# Towards Automatic Visual Inspection: A Weakly Supervised Learning Method for Industrial Applicable Object Detection

## Abstract

Industrial visual inspection for product or defect detection is an essential part of the modern industry. With the recent progress of deep learning, advanced industrial object detectors can be built. However, deep learning method is known data-hungry; the processes of training data collection and annotation are labor-intensive and time-consuming. It is especially impractical in industrial scenarios to collect publicly available datasets due to the inherent diversity and privacy. In this paper, we explore automation of industrial visual inspection and propose a segmentation-aggregation framework to learn object detectors from weakly annotated visual data. The used minimum annotation is only image-level category labels without bounding boxes. The method is implemented and evaluated on collected insulator images and public VOC benchmarks to verify its effectiveness. The experiments show that our method can achieve high detection accuracy and can be applied in industry to achieve automatic visual inspection with minimum annotation cost.

*Keywords:* industrial automation, insulator detection, object detection, weakly supervised learning, deep learning

## 1. Introduction

Industrial automation combines intelligent algorithms with control equipments to provide automatic and productive performance. The ubiquitous sensors and devices in factories collect and generate huge volumes of data every day. However, the data has salient character of variety and variability [1]; thus generalized and flexible automation is needed to perform optimized controls. Complex strategies are usually adapted to analyze the sensor signals. And visual recognition technology is performing a key role in developing intelligent industrial applications.

In recent years, deep learning methods especially Convolutional Neural Networks (CNNs) have achieved great successes in many visual recognition tasks. With the help of deep learning, the latent information of industrial data can be further exploited and utilized. And advanced visual inspection systems can be built to achieve high-level detection accuracy. However, deep learning methods are compute-intensive. Even though FPGAs have been adopted to accelerate the inference of CNNs, it is still hard to meet the huge computational demands of model training in embedded environments.

With the advent of 5G communication technology, the compute-intensive deep learning models can be deployed on the cloud. And real-time massive data communication is enabled by the service provided by 5G with higher data rates, high reliability, and ultra-low latency. Deep learning based solutions for industrial visual inspection become feasible, as illustrated in Fig. 1. Generally, sensors and edge devices are low-power and compute-limited. They acquire image or video signals and perform only basic data preprocessing, e.g., color calibration and format conversion. Simple dedicated algorithms can be also embedded in edge devices. Then on the one hand, the processed data is sent to the cloud for real-time visual recognition. And control instructions are issued based on the detection results. On the other hand, the collect data is also uploaded in the background to the cloud for incremental model learning. The deep learning model can be configured to continuously learn features from the new coming data. The trained model will be automatically updated and deployed on the cloud to provide high-performance service. With minimum manual intervention, deep learning based automatic visual inspection is enabled by cloud computing and 5G communication technology.

In industrial visual inspection systems, the core step is to localize and recognize the target objects. Fig. 2 shows two typical scenarios: automatic content and package quality inspection on production lines (Fig. 2a) and defects detection in electrical insulators of transmission lines (Fig. 2b). In computer vision, object detection is a focused research field and acts as a basic task for high-level visual applications. It aims at recognizing all the objects in an image and mark their localizations and scales using bounding boxes [2]. To train a deep learning model, tens or hundreds of annotated images for each target object category are need as training data. There have been many

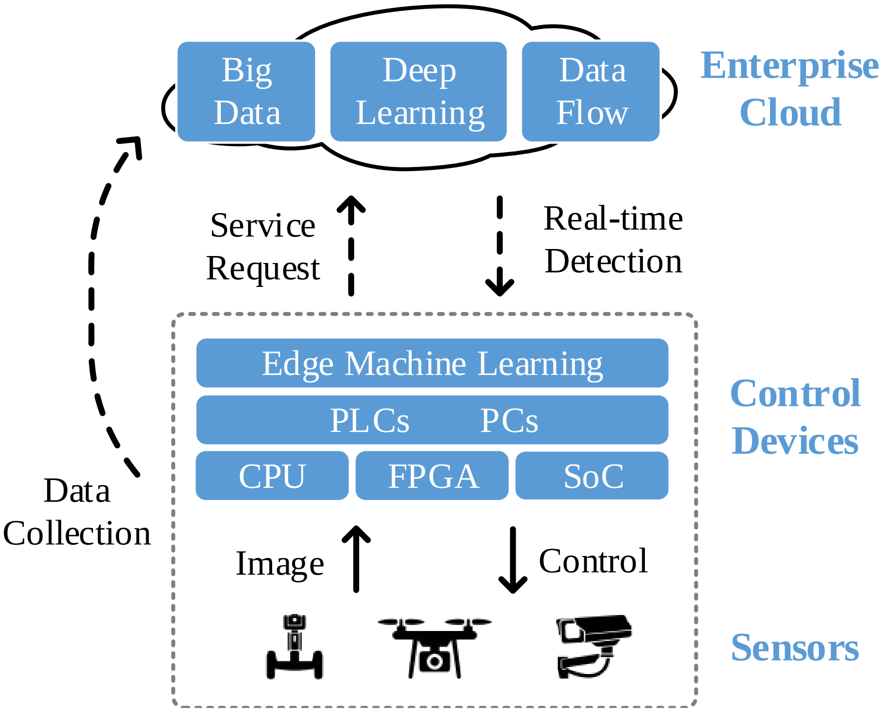


Figure 1: The architecture of cloud-based industrial recognition system.

well-organized and carefully-annotated public datasets [2–4] to facilitate research and production for general object detection. However, the scenarios of industrial inspection are special. Specific datasets need to be collected and annotated for different target applications. For example, in the case of package detection on production lines, the training images mainly contain various types of packaged products; while for insulators inspection, different views of insulators in transmission lines and substations should be captured. There is big gap between the two feature domains. The collected images for one task is useless for another. The tedious annotation process is time-consuming and labor-intensive, and it increases the burden on operators which is against the objective of industrial automation. Industrial automation should not only achieve automatic control, but also realize advanced technologies with minimum manual effort.

In this paper, we introduce weakly supervised learning (WSL) [5] to industrial automation. We aim at achieving automatic visual inspection [6, 7] of industrial ap-

(a) Product detection

(b) Insulator inspection

Figure 2: Examples of two different industrial inspection scenarios.

plicable object detectors from weakly annotated data. Specifically, only image-level labels are needed for deep learning model training without any bounding boxes; and the proposed method is still able to learn the locations and scales of target objects. The paradigm of weakly supervised learning greatly reduces the laborious annotation effort which makes it more practical and applicable in real-word industrial scenarios.

Over the last few years, a few weakly supervised detection methods have sprung up. But recently, researchers [8, 9] have noticed that most solutions encountered two types of failures: 1) surrounding a group of similar objects in one bounding box and 2) covering only parts of target objects. The first failure is caused by the lack of spatial information for each individual object instance. The second case is mainly due to the strong discriminative capacity of CNNs. In this paper, we propose a new segmentation-aggregation framework for weakly supervised object detection to deal with the two failures. The overall architecture of the proposed method is shown in Fig. 3. Firstly, a number of objectness proposals (i.e., hypotheses) are generated from the raw input image. Proposals of the image are then abstracted as a weighted undirected graph according to their overlapping relations. A spectral clustering method is used to split the proposal graph. And the input image is accordingly segmented into several subimages. After image segmentation, the shared CNN layers extract features for each subimage and assign classification scores to each projected Region of Interest (RoI). For score aggregation, we devise a dropscore strategy to partially aggregate the proposal-level (i.e., object-level) scores to subimage level. The scores of each subimage are pooled together as the final output. The main contributions of this paper are summarized as

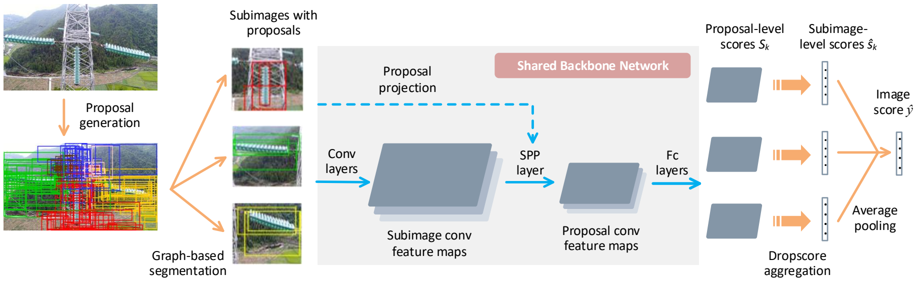


Figure 3: Overall architecture of the proposed segmentation-aggregation weakly supervised learning framework.

follows:

- Weakly supervised deep learning is introduced to industrial automation, and a new segmentation-aggregation framework is proposed to automatically learn object detectors from weakly annotated visual data.
- A graph-based image segmentation method and a novel dropscore regularization strategy are proposed to address the common failures in weakly supervised object detection.
- The proposed method is implemented and evaluated on collected insulator images and public benchmarks to verify its effectiveness which achieved superior results than other alternatives.

The rest of the paper is organized as follows: Section 2 gives a brief overview about industrial visual recognition and weakly supervised object detection. In section 3, the proposed learning framework is explained in detail. And in section 4, the experimental results are present, and ablation studies are conducted. Finally, we conclude our work in section 5.

## 2. Related Work

### *2.1. Industrial Visual Recognition*

Researchers have studied various industrial vision-based recognition problems, such as face detection [10, 11], part detection [12], fire detection [13], flame detection [14], smoke detection [15], and defect detection [16, 17]. Recently, deep CNNs are widely used to realize industrial object detectors. Yeh *et al.* [18] proposed a new moving object detection scheme based on hysteresis thresholding and motion compensation. Gao *et al.* [19] used deep CNNs to fuse RGB and LIDAR data to achieve high object detection accuracy in autonomous vehicle environment. Dias *et al.* [20] fine-tuned a pre-trained CNN to perform flower detection. Coulibaly *et al.* [21] used VGG-16 model and applied transfer learning to extract features of crop for disease detection.

### *2.2. Weakly Supervised Object Detection*

Generally, the goal of weakly supervised learning (WSL) [5] is to learn a function mapping $f : \mathcal{X} \mapsto \mathcal{Y}$ from data set $D = \{(X_1, y_1), (X_2, y_2), \dots, (X_i, y_i)\}$, where the input sample $X_i = \{x_{i1}, x_{i2}, \dots, x_{i,m_i}\} \in \mathcal{X}$ is called a *bag* of feature *instance* $x_{ij}$, and $\mathcal{Y}$ is coarse-grained ground truth. Weakly supervised object detection (WSOD) aims at realizing object detectors from image data without bounding-box annotation. Here the training examples $\mathcal{X}$ are images of objects, and $\mathcal{Y}$ are image-level category labels $\{0, 1, 2, \dots, C\}$. Despite the training ground truth $y_i \subset \mathcal{Y}$ is only coarse-grained image-level annotation, the objective is to predict fine-grained labels and bounding boxes in object level. So that $\mathcal{Y}$ is called weak supervision information [5].

It is demonstrated by Oquab *et al.* [22, 23] that CNNs are able to localize objects using only image-level supervision. Bilen *et al.* [8] designed a two-stream weakly supervised deep detection network named WSDDN. They designed two network branches to simultaneously perform classification and localization. Kantorov *et al.* [24] extended the Fast R-CNN[25] framework and proposed a context-aware method to realize weakly supervised object localization. More recently, Li *et al.* [26] proposed a two-step adaptation approach. They first transferred a pre-trained CNN model to performance multi-label classification, and then applied a multi-instance mining strategy to train the detector. Then Jie *et al.* [27] proposed a seed proposal discovery algorithm and a self-taught learning strategy to further improve the quality of the mined samples. These methods combined end-to-end CNN models with instance mining algorithms to achieve accurate object detection. Our study follows these work and aims at addressing the long-lived common issues.

## 3. Methodology

The architecture of the proposed framework is shown in Fig. 3. We firstly describe the two main stages: image segmentation and dropscore aggregation in Section 3.1 and Section 3.2 respectively. In Section 3.3, we explain the implementation details and the training procedure.

### *3.1. Image Segmentation*

In this section, we detail the image segmentation stage shown in the left part of Fig. 3. This module is mainly designed to address the grouping issue in weakly supervised object detection. In addition to simple segmentation, an outlier filtering algorithm is embedded to improve the quality. For clarity, we subdivide the module into three detailed steps. And the overall procedure is illustrated in Fig. 4.

#### *3.1.1. Graph-based Proposal Grouping*

A few methods [28–32] have been proposed to generate high-quality proposal candidates (i.e., bounding boxes) to indicate possible regions that contain objects. Such techniques are widely used in object detection methods. We use one of the state-of-the-art proposal generating methods EdgeBoxes [30] for our experiments. Given an image $I_i$, the proposals generated by EdgeBoxes are denoted as a set $V_i = \{v_1, v_2, ..., v_l\}$. There are usually thousands of proposals for each image; and they overlap each other to different degrees. Most existing methods learn end-to-end proposal classifiers and then select the top ones as detection results. However, they independently treat each proposal as a possible object. The relationship of proposals is not fully utilized. And the quality of generated proposals is usually not guaranteed. It is often the case that a proposal contains multiple objects. Since deep CNN classifiers rely only on statistical features to make prediction, they often encounter the problem of misidentifying a group of similar instances as a single top-scoring object. So the intuition is that if we take proposal relations into consideration, it is possible to identify the proposals that cover multiple objects and remove misleading ones.

Our solution is to decouple close objects into several subimages. During the procedure, we also filter out some unwanted proposals, such as those that cover more than one subimage. The idea of image segmentation also reduces the number of objects in each subimage. With fewer instances, it is easier to distinguish similar object individuals. Formally, we abstract the proposal relationship as a weighted undirected graph $G = (V, E)$, where each node $v \in V$ corresponds to a proposal and two nodes $v_p$ and $v_q$ are connected if the corresponding proposals overlap each other. The weight $w(p, q)$ of edge $e_{p,q} \in E$ is defined as the Intersection over Union (IoU) area of the proposals $v_p$ and $v_q$:

$$IoU(v_p, v_q) = \frac{area(v_p) \cap area(v_q)}{area(v_p) \cup area(v_q)}. \tag{1}$$

After building the proposals into a graph, we split it into several parts using the concept of graph cuts. In graph theory, the *cut* of two disjoint node sets $A$ and $B$ is defined as the weight sum of edges connecting the two node sets:

$$cut(A, B) = \sum_{u \in A, v \in B} w(u, v). \tag{2}$$

Due to the inherent weakness of cut definition, optimization algorithms of *minimum cut* produce biased partitions, i.e., isolated nodes versus a node group. This is unreasonable in object detection. Thus, we further adopt a normalized cut criterion [33] to balance the graph separation. The disassociation between two node sets $A$ and $B$ is measured by:

$$Ncut(A, B) = \frac{cut(A, B)}{asso(A, V)} + \frac{cut(A, B)}{asso(B, V)}. \tag{3}$$

The function $asso(\cdot)$ measures the association degree between two node sets: $asso(A, V)$ is defined as the sum of edge weights that connect from nodes in set $A$ to all the nodes in graph $G$; and $asso(B, V)$ is similarly defined. Graph separation can be iteratively performed by minimizing the Ncut objective. Given a number $m$, simultaneous partition can be implemented by means of spectral clustering [33, 34]. All the nodes in $V$ are then grouped into $m$ size-balanced clusters. The results of proposal grouping is illustrated in Fig. 4b. The corresponding proposals in each group are drawn using different colors and, for clarity, only a portion of proposals are visualized.

### *3.1.2. Outlier Proposal Filtering*

The proposals are separated into $m$ non-empty groups. In such a way, every proposal falls in one and only one group. It has been observed [8, 35] that some unwanted proposals may cause significant failures, such as proposals that cover multiple object instances or that involve background regions. These types of proposals are still reserved after proposal grouping. But is is clear that most good proposals in each group densely cover around the extent of object, while the unwanted proposals are deviating and sparsely distributed. We implement bad proposal filtering process as an outlier detection problem [36, 37]. A simple outlier detection method is developed based on proposal overlap densities.

In each proposal group, a *mean overlap density* is calculated for every proposal. The mean overlap density of proposal $v_p$ is defined as the average of overlap degrees of other proposals within its coverage:

$$mod_p = \frac{1}{h_p \cdot w_p} \sum_{j=1}^{h_p} \sum_{i=1}^{w_p} d_{ij}, \tag{4}$$

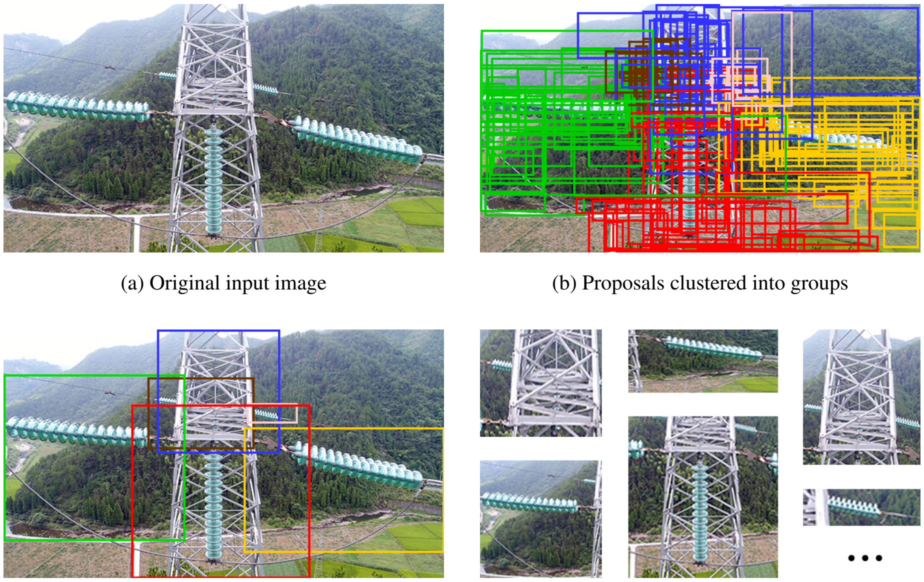
where $w_p$ and $h_p$ is the width and height of proposal $v_p$ (in pixel) and $d_{ij}$ is the number of overlapped proposals at point $(i, j)$. The mean overlap density is further normalized as the relative overlap density:

$$rod_p = \frac{mod_p}{N_p + 1}, \tag{5}$$

where $N_p$ is the total number of proposals that overlaps $v_p$ (plus one for $v_p$ itself). The denominator $N_p + 1$ is interpreted as the possible maximum overlap density of $v_p$, when all the $N_p$ proposals are the same size as $v_p$ and are completely aligned. This relative overlap density measures the degree of proposal concentration. A value close to one indicates that the proposals properly overlap each other. And a value closer to zero means they are scattered. In our experiments, we empirically remove proposals with relative overlap density below a certain value.

### *3.1.3. Subimage generation*

After filtering out poor proposals, the proposals in each group approximately surround the true content of foreground objects. In each proposal group, we take the minimum coordinates $(x_{\min}, y_{\min})$ of all top left corners and the maximum coordinates $(x_{\max}, y_{\max})$ of all bottom right corners. The corresponding rectangular areas (i.e., pixel regions) along these coordinates are cut out. In this manner, subgraphs are



(a) Original input image

(b) Proposals clustered into groups

(c) Rectified region of each group

(d) Segmented subimages

Figure 4: Illustration of the proposed image segmentation procedure. (Better viewed in color and different colors indicate different proposal groups)

converted to subimages as illustrated in Fig. 4c and Fig. 4d. And during implementation, the coordinates of proposals are accordingly transformed from the whole image to each subimage coordinate system.

### *3.2. Dropscore Aggregation*

Assume proposals are separated into $m$ groups $\{P_1, P_2, \ldots, P_m\}$. The shared backbone network extract feature for each subimage. And proposals are projected to convolutional feature maps. Through fully connected layers, each proposal is encoded to a $C$-dim score vector, where $C$ is the number of output classes. For proposal group $P_k$ $(1 \leq k \leq m)$, the network produces a score matrix $S_k \in \mathbb{R}^{C \times |P_k|}$ $(1 \leq k \leq m)$. They are depicted as gray parallelograms in the right part of Fig. 3. To define loss function, the final output should be a single $C$-dim vector to match the training label. Proposal-level score matrices are usually squeezed by summing all the scores over the $P_k$ dimension [8]. However, this is the main cause of focusing on discriminative parts.

We instead propose a dropscore strategy to reduce the overfitting effect. The dropscore strategy is mainly inspired by DropBlock [38]. Similar techniques are used in [39, 40]. DropBlock is a regularization method for convolutional networks. To some extent, it can be considered as an improved version of dropout [41]. It is designed to drop continuous activations in feature maps to remove semantic information and thus enforce CNNs to learn more other features. And in [40], pixel-level features in the last score maps are partial removed to regularize the learning of pointwise object segmentation.

We devise dropscore strategy to regularize the score aggregation by dynamically removing a subset of proposals. For subimage $k$, the scores of class $c$ is aggregated by summing portions of proposals:

$$\hat{s}_k^c = \sum_{i \in \mathcal{P}^+} S_k^{(i,c)} + \alpha \sum_{j \in \mathcal{P}^-} S_k^{(j,c)}, \tag{6}$$

where $\mathcal{P}^+$ and $\mathcal{P}^-$ are the reserved positive and negative proposals. They are proposals with high and low scores respectively. The rest medium-scoring proposals are dropped; and they do not contribute to the gradient backpropagation. The weighting factor $\alpha$ trade off the importances of the two parts. The positive term $\mathcal{P}^+$ is important for correct category classification, while the negative term $\mathcal{P}^-$ helps to distinguish foreground objects from backgrounds.

The $C$-dim squeezed scores $\hat{s}_k$ $(1 \leq k \leq m)$ are then average-pooled together over the $m$ subimages for the final prediction:

$$\hat{y}^c = \underset{k \in \{1,\ldots,m\}}{\text{mean}} \hat{s}_k^c, \ \ c \in \{1, 2, \ldots, C\}. \tag{7}$$

The average pooling operation ensures that the training error is back-propagated to all the subimages to refine CNN weights. The learning objective is to minimize the loss function defined as the sum of $C$ binary logistic losses:

$$\mathcal{L}_{det} = -\sum_{i=1}^{n} \sum_{c=1}^{C} \big( y_i^c \log \sigma(\hat{y}_i^c) + (1 - y_i^c) \log(1 - \sigma(\hat{y}_i^c)) \big), \tag{8}$$

where the sigmoid function $\sigma(x) = 1/(1 + e^{-x})$ scales the scores into real-valued probability range (0, 1).

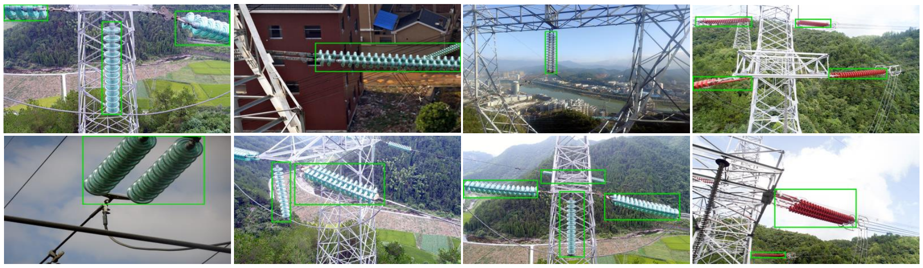


Figure 5: Examples of insulator detection on power transmission lines.

### 3.3. Implementation Details

As our design is flexible, the proposed segmentation-aggregation framework can be integrated with many different backbone networks. In this work, we adopt the popular WSDDN [8] model as backbone. Specifically, VGG-16 [42] net pre-trained on ImageNet are used. The convolutional layers before pooling layer `pool5` are preserved as feature extractor. The max-pooling layer is replaced by SPP layer [25, 43] to project input proposals. And the subsequent fully connected layers are changed to two branches as in WSDDN [8].

We train the model using a multi-segmentation multi-scale feature extraction strategy. During each iteration, the input image is segmented into $m$ subimages by randomly select a number according to different datasets. They are rescaled such that the length of their longest sides is chosen from $\{480, 576, 688, 864, 1200\}$. The proportion of retained positive and negative proposals are both set to 30%. For testing, proposal scores of five image scales and all subimage segmentations are averaged.

## 4. Experiments

### 4.1. Datasets and Metrics

Since there are no benchmark datasets for industrial object detection, we use the insulator images provided by State Grid Corporation for experiments. There are 60 high-resolution original images with complicated background and different number of insulators. We augmented them by cropping, rotating, and horizontal/vertical flipping to a total of 420 images. We randomly select 400 images for training and 20 images for testing. The detected bounding boxes with confident scores above the threshold are kept, and the non-maximum suppression (NMS) [44] algorithm is used to reduce redundancy.

To quantitatively evaluate the performance, we also test the proposed method on public PASCAL VOC2007 [45] dataset. The VOC2007 dataset is the widely used benchmark for visual recognition tasks especially for weakly supervised object detection. It consists of 2,501 training images, 2,510 validation images and 4,952 test images for 20 object categories. As suggested, the *trainval* set (the union of training set and validation set) is used for training. Only image-level class label annotations are used for supervision. We report the results on two metrics: AP and CorLoc. The Average Precision (AP) and the mean of AP (mAP) are standard evaluation protocols for PASCAL VOC object detection challenges [3]. And the Correct Localization (CorLoc) metric is defined to measure the localization accuracy [46].

### *4.2. Experimental Setup*

Before training, the proposals whose longest side is smaller than 20 pixels are empirically removed as in [47]. We randomly select 2k proposals per image (from what EdgeBoxes algorithm generated) for model training. The weights of newly added neural network layers are randomly initialized with Gaussian distribution. The model is firstly fine-tuned on the target datasets without bounding boxes for domain adaptation. The initial learning rate of the convolutional layers and the fully connected layers are set to $10^{-4}$ and $10^{-3}$ respectively. They exponentially decrease every 10 epochs with a decay rate of 0.1 for 30 epochs in total. The loss function is optimized with momentum of 0.9 and weight decay of 0.0005.

### *4.3. Results on Insulator Images*

The detection examples of insulators images are shown in Fig. 5. The proposed model is robust to localize insulators of different views and different colors. Even with complex backgrounds, the majority of the salient insulators are detected. It is notable that the model was trained without bounding-box annotation, but it automatically leaned to localize the true content of insulators. For images that contain multiple

Table 1: Detection average precision (AP)(%) of our method on VOC2007 *test* set and comparison with other state of the arts.

| method | aero | bike | bird | boat | bottle | bus | car | cat | chair | cow | table | dog | horse | mbike | person | plant | sheep | sofa | train | tv | mAP |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Cinbis *et al.* [48] | 38.1 | 47.6 | 28.2 | 13.9 | 13.2 | 45.2 | 48.0 | 19.3 | 17.1 | 27.7 | 17.3 | 19.0 | 30.1 | 45.4 | 13.5 | 17.0 | 28.8 | 24.8 | 38.2 | 15.0 | 27.4 |
| Bilen *et al.* [8] | 39.4 | 50.1 | 31.5 | 16.3 | 12.6 | 64.5 | 42.8 | 42.6 | 10.1 | 35.7 | 24.9 | 38.2 | 34.4 | 55.6 | 9.4 | 14.7 | 30.2 | 40.7 | 54.7 | 46.9 | 34.8 |
| Kantorov *et al.* [24] | 57.1 | 52.0 | 31.5 | 7.6 | 11.5 | 55.0 | 53.1 | 34.1 | 1.7 | 33.1 | 49.2 | 42.0 | 47.3 | 56.6 | 15.3 | 12.8 | 24.8 | 48.9 | 44.4 | 47.8 | 36.3 |
| Li *et al.* [26] | 54.5 | 47.4 | 41.3 | 20.8 | 17.7 | 51.9 | 63.5 | 46.1 | 21.8 | 57.1 | 22.1 | 34.4 | 50.5 | 61.8 | 16.2 | 29.9 | 40.7 | 15.9 | 55.3 | 40.2 | 39.5 |
| Jie *et al.* [27] | 52.2 | 47.1 | 35.0 | 26.7 | 15.4 | 61.3 | 66.0 | 54.3 | 3.0 | 53.6 | 24.7 | 43.6 | 48.4 | 65.8 | 6.6 | 18.8 | 51.9 | 43.6 | 53.6 | 62.4 | 41.7 |
| Tang *et al.* [49] | 58.0 | 62.4 | 31.1 | 19.4 | 13.0 | 65.1 | 62.2 | 28.4 | 24.8 | 44.7 | 30.6 | 25.3 | 37.8 | 65.5 | 15.7 | 24.1 | 41.7 | 46.9 | 64.3 | 62.6 | 41.2 |
| Ours | 49.1 | 53.6 | 43.5 | 21.3 | 18.5 | 66.9 | 64.0 | 55.6 | 11.9 | 53.7 | 26.6 | 45.6 | 48.7 | 64.6 | 20.4 | 23.3 | 50.0 | 44.7 | 55.9 | 60.6 | 43.9 |

Table 2: Correct Localization (CorLoc)(%) of our method on VOC2007 *trainval* set and comparison with other state of the arts.

| method | aero | bike | bird | boat | bottle | bus | car | cat | chair | cow | table | dog | horse | mbike | person | plant | sheep | sofa | train | tv | mean |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Cinbis *et al.* [48] | 57.2 | 62.2 | 50.9 | 37.9 | 23.9 | 64.8 | 74.4 | 24.8 | 29.7 | 64.1 | 40.8 | 37.3 | 55.6 | 68.1 | 25.5 | 38.5 | 65.2 | 35.8 | 56.6 | 33.5 | 47.3 |
| Bilen *et al.* [8] | 65.1 | 58.8 | 58.5 | 33.1 | 39.8 | 68.3 | 60.2 | 59.6 | 34.8 | 64.5 | 30.5 | 43.0 | 56.8 | 82.4 | 25.5 | 41.6 | 61.5 | 55.9 | 65.9 | 63.7 | 53.5 |
| Kantorov *et al.* [24] | 83.3 | 68.6 | 54.7 | 23.4 | 18.3 | 73.6 | 74.1 | 54.1 | 8.6 | 65.1 | 47.1 | 59.5 | 67.0 | 83.5 | 35.3 | 39.9 | 67.0 | 49.7 | 63.5 | 65.2 | 55.1 |
| Li *et al.* [26] | 78.2 | 67.1 | 61.8 | 38.1 | 36.1 | 61.8 | 78.8 | 55.2 | 28.5 | 68.8 | 18.5 | 49.2 | 64.1 | 73.5 | 21.4 | 47.4 | 64.6 | 22.3 | 60.9 | 52.3 | 52.4 |
| Jie *et al.* [27] | 72.7 | 55.3 | 53.0 | 27.8 | 35.2 | 68.6 | 81.9 | 60.7 | 11.6 | 71.6 | 29.7 | 54.3 | 64.3 | 88.2 | 22.2 | 53.7 | 72.2 | 52.6 | 68.9 | 75.5 | 56.1 |
| Tang *et al.* [49] | 81.7 | 80.4 | 48.7 | 49.5 | 32.8 | 81.7 | 85.4 | 40.1 | 40.6 | 79.5 | 35.7 | 33.7 | 60.5 | 88.8 | 21.8 | 57.9 | 76.3 | 59.9 | 75.3 | 81.4 | 60.6 |
| Ours | 78.3 | 74.6 | 62.8 | 42.9 | 36.7 | 71.5 | 84.1 | 66.7 | 20.4 | 79.3 | 34.8 | 59.3 | 66.7 | 87.1 | 35.3 | 54.4 | 75.6 | 52.9 | 66.4 | 74.3 | 61.2 |

objects, like the first and fourth images in the first row, the model detected all the insulators despite various orientations. But there are still several notable false cases. The model tends to enclose the close insulators in one bounding box, for example, the first image in the second row. And some small-size insulators are missed or incorrectly localized due to the complex background. But overall, the model works well for the most primary insulators. The missed detections could be solved by using multi-view images.

### *4.4. Results on VOC2007 Dataset*

We evaluate the model on VOC2007 dataset and compare it with other recent state-of-the-art weakly supervised methods. The results are reported in Table 1 and Table 2. They are all single models and based on the same VGG-16 backbone thus to be comparable. The proposed method achieves overall better results than other alternatives. Without image segmentation and dropscore strategy, our model is nearly the same as

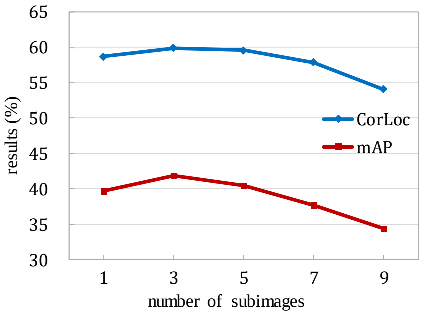


Figure 6: Results on VOC 2007 for different number of subimages.

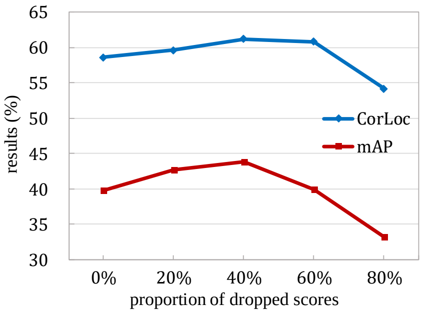


Figure 7: Results on VOC 2007 for different proportion of dropped scores.

the WSDDN model of Bilen *et al.* [8]. Compared with this baseline, we gain more than 9% improvements of mAP. The APs of animals like cat and dog are overall improved. This is mainly attributed to the proposed dropscore strategy. The issue of focusing on discriminative parts is highly mitigated. Considerable improvements are also obtained on small-size objects like the category of bird.

While our method show strong performance in most classes, it does not perform very well in categories of aeroplane and chair. This is explained by the fact that their shapes and outlines are quite complex. For example, the front view of aircraft wings is too thin, which makes it hard to identify them as parts of a foreground object. Some images of chair and plant are also incorrectly enclosed due to their unclear boundaries. Compare the results of AP and CorLoc, it is notable that the localization results of these categories are still acceptable. The CorLoc of other classes are similar to the pattern of their APs. We thus infer that the proposed method is able to find the locations of most objects, but the accurate contents of certain categories are not properly learned. Better detection performance generally means strong localization capacity since the CorLoc metric measures only single object localization performance. In contrast, it is difficult to simultaneously improve the multi-object detection performance across all categories.

### *4.5. Ablation Studies*

#### *4.5.1. Analysis of the number of subimages*

The main hyperparameter in our method is the number of subimages. The input images are segmented to $m$ parts based on graph cut. The outlier filtering step works as a rectifier to refine the quality of image segmentation. To some extent, the number $m$ basically determines the quality of subsequent computation.

We experiment with different number of subimages to study its influence. The results are shown in Fig. 6. We fix the proportion of dropped scores to 40%, and segment the input images into $m$ subimages of 1, 3, 5, 7, 9 for five experiments respectively. As seen from the chart, the mAP and CorLoc increase when using more than one subimages. The highest mAP and CorLoc are obtained when $m$ is set to 3 to 5. The results instead decline when using more subimages, where mAP drops a little quickly. This is due to the limited number of proposals. As proposals are divided into more groups, there are fewer positive proposals in each subimage for effective classification. Moreover, too many subimages may cause oversegmentation. With simple statistics, the majority of images in VOC dataset contain less than four object classes. There are very few images contain more than eight objects. We finally adopted a hybrid multi-segmentation training approach, because it not only preserves the completeness of object instances, but also augments the training data.

#### *4.5.2. Analysis of the dropsocre strategy*

The dropsocre aggregation strategy is designed to avoid the problem of focusing on object parts. In object discrimination, the top-scoring proposals are important for correct classification, while the low-scoring ones are much helpful in determining the object scale. With dynamic selection, the attention to proposal-level features is regularized. We analyze the influence of different proportion of dropped scores in Fig. 7. The horizontal axis presents the dropped medium-scoring proposals; and the rest proposals are evenly divided for the positive proposals $\mathcal{P}^+$ and the negative proposals $\mathcal{P}^-$. There is a upward trend for both mAP and CorLoc as the proportion of dropped scores increases. And the values dropped sharply when more than half of scores are discarded.

Dropping an appropriate number of proposals penalizes the unbalanced score distribution, but overremoving will prevent the model from converging.

## 5. Conclusion

We explored learning automation of visual detection in industrial scenarios. A proposal-guided segmentation-aggregation learning framework was proposed to realize object detectors from weakly annotated images. The graph-based segmentation method and dropscore regularization strategy addressed the persistent issues in existing weakly supervised detection models. Experimental results showed that our design is efficient and flexible for industrial application. We demonstrated that weakly supervised learning based automatic visual inspection has great potential in industrial automation as it requires little manual annotation effort.

## References

[1] M. Mohammadi, A. I. Al-Fuqaha, S. Sorour, M. Guizani, Deep learning for iot big data and streaming analytics: A survey, CoRR abs/1712.04301.

[2] O. Russakovsky, J. Deng, H. Su, J. Krause, S. Satheesh, S. Ma, Z. Huang, A. Karpathy, A. Khosla, M. S. Bernstein, A. C. Berg, L. Fei-Fei, Imagenet large scale visual recognition challenge, Int. J. Comput. Vis. 115 (2015) 211–252.

[3] M. Everingham, L. Van Gool, C. K. I. Williams, J. Winn, A. Zisserman, The pascal visual object classes (voc) challenge, Int. J. Comput. Vis. 88 (2) (2010) 303–338.

[4] T. Lin, M. Maire, S. J. Belongie, J. Hays, P. Perona, D. Ramanan, P. Dollár, C. L. Zitnick, Microsoft COCO: common objects in context, in: Proc. ECCV, Vol. 8693, 2014, pp. 740–755.

[5] Z.-H. Zhou, A brief introduction to weakly supervised learning, Natl. Sci. Rev. 5 (1) (2017) 44–53.

[6] C. Mera, M. Orozco-Alzate, J. Branch, D. Mery, Automatic visual inspection: An approach with multi-instance learning, Comput. Ind. 83 (2016) 46–54.

[7] S. Huang, Y. Pan, Automated visual inspection in the semiconductor industry: A survey, Comput. Ind. 66 (2015) 1–10.

[8] H. Bilen, A. Vedaldi, Weakly supervised deep detection networks, in: Proc. IEEE CVPR, 2016, pp. 2846–2854.

[9] C. Ge, J. Wang, Q. Qi, H. Sun, J. Liao, Fewer is more: Image segmentation based weakly supervised object detection with partial aggregation, in: Proc. BMVC, 2018, p. 136.

[10] S. Jin, D. Kim, T. T. Nguyen, D. Kim, M. Kim, J. W. Jeon, Design and implementation of a pipelined datapath for high-speed face detection using fpga, IEEE Trans. Ind. Informat. 8 (1) (2012) 158–167.

[11] Q. Huang, C. K. Jia, X. Zhang, Y. Ye, Learning discriminative subspace models for weakly supervised face detection, IEEE Trans. Ind. Informat. 13 (6) (2017) 2956–2964.

[12] C. Cusano, P. Napoletano, Visual recognition of aircraft mechanical parts for smart maintenance, Comput. Ind. 86 (2017) 26–33.

[13] K. Muhammad, S. Khan, M. Elhoseny, S. Hassan Ahmed, S. Wook Baik, Efficient fire detection for uncertain surveillance environment, IEEE Trans. Ind. Informat. 15 (5) (2019) 3113–3122.

[14] Z. Li, L. S. Mihaylova, O. Isupova, L. Rossi, Autonomous flame detection in videos with a dirichlet process gaussian mixture color model, IEEE Trans. Ind. Informat. 14 (3) (2018) 1146–1154.

[15] A. Filonenko, D. C. Hernández, K. Jo, Fast smoke detection for video surveillance using CUDA, IEEE Trans. Ind. Informat. 14 (2) (2018) 725–733.

[16] L. Wang, Z. Zhang, Automatic detection of wind turbine blade surface cracks based on uav-taken images, IEEE Trans. Ind. Electron. 64 (9) (2017) 7293–7303.

[17] C. Mera, M. Orozco-Alzate, J. Branch, Incremental learning of concept drift in multiple instance learning for industrial visual inspection, Comput. Ind. 109 (2019) 153–164.

[18] C. Yeh, C. Lin, K. Muchtar, H. Lai, M. Sun, Three-pronged compensation and hysteresis thresholding for moving object detection in real-time video surveillance, IEEE Trans. Ind. Electron. 64 (6) (2017) 4945–4955.

[19] H. Gao, B. Cheng, J. Wang, K. Li, J. Zhao, D. Li, Object classification using cnn-based fusion of vision and lidar in autonomous vehicle environment, IEEE Trans. Ind. Informat. 14 (9) (2018) 4224–4231.

[20] P. A. Dias, A. Tabb, H. Medeiros, Apple flower detection using deep convolutional networks, Comput. Ind. 99 (2018) 17–28.

[21] S. Coulibaly, B. Kamsu-Foguem, D. Kamissoko, D. Traore, Deep neural networks with transfer learning in millet crop images, Comput. Ind. 108 (2019) 115–120.

[22] M. Oquab, L. Bottou, I. Laptev, J. Sivic, Learning and transferring mid-level image representations using convolutional neural networks, in: Proc. IEEE CVPR, 2014, pp. 1717–1724.

[23] M. Oquab, L. Bottou, I. Laptev, J. Sivic, Is object localization for free? - weakly-supervised learning with convolutional neural networks, in: Proc. IEEE CVPR, 2015, pp. 685–694.

[24] V. Kantorov, M. Oquab, M. Cho, I. Laptev, Contextlocnet: Context-aware deep network models for weakly supervised localization, in: Proc. ECCV, Vol. 9909, 2016, pp. 350–365.

[25] R. Girshick, Fast r-cnn, in: Proc. IEEE ICCV, 2015, pp. 1440–1448.

[26] D. Li, J. B. Huang, Y. Li, S. Wang, M. H. Yang, Weakly supervised object localization with progressive domain adaptation, in: Proc. IEEE CVPR, 2016, pp. 3512–3520.

[27] Z. Jie, Y. Wei, X. Jin, J. Feng, W. Liu, Deep self-taught learning for weakly supervised object localization, in: Proc. IEEE CVPR, 2017, pp. 4294–4302.

[28] B. Alexe, T. Deselaers, V. Ferrari, Measuring the objectness of image windows, IEEE Trans. Pattern Anal. Mach. Intell. 34 (11) (2012) 2189–2202.

[29] J. R. R. Uijlings, K. E. A. van de Sande, T. Gevers, A. W. M. Smeulders, Selective saerch for object recognition, Int. J. Comput. Vis. 104 (2013) 154–171.

[30] C. L. Zitnick, P. Dollár, Edge boxes: Locating object proposals from edges, in: Proc. ECCV, Vol. 8693, 2014, pp. 391–405.

[31] P. ArbelÃ¡ez, J. Pont-Tuset, J. Barron, F. Marques, J. Malik, Multiscale combinatorial grouping, in: Proc. IEEE CVPR, 2014, pp. 328–335.

[32] M. M. Cheng, Z. Zhang, W. Y. Lin, P. Torr, Bing: Binarized normed gradients for objectness estimation at 300fps, in: Proc. IEEE CVPR, 2014, pp. 3286–3293.

[33] J. Shi, J. Malik, Normalized cuts and image segmentation, IEEE Trans. Pattern Anal. Mach. Intell. 22 (8) (2000) 888–905.

[34] Z. Zhang, F. Xing, H. Wang, Y. Yan, Y. Huang, X. Shi, L. Yang, Revisiting graph construction for fast image segmentation, Pattern Recognit. 78 (2018) 344 – 357.

[35] B. Lai, X. Gong, Saliency guided end-to-end learning for weakly supervised object detection, in: Proc. IJCAI, 2017, pp. 2053–2059.

[36] M. M. Breunig, H.-P. Kriegel, R. T. Ng, J. Sander, LOF: identifying density-based local outliers, in: Proc. ACM SIGMOD, 2000.

[37] E. Schubert, A. Zimek, H.-P. Kriegel, Local outlier detection reconsidered: a generalized view on locality with applications to spatial, video, and network outlier detection, Data Min. Knowl. Discov. 28 (1) (2012) 190–237.

[38] G. Ghiasi, T. Lin, Q. V. Le, Dropblock: A regularization method for convolutional networks, in: Proc. NIPS, 2018, pp. 10750–10760.

[39] T. Durand, N. Thome, M. Cord, Weldon: Weakly supervised learning of deep convolutional neural networks, in: Proc. IEEE CVPR, 2016, pp. 4743–4752.

[40] T. Durand, T. Mordan, N. Thome, M. Cord, Wildcat: Weakly supervised learning of deep convnets for image classification, pointwise localization and segmentation, in: Proc. IEEE CVPR, 2017, pp. 5957–5966.

[41] G. E. Hinton, N. Srivastava, A. Krizhevsky, I. Sutskever, R. Salakhutdinov, Improving neural networks by preventing co-adaptation of feature detectors, CoRR abs/1207.0580.

[42] K. Simonyan, A. Zisserman, Very deep convolutional networks for large-scale image recognition, CoRR abs/1409.1556.

[43] K. He, X. Zhang, S. Ren, J. Sun, Spatial pyramid pooling in deep convolutional networks for visual recognition, IEEE Trans. Pattern Anal. Mach. Intell. 37 (9) (2015) 1904–1916.

[44] A. Neubeck, L. Van Gool, Efficient non-maximum suppression, in: Proc. IEEE ICPR, Vol. 3, 2006, pp. 850–855.

[45] M. Everingham, L. Van Gool, C. K. I. Williams, J. Winn, A. Zisserman, The PASCAL Visual Object Classes Challenge 2007 (VOC2007) Results, `http://www.pascal-network.org/challenges/VOC/voc2007/workshop/index.html`.

[46] T. Deselaers, B. Alexe, V. Ferrari, Weakly supervised localization and learning with generic knowledge, Int. J. Comput. Vis. 100 (3) (2012) 275–293.

[47] Y. Wei, W. Xia, M. Lin, J. Huang, B. Ni, J. Dong, Y. Zhao, S. Yan, Hcp: A flexible cnn framework for multi-label image classification, IEEE Trans. Pattern Anal. Mach. Intell. 38 (9) (2016) 1901–1907.

[48] R. G. Cinbis, J. Verbeek, C. Schmid, Weakly supervised object localization with multi-fold multiple instance learning, IEEE Trans. Pattern Anal. Mach. Intell. 39 (1) (2017) 189–203.

[49] P. Tang, X. Wang, X. Bai, W. Liu, Multiple instance detection network with online instance classifier refinement, in: Proc. IEEE CVPR, 2017, pp. 3059–3067.