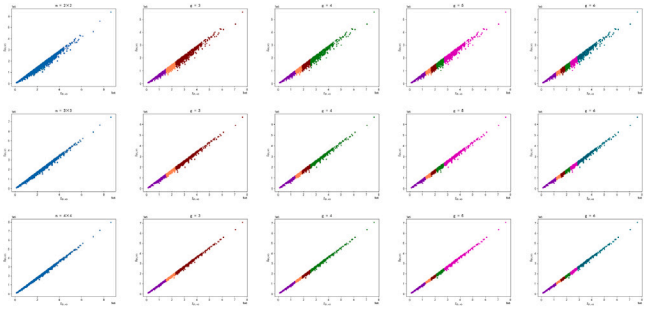

Fig. 6. Group results under different hyperparameter combinations. The dataset we used here is the training set of PASCAL VOC 2007.

Those redesigned approaches consider the uncertainty of all objects in the image rather than just one object. This is mainly because we have observed that the complexity of the images used for object detection is much greater than image classification, that is, the images used for detection often contain many objects, and the test confidence of these objects is different greatly because of the impact of category, size, etc.

3.3. Uncertainty for object detection

Although we have redesigned several classic active learning methods for classification, these approaches still only calculate the class uncertainty of objects in the image, and this will inevitably cause data bias. Here we propose an uncertainty calculation approach for object detection, that is, jointly using W-filter and redesigning uncertainty approaches to determine the uncertainty of the image. The selected data by our method is obviously more representative than just using uncertainty. Meanwhile, considering that W-filter prefers to select images with more scattered edges, we believe that the method proposed here is more suitable for calculating the uncertainty for detection.

Uncertainty in the first epoch. Unlike those active learning methods of random sampling in the first epoch, we use W-filter to calculate the uncertainty of the image, which can guarantee the performance and stability of the model. For each image $I_{(h,w)}$ in unlabeled data pool $U^n_{I_{(h,w)}}$, we first calculate $I_{(h,w)}$ and $I_{(\mathcal{H},\mathcal{W})}$. Then, we divide the data into g groups according to $I_{(h,w)}$ and $I_{(\mathcal{H},\mathcal{W})}$, and get sampling rate of each group. At last, we select the corresponding amount of data in these groups according to the sampling rate. The calculation details of the above process have been described in detail in Section 3.1, and knowing from that section there are two hyperparameters n and g in the calculation process. We visually show the sampling process under different hyperparameter combinations in Fig. 6. From Fig. 6, different combinations of n and g have little effect on the distribution of data, so we believe that the value of this set of hyperparameters will not have a great impact on our proposed method. In this paper, we setup n=2 × 2 and $g = 4$.

Uncertainty from the second epoch. From the second epoch, we jointly calculate the uncertainty of the images with W-filter and redesigned uncertainty approaches. We first calculate the uncertainty of images in unlabeled data by redesigned uncertainty approaches and select part of the data with larger uncertainty. Then, we use W-filter to resample the above-selected data, and the sampling rules are the same as approaches in the first epoch.

4. Experimental results

In this section, we firstly introduce the experiment details, including dataset introduction, comparing baselines and evaluating metrics. Next, we evaluate our proposed under the metric of fixed labeling budget. Then, performance under the another metric expected model performance also proves the efficiency of our work. Lastly, we further demonstrate the importance of W-filter and redesigned uncertainty approach through the ablation study.

4.1. Experiment details

Dataset. PASCAL VOC 2007 and VOC 2012 are both standard datasets commonly used for vision task such as classification [32], detection [1,3,5], segmentation [33], etc. We use them for verifying the deep active learning technology proposed in this paper. VOC 2007 [7] contains 20 object categories, including 2.5k training images, 2.5k validation images, and 5k test images. VOC 2012 is an augmented version of VOC 2007, which contains about 5k training images and 5k validation images. Although all objects in these two datasets contain classification and location annotations, we will only use selected images and their annotations obtained by active learning when training the model.

Comparing baselines. The following methods are used for comparing baselines here: (1) random sample: sampling the training data to be labeled uniformly at random from the unlabeled set, (2) redesigned classical uncertainty approaches: as described in Section 3.3, we redesign several classical uncertainty methods for classification to make them more suitable for object detection, (3) recent state-of-the-art methods: Loss Prediction Module (LPM) [25] train with target active model, and it can be used to predict the target loss of unlabeled inputs. For all the detectors used here, we use the same module and model connected to three layers of the target model. The internal structure and training of the module follow the setting of the original paper. Localization-Aware (L-Aware) [31] based active learning method uses localization tightness and localization stability to calculate uncertainty. For Faster R-CNN [1], we use the region proposals provided by its Region Proposal Network (RPN) to calculate Localization tightness. While for EfficientDet [3] and SSD [6], since they do not have intermediate proposal, we directly calculate localization stability.

Training details. In order to prove that our active learning framework has good generalization ability, we use multiple types of detectors, including EfficientDet [3], Faster R-CNN [1], SSD [6]. The setups of EfficientDet are: the backbone is EfficientDet-D0, optimizer is momentum, initial learning rate=0.08, the warmup learning rate is 0.001, and warmup steps is 2500, batchsize = 128. The hyperparameters of Faster R-CNN are: the backbone is ResNet-101 [34], optimizer is SGD,

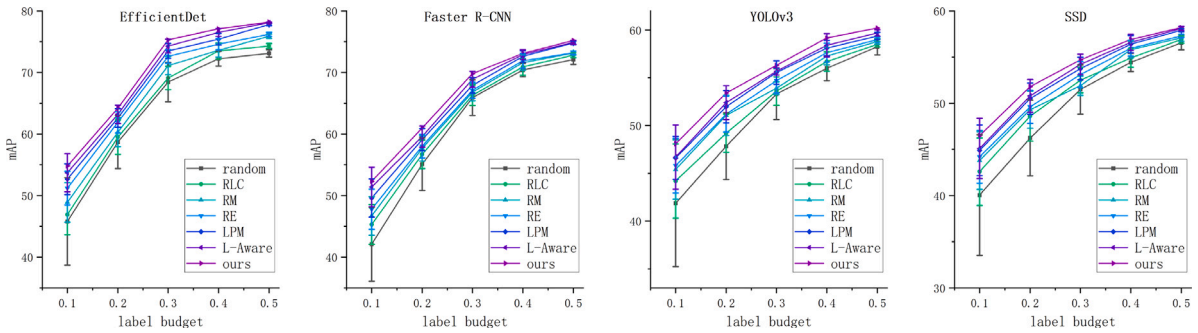

Fig. 7. Performance on PASCAL VOC 2007 under fixed labeling budget. We repeat each experiment five times and report standard deviation in the form of error bar. The compared baselines include classic uncertainty methods RLC [16], RM [26] and RE [28], recent state-of-the-art methods L-Aware [31] and LPM [25], and random sample.

weight_decay is 0.00005, learning_rate = 0.0001, batch_size = 32 . The setup of SSD including: the backbone is MobileNetV2 [35], optimizer is 0.94, batchsize = 24. The two evaluation metrics performance under fixed labeling budget and labeling budget under expected model performance are still followed here.

Evaluate metric. The major application scenario of deep active learning is to select the most valuable subset from a large amount of unlabeled datasets under a limited budget. Based on the scenario of this technology, we adopt two reasonable evaluation metrics in this paper.

Metric 1: performance under fixed labeling budget. Evaluate under this metric, the higher the model performance, the better the data selection technology. Considering that if the selected data requires a very large amount of labeling costs, then active learning loses its meaning. Therefore, the labeling budget we set during the experiment does not exceed 50% of the original unlabeled data.

Metric 2: labeling budget under expected model performance. Under this metric, the less the labeling cost, the better the sampling approaches. Similarly, considering the usage scenarios of active learning, the expected performance we set will be adjusted appropriately according to the specific model.

In addition to the above two evaluation metrics for active learning, our work is mainly used for detection. Here we use mean Average Precision (mAP) as the evaluation metrics of detectors, which considers both precision and recall rate at the same time.

4.2. Performance under fixed labeling budget

4.2.1. Experiment setup

Active learning details. This section we regard the training set of PASCAL VOC 2007 (about 2.5k images) as the original unlabeled data $U^{2.5k}_{I_{(h,w)}}$. We let 10% of $U^{2.5k}_{I_{(h,w)}}$ as the training data for the first epoch, and in each subsequent epochs, we select 5% of the original unlabeled data as the new training data. In order to make the experiment more convincing, we set up multiple labeling budgets and denote it as $\{b_i\}^{l}_{i=1}$, where b_i represents the labeling budget of ith experiment. The budgets we setup in this paper is {20%, 25%, 30%, 35%, 40%, 45%, 50% }.

Target model. The target object detectors we used in experiments cover two-stage (Faster R-CNN) and one-stage (EfficientDet, YOLOv3 and SSD), which shows that our framework is detector-agnostic and thus can assist networks with any structure perform active learning. For each object in the image, these three models can output the corresponding classification probability. We extract this probability for the active learning algorithm to calculate the uncertainty of the unlabeled images.

4.2.2. Performance analysis

We report the performance of our proposed deep active learning and baselines under fixed labeling budget in Fig. 7, and we have several observations: (1) Our approach exceeds the baselines including random sample, redesigned uncertainty approaches and recent state-of-the-art methods in the entire budget set. Meanwhile, the target models used here are Faster R-CNN, EfficientDet, and SSD, covering one-stage and two-stage detectors, which fully demonstrates that our deep active learning has good generalization in the face of different types of detectors, (2) for random sample and original uncertainty approaches, we repeat the experiment five times and report the mean and Standard Deviation (SD) of their performance. From Fig. 7, it can be seen that the stability of these comparing baseline methods is poor, that is, under a fixed labeling budget, the results of different experiments are very different. We argue that the source of this instability is due to their training data in the first epoch being randomly selected. We will continue to discuss this point in the following section, (3) our proposed only reports experiment results of one time. This is because its first training data is selected by W-filter, its entire active learning process will be relatively stable, (4) as the labeling budget increases, the advantages of our proposed gradually decrease. But considering that active learning is usually used in scenarios where the budget is very little, our method should perform well in realistic, (5) for classical uncertainty approaches, Entropy is the best and Least Confidence is the worst, which shows that it is effective to make full use of the confidence of different classifications of objects. As described in Section 3, redesigned uncertainty approaches calculate the classification uncertainty of all objects. We believe this is an important reason why our method can achieve good performance. We will prove this in the ablation study.

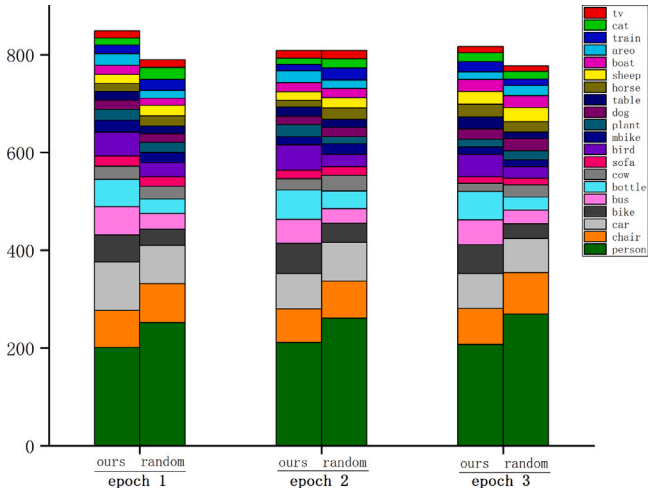

Fig. 8. The distribution of classes in the selected training data of the first three epochs. The model and dataset we used here are YOLOv3 and PASCAL VOC 2007 respectively.

4.2.3. Process analysis

In this section, we will further analyze the process to explore the difference between different sample methods. The labeling budgets we

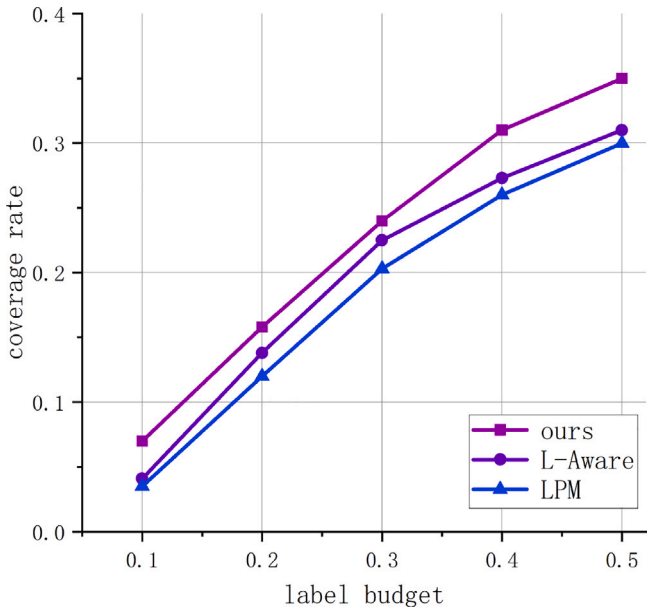

Fig. 9. Coverage rate under fixed labeled budget. The data and detector used here are EfficientDet and PASCAL VOC 2007 respectively, and the setup of the experiment is the same as that in Section 4.2. The compared baselines here are recent state-of-the-art methods L-Aware [31] and LPM [25].

set here are {20%, 25%, 30%, 35%, 40%, 45%, 50% } and the first epoch of training data is 10% of the original unlabeled data, so 9 epochs are required for active learning.

Compared with random sample. From Fig. 7, random sampling has the weakest performance and the worst stability, mainly because this method cannot adaptively adjust sampling preferences according to the learning status of the model. While our redesigned uncertainty approaches will select samples with the poor learning effect of the current model in each epoch, that is, these sampling preferences are dynamically adjusted. We show the distribution of classifications in selected data of our proposed and random sample in Fig. 8, and we can observe that classifications in different epochs selected by random samples probably obey the same distribution. According to the characteristics of random selection, we believe that this distribution should be the distribution of the classification in the original unlabeled data. While classification distribution selected by our proposed changes each epoch, and we believe this is the result of uncertainty approaches changing the sampling preferences.

Compared with classical uncertainty approaches. In this paper, we redesigned three classic uncertainty methods including Least Confidence, Margin, and Entropy as comparing baselines. Although these methods can adaptively change the sampling preference according to the current performance of the network, their processes of sampling preference adjustment are very unstable due to their random selection of training data in the first epoch. We report the performance of the first two epochs of different experiments in Table 1. From this table, the randomly selected training data in the first epoch leads to the very unstable performance of the model in this epoch. Not only that, this instability will continue to pass on to the next epoch, that is, the performance difference of the model in the second epoch is still obvious. The above-passed instability will obviously interfere with the calculation of image uncertainty. We have shown classification distribution in the first three epochs of different experiments in Fig. 8, and these results fully prove our point.

Compared with state-of-the-art methods. Our approach exceeds the recent state-of-the-art methods in the entire budget set. In order to further prove that this mainly benefited from the more optimized sampling results of our method, we define the metric of coverage rate here, which can be used to measure the coverage degree of the sampled data to the original unlabeled data. The coverage rate is $\mathcal{S}/\mathcal{O}$, where $\mathcal{O}$ is the original data and $\mathcal{S}$ is the residual data after removing a similar part from the sampled data. We report the coverage rate of different methods in Fig. 9, which shows that our proposed active learning framework yields better sampling results than recent state-of-the-art methods.

4.3. Labeling budget under expected performance

4.3.1. Experiment setup

Active learning details. In this section, we unite the training set of PASCAL VOC 2007 (about 2.5k images) and that of VOC 2012 (about 5k images) as original unlabeled data $U^{7.5k}_{I_{(h,w)}}$. we let 10% of $U^{7.5k}_{I_{(h,w)}}$ as the training set for the first epoch, and in each subsequent epoch, we select 5% of the original unlabeled data as the added training data. Considering that the different accuracy requirements of realistic tasks, we set multiple sets of expected performance and denote it as $\{(p_i, D)\}^{l}_{i=1}$, where p_i is ith expected performance and D represents the object detector. Different from the previous experiment setting the same labeling budge for various detectors, here we set a separate expected performance for each detector because the performance of these detectors is different.

Target model. Experiment here we use Faster R-CNN, R-FCN and SSD as target models for object detection. Similar to the above experiment, used detectors also cover one-stage and two-stage detectors. The original performance of Faster R-CNN with the backbone of ResNet-101 [34] trained by training set VOC 2007 and VOC 2012 is 76.4, so we set {(50,55,60,65,70),R-FCN} for this detector. The original performance of R-FCN with backbone of ResNet-101 [34] trained by training set VOC 2007 and VOC 2012 is 79.5, so we set {(55,60,65,70,75),R-FCN}. The original performance of SSD with backbone of MobileNetV2 [35] is 65.07, and the expected performance we set for it is {(40,45,50,55,60),SSD}. The method of calculating the uncertainty of the image here is the same as that used in the experiment in the above section.

4.3.2. Performance analysis

We report the performance of our proposed deep active learning and baselines under expected performance in Fig. 10, and we can conclude that: (1) our proposed is better than the baselines including the random sample method and original uncertainty approach, that is, our method uses fewer labeling budget for the same expected performance, (2) our proposed has a great advantage when expected performance is little, and we believe this benefit mainly stems from the training data in the first epoch sampled by W-filter, (3) as expected performance increases, the advantages of our method gradually become smaller. We conjecture that when expected performance approaches performance trained by the entire training set, our proposed will be close or even worse than the baselines, we believe that the main reason is that when the expected performance is very large, the training set required by the model will be close to the entire original training data. In this case, the effect of the W-filter will be very small or even negative because it may affect sampling preferences. However, considering that active learning is generally used in scenarios of very limited labeling budget, our method should maintain a considerable advantage in realistic, (4) for classical uncertainty approaches, here we can get a conclusion similar to the previous experiments, that is, Entropy performed the best and Least Confidence the worst.

4.3.3. Process analysis

Like process analysis in the previous section, the experiment here still shows that our proposed is more stable than comparing baselines. But apart from this, we find that our approach has another advantage. Under part expected performance, there will be epoch gap between our proposed and the comparing baseline, that is, our approach uses

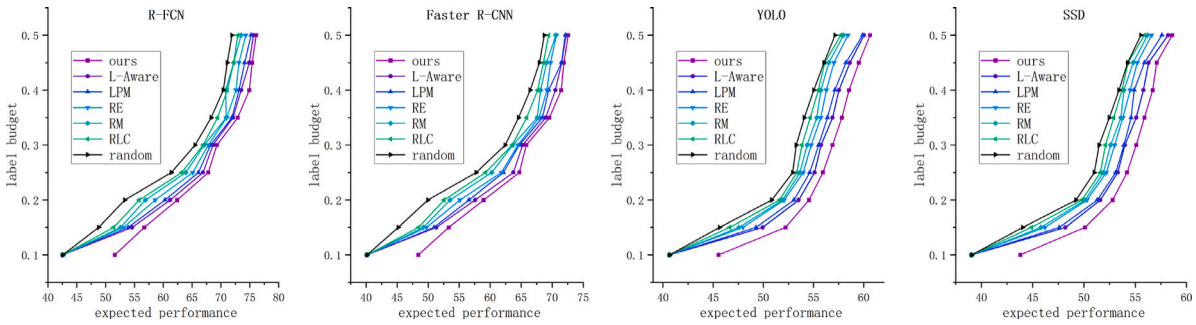

Fig. 10. Performance on unity dataset of PASCAL VOC 2007 and VOC 2012 under expected performance. We repeat each experiment five times and report the mean. The compared baselines include classic uncertainty methods RLC [16], RM [26] and RE [28], recent state-of-the-art methods L-Aware [31] and LPM [25], and random sample.

Table 2
The mean Average Precision (mAP) of with and without W-filter under fixed labeling budget.

Method	Backbone	Labeling budget							
		With W-filter				Without W-filter			
		0.2	0.25	0.3	0.35	0.2	0.25	0.3	0.35
R-FCN [2]	ResNet-101	62.4	67.8	69.4	74.9	61	65.2	68.2	71.9
EfficientDet [3]	D0	64.1	67.5	69.6	70.9	62.2	65.1	67.3	69.2
CenterNet [4]	ResNet-50	60.3	63.5	67.2	68.9	58.8	62.7	65.2	66.8
Faster R-CNN [1]	ResNet-101	61.0	66.3	69.8	71.9	58.7	64.2	68.1	70.3
SSD [6]	MobileNetV2	51.8	53.2	54.9	55.6	50.2	52.4	53.7	54.7

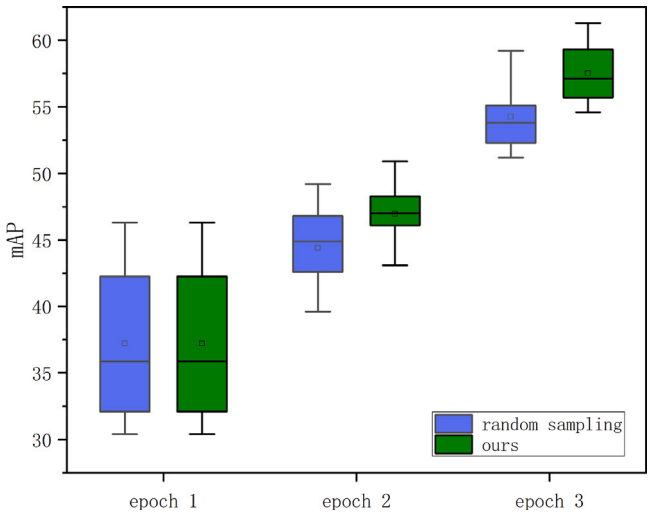

Fig. 11. Training performance under different data sampling methods. The model and dataset we used here are EfficientDet with backbone of EfficientDet-D0 and PASCAL VOC 2007. Note that for fair comparison, our proposed also uses random sampling in the first epoch.

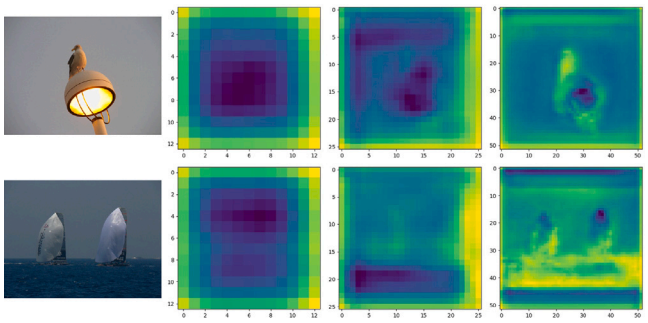

Fig. 12. Visualization of feature maps of different layers. The model we used here is YOLOv3 with backbone of DarkNet-53, and the sizes of the extracted feature maps are 12×12, 25×25, 50×50 respectively.

Table 3
The performance under different median filtering windows. $k = 1$ represents that training images are not processed.

Method	Backbone	Median filtering windows k			
		1	5	7	9
R-FCN [2]	ResNet-101	79.5	78.6	76.3	74.9
EfficientDet [3]	D0	77.8	77.1	74.8	72.3
CenterNet [4]	ResNet-50	70.9	70.1	68.5	67.2
Faster R-CNN [1]	ResNet-101	65.7	65.1	64.2	62.1
SSD [6]	MobileNetV2	76.4	75.8	73.6	71.8

fewer epochs (e.g., SSD with the backbone of MobileNetV2 uses 2 epochs while baseline of original Least Confidence needs 3 to meet the expected performance of 50). Epoch gap not only shows that our method can use less labeling budget but also means that more training resources can be saved.

4.4. Ablation study

4.4.1. Without W-filter

W-filter we proposed is an image filtering algorithm. It can complete data sampling in the first epoch and cooperate with redesigned uncertainty approaches to calculate the image uncertainty in other epochs. It is conceivable that without W-filter, the active learning process will become unstable because the first epoch of training data is randomly selected, and its performance will also deteriorate because sampling only through uncertainty will lead to data bias. To verify our conjecture, we report the performance of the first three epochs of the algorithm with and without W-filter in Fig. 11, and we can find that performance without W-filter is very unstable, which is obviously caused by the random selection of the training data in the first epoch. Meanwhile, we report performance under different labeling budgets in Table 2. Obviously, this result fully meets our expectations. In the following, we will explain why W-filter is effective from two perspectives.

Edges sensitivity. W-filter preferentially selects images with more and scattered edges information while the neural network is sensitive to the edges [21–24], so the selection result of W-filter just meets the expectation of the neural network to train data. Here we will prove through experiments that neural networks are sensitive to the edges

Table 4
The performance under redesigned uncertainty approaches and original uncertainty approaches. For a fairer comparison, original Least Confidence also used W-filter for resampling when calculating uncertainty.

Method	Backbone	labeling budget							
		W-filter + RE				W-filter + Entropy			
		0.2	0.25	0.3	0.35	0.2	0.25	0.3	0.35
R-FCN [2]	ResNet-101	62.4	67.8	69.4	74.9	61	61.5	68.5	73.5
EfficientDet [3]	D0	62	66.2	69.1	69.3	61	64.6	68.7	68.9
CenterNet [4]	ResNet-50	57.9	62.3	65.6	67.2	56.7	60.2	64.3	66.5
Faster R-CNN [1]	ResNet-101	61.0	66.3	69.8	71.9	60.4	65.2	68.3	71.2
SSD [6]	MobileNetV2	51.8	53.2	54.9	55.6	51.2	52.9	53.7	55.1

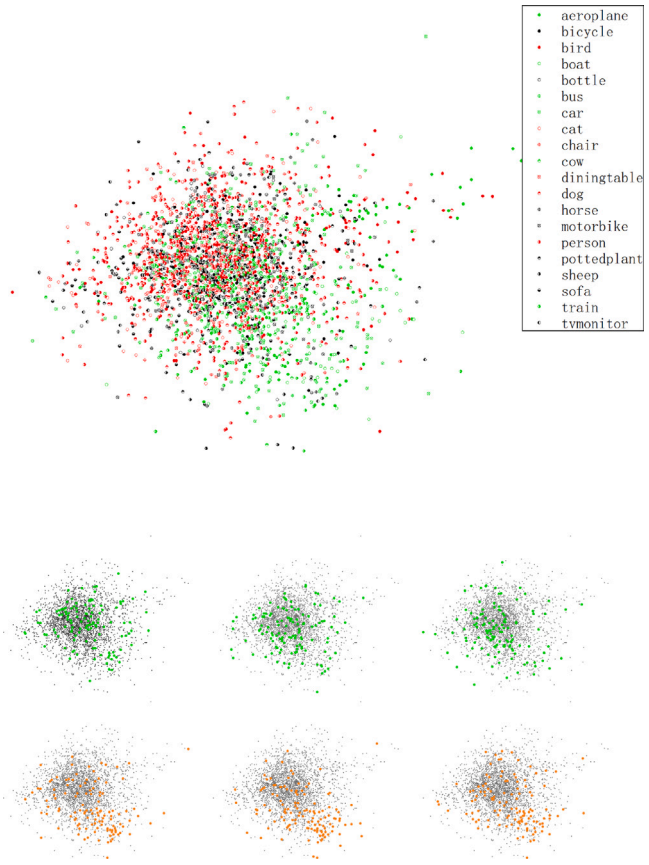

Fig. 13. t-SNE embedding of images on PASCAL VOC 2007 training set. The high-dimensional vectors of images are generated by YOLOv3 with backbone of DarkNet-53. The first line is the original distribution, and the other two lines are the sampling results of our proposed and original uncertainty approaches. The cyan points represent the sampling of our proposed, while the orange points are the original uncertainty approaches.

of objects to be detected. Median filtering is a nonlinear smoothing technology, it sets the gray value of each pixel as the median value of all pixels in a neighborhood window $W_{(k\times k)}$, where $k \times k$ represents the size of the window. Obviously, the larger the window of median filtering, the more blurred the edges of objects in the image processed by it. We report the detector performance trained by data processed by median filter with different sizes of windows in Table 3. And from this table, we can find that the more blurred the edges of objects in the image, the worse the learning effect of the neural network. Meanwhile, the results of features visualization of the trained model also show that the edges of objects are the focus of neural network learning. We visualize the feature maps of different layers of the network, which show the network learning process, see Fig. 12 for more. From the above experiments, we have full reasons to believe that those images with more edges and scattered can have the greatest improvement in the network training, and our proposed W-filter can find such data.

Data bias. Data bias is a tough issue conundrum for uncertainty approaches, and we believe that there are two main reasons. One is the calculation of uncertainty is based on trained networks or specific algorithms. So it will naturally have a preference for certain types of data. And the other is the randomly selected data in the first epoch is biased and not representative, and this bias will be passed on to subsequent epochs. Our proposed uses W-filter to sample the training data in the first epoch, which not only helps improve the accuracy of the model but also guarantees the stability of the active learning process. Meanwhile, unlike those methods that only consider classification confidence in unlabeled images, we resample data with high uncertainty through W-filter from the second epoch. We believe that our approaches can get training data with better distribution and the purpose is to remove similar ones in selected data. To prove this, we use t-distributed stochastic neighbor embedding (t-SNE) [36] to display the distribution of sampled data, which is a nonlinear data dimensionality reduction technology. The results show that the data selected by our proposed is obviously more representative, that is, it will balance the uncertainty and the value of the network. While the data selected by other methods only based on uncertainty approaches are obviously more biased towards partial data. See Fig. 13 for more details.

4.4.2. Without redesigned uncertainty approaches

This paper redesigns several classic uncertainty technologies for classification to make them more fit the task of object detection, those redesigned approaches consider the uncertainty of all objects in the image instead of just one. This improvement is mainly because we have observed that the complexity of the images used for detection is much greater than classification, that is, the images used for detection often contain many objects, and the test confidence of these objects is different greatly because of the impact of classification, size, etc. We report the performance of our proposed and original uncertainty approaches in Table 4. Note that, for fairness, the training data in the first epoch of all experiments here are sampled by W-filter. From the results in this table, our redesigned uncertainty approaches are more suitable for the task of object detection.

5. Conclusion and future work

In this paper, we propose a novel deep active learning framework for object detection. This technology is shown to alleviate the problem of data bias in previous uncertainty-based active learning methods. W-filter we proposed can also help active learning obtain a more stable training process and better performance because the first epoch training data in W-filter based framework is selected by active learning method while that of in other methods are randomly sampled. Meanwhile, we redesign several classical uncertainty approaches to make them more fit the task of object detection, and we have proved their effectiveness by ablation study.

Although our novel active learning approach has been effective for several classical detectors and networks, class differences and location

information of objects in unlabeled images were not considered, and we think those are also important for object detection. In the future, we will continue to improve our active learning framework based on the above points.

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Data availability

No data was used for the research described in the article.

References

[1] S. Ren, K. He, R. Girshick, J. Sun, Faster R-CNN: Towards real time object detection with region proposal networks, IEEE Trans. Pattern Anal. Mach. Intell. 39 (6) (2017) 1137–1149.

[2] J. Dai, Y. Li, K. He, J. Sun, R-FCN: Object detection via region-based fully convolutional networks, in: Proceedings of the 30th International Conference on Neural Information Processing Systems, Ser, NIPS'16, 2016, pp. 379–387.

[3] M. Tan, R. Pang, Q.V. Le, Efficientdet: Scalable and efficient object detection, in: Proc. IEEE Conf. Comput. Vis. Pattern Recognit, CVPR, Jun., 2020, pp. 10778–10787.

[4] T. Li, M. Ye, J. Ding, Discriminative hough context model for object detection, Vis. Comput. 30 (2014) 59–69.

[5] J. Redmon, A. Farhadi, YOLOv3: An incremental improvement, 2018, arXiv: 1804.02767, 04.

[6] W. Liu, D. Anguelov, D. Erhan, C. Szegedy, S. Reed, C.Y. Fu, A.C. Berg, SSD: Single shot multibox detector, in: Proc. Eur. Conf. Comput. Vis., Vol. 9905, ECCV, p. 2016.

[7] M. Everingham, L.V. Gool, C.K.I. Williams, J. Winn, A. Zisser man, The pascal visual object classes (VOC) challenge, Int. J. Comput. Vis. 88 (2) (2010) 303–338.

[8] J. Deng, W. Dong, R. Socher, L. Li, Kai. Li, Li. Fei-Fei, Imagenet: A large-scale hierarchical image database, in: Proc. IEEE Conf. Comput. Vis. Pattern Recognit, CVPR, Jun., 2009, pp. 248–255.

[9] D. Gudovskiy, A. Hodgkinson, T. Yamaguchi, S. Tsukizawa, Deep active learning for biased datasets via fisher kernel self-supervision, in: Proc. IEEE Conf. Comput. Vis. Pattern Recognit., CVPR, Jun., 2020, pp. 9038–9046.

[10] D. Angluin, Queries and concept learning, 1988, pp. 319–342, 2 (4).

[11] R.D. King, K.E. Whelan, F.M. Jones, P. Reiser, C.H. Bryant, S.H. Muggleton, D.B. Kell, S.G. Oliver, Functional genomic hypothesis generation and experimentation by a robot scientist, Nature 427 (6971) (2004) 247–252.

[12] I. Dagan, S.P. Engelson, Committee-based sampling for training probabilistic classifiers, in: Machine Learning Proceedings, 1995, pp. 150–157.

[13] V. Krishnamurthy, B. Wahlberg, Finite dimensional algorithms for optimal scheduling of hidden Markov model sensors, in: 2001 IEEE International Conference on Acoustics, Speech, and Signal Processing. Proceeding, Vol. 6, 2001, pp. 3973–3976.

[14] W.H. Beluch, T. Genewein, A. N¨urnberger, J.M. K¨ohler, The power of ensembles for active learning in image classification, in: Proc. IEEE Conf. Comput. Vis. Pattern Recognit., CVPR, Jun., 2018, pp. 9368–9377.

[15] Y. Shen, H. Yun, Z.C. Lipton, Y. Kronrod, A. Anandkumar, Deep active learning for named entity recognition, in: International Conference on Learning Representations, 2018.

[16] D.D. Lewis, W.A. Gale, A sequential algorithm for training text classifiers, in: Conference on Research and Development in Information Retrieval, 1994, pp. 3–12.

[17] Z. Wang, J. Ye, Querying discriminative and representative sam ples for batch mode active learning, in: International Conference on Knowledge Discovery and Data Mining, 2013, pp. 158–166.

[18] Y. Gal, R. Islam, Z. Ghahramani, Deep bayesian active learning with image data, in: Proceedings of the 34th International Conference on Machine Learning - Volume 70, Ser., ICML'17, 2017, pp. 1183–1192.

[19] N. Roy, A. McCallum, Toward optimal active learning through sampling estimation of error reduction, in: Eighteenth International Conference on Machine Learning, 2001, pp. 441–448.

[20] A. Freytag, E. Rodner, J. Denzler, Selecting influential examples: Active learning with expected model output changes, in: D. Fleet, T. Pajdla, B. Schiele, T. Tuytelaars (Eds.), Computer Vision, ECCV 2014, 2014, pp. 562–577.

[21] D. Bau, B. Zhou, A. Khosla, A. Oliva, A. Torralba, Network dissection: Quantifying interpretability of deep visual representations, in: Proc. IEEE Conf. Comput. Vis. Pattern Recognit., CVPR, Jun., 2017, pp. 3319–3327.

[22] W. Zeng, W. Luo, S. Suo, A. Sadat, B. Yang, S. Casas, R. Urtasun, End-to-end interpretable neural motion planner, in: Proc. IEEE Conf. Comput. Vis. Pattern Recognit., CVPR, Jun., 2019, pp. 8652–8661.

[23] P.E. Pope, S. Kolouri, M. Rostami, C.E. Martin, H. Hoffmann, Explainability methods for graph convolutional neural networks, in: Proc. IEEE Conf. Comput. Vis. Pattern Recognit., CVPR, Jun., 2019, pp. 10764–10773.

[24] H. Zhang, Z. Wang, D. Liu, A comprehensive review of stability analysis of continuous-time recurrent neural networks, IEEE Trans. Tions Neural Netw. Learn. Syst. 25 (7) (2014) 1229–1262.

[25] D. Yoo, I.S. Kweon, Learning loss for active learning, in: Proc. IEEE Conf. Comput. Vis. Pattern Recognit., CVPR, Jun., 2019, pp. 93–102.

[26] E. Elhamifar, G. Sapiro, A. Yang, S.S. Sasrty, A convex optimiza tion framework for active learning, in: Proc. IEEE Int. Conf. Comput. Vis., ICCV, Dec., 2013, pp. 209–216.

[27] H.H. Aghdam, A. Gonzalez-Garcia, A. L'opez, J. Weijer, Active learning for deep detection neural networks, in: Proc. IEEE Int. Conf. Comput. Vis., ICCV, Dec., 2019, pp. 3671–3679.

[28] A.J. Joshi, F. Porikli, N. Papanikolopoulos, Multi-class active learning for image classification, in: Proc. IEEE Conf. Comput. Vis. Pattern Recognit., CVPR, Jun., 2009, pp. 2372–2379.

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

[30] C. Huang, Y. Li, C.C. Loy, X. Tang, Learning deep representation for imbalanced classification, in: Proc. IEEE Conf. Comput. Vis. Pattern Recognit., CVPR, Jun., 2016, pp. 5375–5384.

[31] C.C. Kao, T.Y. Lee, P. Sen, M.Y. Liu, Localization-aware active learning for object detection, in: Proc. ACCV, 2019, pp. 506–522.

[32] K. Simonyan, A. Zisserman, Very deep convolutional networks for large-scale image recognition, Comput. Sci. (2014).

[33] Y. Chen, G. Lin, S. Li, O. Bourahla, Y. Wu, F. Wang, J. Feng, M. Xu, X. Li, BANet: Bidirectional aggregation network with occlusion handling for panoptic segmentation, in: Proc. IEEE Conf. Comput. Vis. Pattern Recognit., CVPR, Jun., 2020, pp. 3792–3801.

[34] K. He, X. Zhang, S. Ren, J. Sun, Deep residual learning for image recognition, in: Proc. IEEE Conf. Comput. Vis. Pattern Recognit., CVPR, Jun., 2016, pp. 770–778.

[35] M. Sandler, A. Howard, M. Zhu, A. Zhmoginov, L. Chen, MobileNetV2: Inverted residuals and linear bottlenecks, in: Proc. IEEE Conf. Comput. Vis. Pattern Recognit., CVPR, 2018, pp. 4510–4520.

[36] G. Hinton, S. Roweis, Stochastic neighbor embedding, in: International Conference on Neural Information Processing Systems, 2002, pp. 857–864.