Learning From Weakly Supervised Data by The Expectation Loss SVM (e-SVM) Algorithm

Jun Zhu, Junhua Mao, Alan Yuille
Department of Statistics
University of California, Los Angeles
{jzh@,mjhustc@,yuille@stat.}ucla.edu

Abstract

In many situations we have some measurement of confidence on "positiveness" for a binary label. The "positiveness" is a continuous value whose range is a bounded interval. It quantifies the affiliation of each training data to the positive class. We propose a novel learning algorithm called *expectation loss SVM* (e-SVM) that is devoted to the problems where only the "positiveness" instead of a binary label of each training sample is available. Our e-SVM algorithm can also be readily extended to learn segment classifiers under weak supervision where the exact positiveness value of each training example is unobserved. In experiments, we show that the e-SVM algorithm can effectively address the segment proposal classification task under both strong supervision (e.g. the pixel-level annotations are available) and the weak supervision (e.g. only bounding-box annotations are available), and outperforms the alternative approaches. Besides, we further validate this method on two major tasks of computer vision: semantic segmentation and object detection. Our method achieves the state-of-the-art object detection performance on PASCAL VOC 2007 dataset.

1 Introduction

Recent work in computer vision relies heavily on manually labeled datasets to achieve satisfactory performance. However, the detailed hand-labelling of datasets is expensive and impractical for large datasets such as ImageNet [6]. It is better to have learning algorithms that can work with data that has only been weakly labelled, for example by putting a bounding box around an object instead of segmenting it or parsing it into parts.

In this paper we present a learning algorithm called expectation loss SVM (e-SVM). It requires a method that can generate a set of proposals for the true class label (e.g., the exact silhouette of the object). But this set of proposals may be very large, each proposal tends to be only partially correct (the correctness can be quantified by a continues value between 0 and 1 called "positiveness"), and several proposals may be required to obtain the correct label. In the training stage, our algorithm can deal with the strong supervised case where the positiveness of the proposals is observed, and can easily extend to the weakly supervised case by treating the positiveness as latent variable. In the testing stage, it predicts the class label for each proposal and provides a confidence score.

There are some alternative approaches for this problem, such as support vector classification (SVC), support vector regression (SVR), and logistic regression (LR). For the SVC algorithm, because this is not a standard binary classification problem, one might need to binarize the positiveness using ad-hoc heuristics to determine a threshold, which may degrade its performance [19]. To address this problem, previous works usually use SVR [4, 19] to train the class confidence scoring model in semantic segmentation task. We compare our e-SVM to these three related methods in the segment proposal's class confidence prediction problem. The positiveness of each segment proposal is set as the *intersection over union* (IoU) overlap rate between this proposal and the pixel-level instance

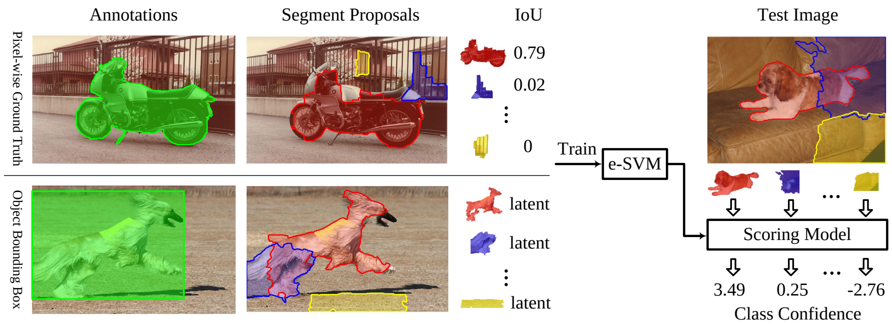

Figure 1: Illustration of the framework on class confidence prediction of segment proposals. In training, our e-SVM algorithm can handle two different annotation types: pixel-level (strong supervision) and bounding-box (weak supervision) annotations. For pixel-level annotation, we set the positiveness of segment proposals as the IoU overlap rate w.r.t. ground truth and train scoring models with basic e-SVM. For bounding-box annotation, we treat the positiveness as latent variable and use the latent e-SVM version for training scoring models. In testing stage, the learned scoring model can predict confidence scores of segment proposals for each object class. (Best viewed in color)

ground truth. We test our algorithm under two different data annotation scenarios: the pixel-level annotation (positiveness is observed) and the bounding-box annotation (positiveness is unobserved). The experimental results show that our e-SVM outperforms SVC, SVR, and LR in both scenarios. Figure 1 illustrates the framework on class confidence prediction of segment proposals.

We further validate our approach on two fundamental computer vision tasks: (i) semantic segmentation, and (ii) object detection. Firstly, we consider semantic segmentation. There has recently been impressive progress at this task using rich appearance cues. Segments are extracted from images [1, 3, 4, 12], appearance features are computed for each segment [5, 22, 26], and classifiers are trained using groundtruth pixel labeling [19]. Methods of this type are almost always among the winners of the PASCAL VOC segmentation challenge [5]. But all these methods rely on datasets which have been hand-labelled at the pixel level. For this application we generate the segment proposals using CPMC segments [4]. The positiveness of each proposal is set as the IoU overlap rate. The class confidence scoring models learnt by our e-SVM, using either the pixel-level or bounding-box annotation, can obtain comparable semantic segmentation accuracy w.r.t. the state-of-the-art learning algorithm used in semantic segmentation literature.

Secondly, we address object detection by exploiting the effectiveness of segments' appearance cues and coupling them to existing object detection systems. For this application, the data is only weakly labeled because the ground-truth annotation for object detection is typically specified by bounding boxes (e.g. PASCAL VOC [8, 9] and ImageNet [6]), which means that the pixel-level ground truth is not available. We also use the CPMC method to produce segment proposals. The IoU w.r.t. object instance bounding boxes is used to represent the positiveness of the proposals. We test our approach on the PASCAL VOC dataset using, as the base detector, the regions with CNN (RCNN) features [14] (currently the state of the art on PASCAL VOC and outperforms previous works by a large margin). This method first used selective search method [25] to extract candidate bounding boxes. For each candidate bounding box, it extracts features by deep networks [17] learned on Imagenet dataset and fine-tuned on PASCAL. We couple the segment-based appearance cues to this system by simply concatenating a new segment confidence map feature based on the learned e-SVM models and the deep learning feature, and then train a linear SVM. We show that this simple approach yields a performance gain of 1.5 percent on per-class mean average precision (mAP) over the state-of-the-art RCNN feature on PASCAL VOC 2007 dataset.

Note that this approach is general. It can use any segment proposal detectors, any image features, and any classifiers. When applied to object detection it could use any base detector, and we could couple the appearance cues with the base detector in many different ways (we choose the simplest). In addition, it can handle other problems where only the "positiveness" instead of binary labels are available in training.

μ_i.

2 Related work on weakly supervised learning and weighted SVMs

We have introduced some of the most relevant works in semantic segmentation or object detection. In this section, we will briefly review related work of weakly supervised learning methods for segment classification, and discuss the connection to instance weighted SVM approaches in literature.

The problem settings for most previous works generally assumed that they only get a set of accompanying words of an image or a set of image-level labeling, which is different from the problem settings in this paper. The multiple instance learning (MIL) algorithms [7, 2] were adopted to solve these problems [21, 23]. MIL handles the situations where at least one positive instance is presented in the positive bag and only the bag labels of training examples are available. Vezhnevets *et.al.* [27] proposed a multi-image model (MIM) to solve this weakly-supervised learning problem. Recently, Liu *et.al.* [20] presented a weakly-supervised dual clustering approach to handle this task.

Our weakly supervised problem setting is in the middle between these settings and the strong supervision case (i.e. the full pixel-level annotations are available). It is also very important and useful because bounding-box annotations of large-scale image dataset are already available (e.g. ImageNet [6]) while the pixel-level annotations of large datasets are still hard to obtain. This weakly supervised problem cannot be solved by MIL. We cannot assume that at least one "completely" positive instance (i.e. a CPMC segment proposal) is present in a positive bag (i.e. a groundtruth object instance) since most of the proposals contain both foreground and background pixels. We will show how our e-SVM and its latent extension address this problem in the next sections.

In machine learning literature, the weighted SVM (WSVM) approaches [24, 28, 18] also use an instance-dependent weight on the cost of each example, and can improve the robustness of model estimation [24], alleviate the effect of outliers [28], leverage privileged information [18] or deal with unbalanced classification problems. The difference between our e-SVM and WSVMs mainly lies in that it weights class labels instead of data points, which leads to each example contributing both to the costs of positive and negative labels. Although the loss function of our e-SVM algorithm is different from those of WSVMs, it can be effortlessly solved by any standard SVM solver (e.g., LibLinear [10]) like those used in WSVMs. This is an advantage because it does not require a specific solver for the implementation of our e-SVM.

3 The expectation loss SVM algorithms

In this section, we will first describe the basic formulation of our expectation loss SVM algorithm in section 3.1 when the positiveness of each segment proposal is observed. Then, in section 3.2, a latent e-SVM is introduced to handle the weak supervision situation where the positiveness of each segment proposal is not observed.

3.1 The basic e-SVM algorithm

We are given a set of training images $\mathcal{D}$. Using some segmentation method (we adopt CPMC [4] in this work), we can generate a set of foreground segment proposals $\{S_1, S_2, \ldots, S_N\}$ from these images. For each segment S_i, we extract feature $\mathbf{x}_i$, $\mathbf{x}_i \in \mathbb{R}^d$.

Suppose the pixelwise annotations are available for all the groundtruth instances in $\mathcal{D}$. For each object class, we can calculate the IoU ratio u_i ($u_i \in [0, 1]$) between each segment S_i and the groundtruth instances labeling, and set the positiveness of S_i as u_i (although positiveness can be some functions of IoU ratio, for simplicity, we just set it as IoU and use u_i to represent the positiveness in the following paragraphs). Because many foreground segments overlap partially with the groundtruth instances (i.e. $0 < u_i < 1$), it is not a standard binary classification problem for training. Of course, we can define a threshold τ_b and treat all the segments whose $u_i \geq \tau_b$ as positive examples and the segments whose $u_i < \tau_b$ as negative examples. In this way, this problem is transferred to a Support Vector Classification (SVC) problem. But it needs some heuristics to determine τ_b and its performance is only partially satisfactory [19].

To address this issue, we proposed our expectation loss SVM model as an extension of the classical SVC models. In this model, we treat the label Y_i of each segment as an unobserved random variable. $Y_i \in \{-1, +1\}$. Given $\mathbf{x}_i$, we assume that Y_i follows a Bernoulli distribution. The probability of $Y_i = 1$ given $\mathbf{x}_i$ (i.e. the success probability of the Bernoulli distribution) is denoted as μ_i. We assume that μ_i is a function of the positiveness u_i, i.e. $\mu_i = g(u_i)$. In the experiment, we simply set $\mu_i = u_i$.

Similar to the traditional linear SVC problem, we adopt a linear function as the prediction function: $F(\mathbf{x}_i) = \mathbf{w}^T\mathbf{x}_i + b$. For simplicity, we denote $[\mathbf{w}\ b]$ as $\mathbf{w}$, $[\mathbf{x}_i\ 1]$ as $\mathbf{x}_i$ and $F(\mathbf{x}_i) = \mathbf{w}^T\mathbf{x}_i$ in the remaining part of the paper. The loss function of our e-SVM is the expectation over the random variables Y_i:

$$\begin{aligned}
\mathcal{L}(\mathbf{w}) =& \lambda_{\mathbf{w}} \cdot \frac{1}{2}\mathbf{w}^T\mathbf{w} + \frac{1}{N}\sum_{i=1}^{N}\mathbb{E}_{Y_i}[\max(0, 1 - Y_i\mathbf{w}^T\mathbf{x}_i)] \\
=& \lambda_{\mathbf{w}} \cdot \frac{1}{2}\mathbf{w}^T\mathbf{w} + \frac{1}{N}\sum_{i=1}^{N}[l_i^+ \cdot \Pr(Y_i = +1|\mathbf{x}_i) + l_i^- \cdot \Pr(Y_i = -1|\mathbf{x}_i)] \\
=& \lambda_{\mathbf{w}} \cdot \frac{1}{2}\mathbf{w}^T\mathbf{w} + \frac{1}{N}\sum_{i=1}^{N}\{l_i^+ \cdot g(u_i) + l_i^- \cdot [1 - g(u_i)]\}
\end{aligned} \tag{1}$$

where $l_i^+ = \max(0, 1 - \mathbf{w}^T\mathbf{x}_i)$ and $l_i^- = \max(0, 1 + \mathbf{w}^T\mathbf{x}_i)$.

Given the pixelwise groundtruth annotations, $g(u_i)$ is known. From Equation 1, we can see that it is equivalent to "weight" each sample with a function of its positiveness. The standard linear SVM solver is used to solve this model with loss function of $\mathcal{L}(\mathbf{w})$. In the experiments, we show that the performance of our e-SVM is much better than SVC and slightly better than Support Vector Regression (SVR) in the segment classification task.

3.2 The latent e-SVM algorithm

One of the advantage of our e-SVM model is that we can easily extend it to the situation where only bounding box annotations are available (this type of labeling is of most interest in the paper). Under this weakly supervised setting, we cannot obtain the exact value of the positiveness (i.e., IoU) u_i for each segment. Instead, u_i will be treated as a latent variable which will be determined by minimizing the following loss function:

$$\mathcal{L}(\mathbf{w}, \mathbf{u}) = \lambda_{\mathbf{w}} \cdot \frac{1}{2}\mathbf{w}^T\mathbf{w} + \frac{1}{N}\sum_{i=1}^{N}\{l_i^+ \cdot g(u_i) + l_i^- \cdot [1 - g(u_i)]\} + \lambda_R \cdot R(\mathbf{u}) \tag{2}$$

where $\mathbf{u}$ denotes $\{u_i\}_{i=1,\ldots,N}$. $R(\mathbf{u})$ is a regularization term for $\mathbf{u}$. We can see that the loss function in Equation 1 is a special case of that in Equation 2 by setting $\mathbf{u}$ as constant and λ_R equal to 0.

When $\mathbf{u}$ is fixed, $\mathcal{L}(\mathbf{w}, \mathbf{u})$ is a standard linear SVM loss, which is convex with respect to $\mathbf{w}$. When $\mathbf{w}$ is fixed, $\mathcal{L}(\mathbf{w}, \mathbf{u})$ is also a convex function if $R(\mathbf{u})$ is a convex function with respect to $\mathbf{u}$. The IoU between a segment S_i and groundtruth bounding boxes, denoted as u_i^{bb}, can serve as an initialization for u_i. We can iteratively fix $\mathbf{u}$ and $\mathbf{w}$, and solve the two convex optimization problems until it converges. The pseudo-code for the optimization algorithm is shown in Algorithm 1.

If we do not add any regularization term on $\mathbf{u}$ (i.e. set $\lambda_R = 0$), $\mathbf{u}$ will become 0 or 1 in the optimization step in line 4 of algorithm 1 because the loss function becomes a linear function with respect to $\mathbf{u}$ when $\mathbf{w}$ is fixed. It turns to be similar to a latent SVM and can lead the algorithm to stuck in the local minimal as shown in the experiments. The regularization term will prevent this situation under the assumption that the true value of $\mathbf{u}$ should be around $\mathbf{u}^{bb}$.

There are a lot of different designs of the regularization term $R(\mathbf{u})$. In practice, we use the following one based on the cross entropy between two Bernoulli distributions with success probability u_i^{bb} and

Algorithm 1 The optimization algorithm for training latent e-SVM

Initialization:

1: $\mathbf{u}^{(cur)} \leftarrow \mathbf{u}^{bb}$;

Process:

2: **repeat**

3: $\quad \mathbf{w}^{(new)} \leftarrow \arg\min_{\mathbf{w}} \mathcal{L}(\mathbf{w}, \mathbf{u}^{(cur)})$;

4: $\quad \mathbf{u}^{(new)} \leftarrow \arg\min_{\mathbf{u}} \mathcal{L}(\mathbf{w}^{(new)}, \mathbf{u})$;

5: $\quad \mathbf{u}^{(cur)} \leftarrow \mathbf{u}^{(new)}$;

6: **until** Converge

u_i respectively.

$$\begin{aligned} R(\mathbf{u}) &= -\frac{1}{N}\sum_{i=1}^{N}[u_i^{bb}\cdot\log(u_i) + (1-u_i^{bb})\cdot\log(1-u_i)] \\ &= -\frac{1}{N}\sum_{i=1}^{N} D_{KL}[\text{Bern}(u_i^{bb})||\text{Bern}(u_i)] + C \end{aligned} \tag{3}$$

where C is a constant value with respect to $\mathbf{u}$. $D_{KL}(.)$ represents the KL distance between two Bernoulli distributions. This regularization term is convex w.r.t. $\mathbf{u}$ and achieves its minimal when $\mathbf{u} = \mathbf{u}^{bb}$. It is a strong regularization term since its value increases very fast when $\mathbf{u} \neq \mathbf{u}^{bb}$.

4 Visual Tasks

4.1 Semantic segmentation

We can easily apply our e-SVM to the semantic segmentation task with the framework proposed by Carreira et al. [5]. Firstly, CPMC segment proposals [4] are generated and the second-order pooling features [5] are extracted from each segment. Then we train the class confidence scoring models using either e-SVM or latent e-SVM according to whether the pixel-level annotation is available. In the testing stage, the CPMC segments are sorted based on their confidence scores. The top ones will be selected to produce the semantic label map.

4.2 Object detection

For the task of object detection, we can only acquire bounding-box annotation instead of pixel-level labeling. Therefore, it is natural to apply our latent e-SVM in this task to provide segment-based appearance cues for object detection.

In the state-of-the-art object detection systems [11, 13, 25, 14], the window candidates of foreground object are extracted from images and the confidence scores are predicted on them. Window candidates are extracted either by sliding window approaches (used in e.g. the deformable part-based model [11, 13]) or most recently, the selective search approach [25] (used in e.g. the RCNN framework [14]). It can lower down the number of window candidates compared to traditional sliding window approaches.

It is not easy to directly incorporate confidence scores of the segments into these object detection systems based on window candidates. The difficulty lies in two aspects: First, only some of the segments are totally inside or totally outside a window candidate. It might be hard to calculate the contribution of the confidence score of a segment that only partially overlaps with a window candidate. Second, the window candidates (even the groundtruth bounding boxes) may contain some of the background regions. Some regions (e.g. the regions near the boundary of the window candidates) will have higher probability to be the background region than the regions in the center. Treating them equally may harm the accuracy of the whole detection system.

In order to solve these issues, we propose a new segment confidence map feature (SCMF) for each candidate window. Given an image and a set of window candidates, we first calculate the

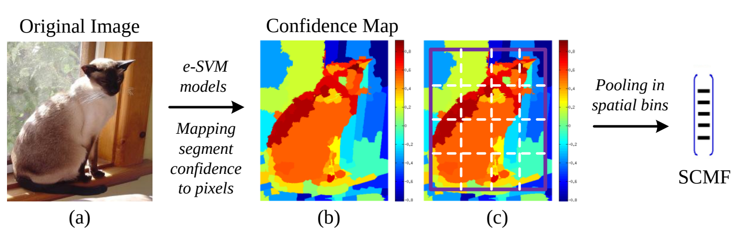

Figure 2: Illustration of generating the segment confidence map feature for window candidates based on learned e-SVM models. The confidence scores of the segments are mapped to pixels to generate a pixel-level confidence map. We will divide a window candidate into $m \times m$ spatial bins and pool the confidence scores of the pixels in each bin. It leads to a $m \times m$ dimensional vector for our SCMF.

confidence scores of all the segment proposals in the image using the learned e-SVM models. The confidence score for a segment S is denoted as $\mathrm{CfdScore}(S)$. For each pixel p, the confidence score is set as the maximum confidence score of all the segments that contain this pixel: $\mathrm{CfdScore}(p) = \max_{\forall S, p \in S} \mathrm{CfdScore}(S)$. In this way, we can handle the difficulty of partial overlapping between segments and candidate windows. For the second difficulty, we divide each candidate window into $M = m \times m$ spatial bins and pool the confidence scores of the pixels in each bin. Because the models are trained with the one-vs-rest scheme, our SCMF is class-specific. It leads to a $(M \times K)$-dimensional feature for each candidate window, where K refers to the total number of object classes. After that, we encode it by additive kernels approximation mapping [26] and obtain the final feature representation of candidate windows. The feature generating process is illustrated in Figure 2. In the testing stage, we can concatenate this SCMF with the features from other object detection systems.

5 Experiments

In this section, we first evaluate the performance of e-SVM on the segment proposal's class confidence prediction problem, by using two new evaluation criterions for this task. After that, we apply our method to two essential tasks in computer vision: semantic segmentation and object detection. For semantic segmentation task, we test the proposed e-SVM and latent e-SVM on two different data annotation scenarios (i.e., with pixel-level groundtruth label annotation and with only bounding-box object annotation) respectively. For object detection task, we combine our SCMF with the state-of-the-art object detection system, and show it can obtain non-trivial improvement on detection performance.

5.1 Performance evaluation

We use PASCAL VOC 2011 [9] segmentation dataset in this experiment. It is a subset of the whole PASCAL 2011 dataset with 1112 images in the training set and 1111 images in the validation set, and has 20 foreground object classes in total. We use the official training set and validation set for training and testing respectively. Similar to [5], we extract 150 CPMC [4] segment proposals for each image and compute the second-order pooling features on each segment.

5.1.1 Evaluation criteria

In literature [5], the supervised learning framework of segment-based prediction model either regressed the overlapping value or converted it to a binary classification problem via a threshold value, and evaluate the performance by certain task-specific criterion (i.e., the pixel-wise accuracy used for semantic segmentation). In this paper, we adopt a direct performance evaluation criteria for class confidence prediction of segment proposals, which is consistent with the learning problem itself and not biased to particular tasks. Unfortunately, we have not found any work on this sort of direct performance evaluation, and thus introduce two new evaluation criteria for this purpose. We briefly describe them as follows:

Mean precision recall volume
Although the ground-truth target value (i.e., the overlap rate of segment and bounding box) is a real value in the range of [0, 1], we can transform original class confidence prediction problem to a series of binary classification problems, each of which corresponds to a threshold value for binarizing the groundtruth overlap rate of segments. After that, we calculate the Precison Recall (PR) Curve for each of these binary classification problems, and it forms a PR surface w.r.t. different threshold values. Thus, we can compute the volume under this PR surface as in [15], and use the mean PR volume (mPRV) over all classes as a performance metric for the segment-based class confidence prediction problem.

Normalized discounted cumulative gain [16]
Considering that a higher confidence value is expected to be predicted for the segment with higher overlap rate, we think this prediction problem can be treated as a ranking problem, and thus we use the normalized discounted cumulative gain (NDCG) [16], which is a common performance measurement for ranking problem, as another performance evaluation criterion in this paper.

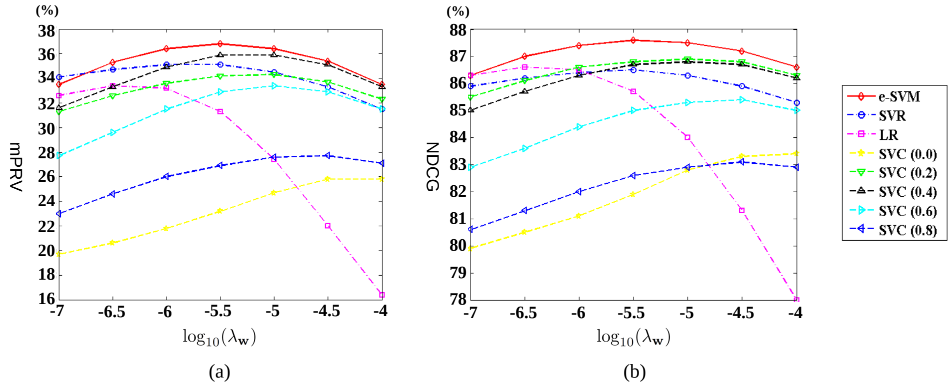

Figure 3: Comparison on class confidence prediction results of e-SVM, SVR, LR and SVCs (using pixel-level groundtruth annotation). (a) mPRV, (b) NDCG. Best viewed in color.

	e-SVM	SVR	LR	SVM (0.0)	SVM (0.2)	SVM (0.4)	SVM (0.6)	SVM (0.8)
mPRV (%)	**36.8**	35.1	33.4	25.8	34.3	35.9	33.4	27.7
NDCG (%)	**87.6**	86.5	86.6	83.4	86.9	86.8	85.4	83.1

Table 1: Results on class confidence prediction of segment proposals (using pixel-level groundtruth annotation). The number in bracket refers to the threshold value of overlap rate for training SVC.

	Le-SVM	SVR	LR	SVM (0.0)	SVM (0.2)	SVM (0.4)	SVM (0.6)	SVM (0.8)
mPRV (%)	**31.9**	29.9	29.0	24.4	30.7	30.4	24.8	14.7
NDCG (%)	**85.8**	84.7	84.8	82.6	85.5	84.9	82.2	76.6

Table 2: Results on class confidence prediction of segment proposals (using object bounding-box annotation). The number in bracket refers to the threshold value of overlap rate for training SVC. “Le-SVM” refers to the latent e-SVM algorithm.

5.1.2 Experimental results and comparison to other methods

Based on the mPRV and NDCG introduced above, We evaluate the performance of our e-SVM algorithm on PASCAL VOC 2011 segmentation dataset, and compare it with three classic methods (i.e. SVC, SVR and LR) in literature. Note that we test SVCs’ performance with a variety of binary classification problems, each of which is trained by using different threshold values of overlap rates (e.g., 0.0, 0.2, 0.4, 0.6 and 0.8 as shown in figure 3) to get positive and negative examples. In figure 3 (a) and (b), we show the results of mPRV and NDCG for e-SVM, SVR, LR and SVCs respectively. We evaluate the performance w.r.t. different values of $\lambda_{\mathbf{w}}$ as shown in figure 3. In addition, we compare their results[1] trained with pixel-wise ground truth and weakly-labelled bounding-box annotation (The latent e-SVM is used in this case.) in tables 1 and 2 respectively. Our e-SVM obtains consistently superior mPRV and NDCG than other methods in both of these two annotation types.

5.2 Semantic segmentation results

For the semantic segmentation task, we test our method with PASCAL VOC 2011 dataset using training set for training and validation set for testing. Following the framework proposed by [5], we use the sequential pasting inference approach in testing stage. The per-class accuracies w.r.t. the groundtruth pixel-level semantic label map and object bounding-box annotation are 36.8% and 27.7% respectively, which are comparable to those of the state-of-the-art class confidence scoring model learning algorithm (i.e., SVR) [5] used in semantic segmentation literature.

[1]We report the best performance w.r.t. different $\lambda_{\mathbf{w}}$ values for each method in tables 1 and 2.

	plane	bike	bird	boat	bottle	bus	car	cat	chair	cow	
RCNN	**64.1**	69.2	50.4	41.2	33.2	62.8	70.5	61.8	32.4	58.4	
Ours	63.7	**70.2**	**51.9**	**42.5**	**33.4**	**63.2**	**71.3**	**62.0**	**34.7**	**58.7**	
Gain	-0.4	1.0	1.5	1.3	0.2	0.4	0.8	0.2	2.3	0.2	
RCNN (bb)	68.1	72.8	56.8	43.0	36.8	66.3	74.2	67.6	34.4	63.5	
Ours (bb)	**70.4**	**74.2**	**59.1**	**44.7**	**38.0**	**67.2**	**74.6**	**69.0**	**36.7**	**64.3**	
Gain	2.3	1.4	2.3	1.6	1.2	1.0	0.3	1.3	2.3	0.8	
	table	dog	horse	motor.	person	plant	sheep	sofa	train	tv	Average
RCNN	45.8	55.8	61.0	66.8	53.9	30.9	53.3	49.2	56.9	64.1	54.1
Ours	**47.8**	**57.9**	**61.2**	**67.5**	**54.9**	**34.5**	**55.8**	**51.0**	**58.4**	**65.0**	**55.3**
Gain	2.0	2.1	0.3	0.8	1.0	3.7	2.5	1.8	1.6	0.9	1.2
RCNN (bb)	54.5	61.2	69.1	68.6	58.7	33.4	62.9	51.1	62.5	64.8	58.5
Ours (bb)	**56.4**	**62.9**	**69.3**	**69.9**	**59.6**	**35.6**	**64.6**	**53.2**	**64.3**	**65.5**	**60.0**
Gain (bb)	1.9	1.8	0.2	1.4	0.9	2.2	1.7	2.1	1.8	0.7	1.5

Table 3: Object detection results on PASCAL VOC 2007 dataset. "bb" means the result after applying bounding box regression. Gain means the improved AP of our system compared to RCNN under the same settings (both with bounding box or without). The better results in the comparisons are shown in bold.

5.3 Object detection results

As mentioned in Section 4.2, another application of our e-SVM is the object detection task. Most recently, Girshick et.al [14] presented a Regions with CNN features method (RCNN) using the Convolutional Neural Network pre-trained on the ImageNet Dataset [6] and fine-tuned on the PASCAL VOC datasets. They achieved a significantly improvement over the previous state-of-the-art algorithms (e.g. Deformable Part-based Model (DPM) [11])and push the detection performance into a very high level (The mAP is 58.5 with bounding-box regression on PASCAL VOC 2007).

One question arises: can we further improve their performance? The answer is yes. In our method, we first learn the latent e-SVM models based on the object bounding-box annotation, and calculate the spatial confidence map features as in section 4.2. Then we simply concatenate them with RCNN the features to train object classifiers on candidate windows. We use PASCAL VOC 2007 dataset in this experiment. As shown in table 3, our method can improve mAP by 1.2 before applying bounding boxes regression. For some categories that the original RCNN does not perform well, such as potted plant, the gain of mAP is up to 3.65. After applying bounding box regression for both RCNN and our algorithm, the gain of performance is 1.5 on average.

In the experiment, we set $m = 5$ and adopt average pooling on the pixel level confidence scores within each spatial bin. We also modified the bounding box regularization method used in [14] by augmenting the fifth layer features with additive kernels approximation methods [26]. It will lead to a slightly improved performance. In summary, we achieved an average AP of 60.0, which is 1.5 higher than the best known result (the original RCNN with bounding box regression) of this dataset.

6 Conclusion

We present a novel learning algorithm call e-SVM that can well handle the situation in which the labels of training data are continuous values whose range is a bounded interval. It can be applied to segment proposal's class confidence prediction problem and can be easily extended to learn the class confidence scoring models under weak supervision (e.g. only bounding-box annotation is available). We apply this method on two major tasks of computer vision (i.e., semantic segmentation and object detection), and obtain the state-of-the-art object detection performance on PASCAL VOC 2007 dataset. We believe that, with the ever growing size of datesets, it is increasingly important for learning with weak supervision to reduce the amount of labeling overhead required.

Acknowledgements. We gratefully acknowledge the funding support from the National Science Foundation (NSF) with award CCF-1317376, and from the National Institute of Health NIH Grant 5R01EY022247-03. We also thank NVIDIA Corporation for providing GPUs in our experiments.

References

[1] R. Achanta, A. Shaji, K. Smith, A. Lucchi, P. Fua, and S. Susstrunk. SLIC superpixels compared to state-of-the-art superpixel methods. *TPAMI*, 34(11):2274–2282, 2012.

[2] S. Andrews, I. Tsochantaridis, and T. Hofmann. Support vector machines for multiple-instance learning. In *Advances in Neural Information Processing Systems 15*, pages 561–568. MIT Press, 2003.

[3] P. Arbelaez, B. Hariharan, C. Gu, S. Gupta, and J. Malik. Semantic segmentation using regions and parts. In *CVPR*, 2012.

[4] J. Carreira and C. Sminchisescu. Cpmc: Automatic object segmentation using constrained parametric min-cuts. *TPAMI*, 34(7):1312–1328, 2012.

[5] J. a. Carreira, R. Caseiro, J. Batista, and C. Sminchisescu. Semantic segmentation with second-order pooling. In *ECCV*, pages 430–443, 2012.

[6] J. Deng, A. Berg, , J. Winn, and A. Zisserman. The PASCAL Visual Object Classes Challenge 2010 (VOC2010) Results. http://www.image-net.org/challenges/LSVRC/2012/index.

[7] T. G. Dietterich, R. H. Lathrop, and T. Lozano-Pérez. Solving the multiple instance problem with axis-parallel rectangles. *Artif. Intell.*, 89(1-2):31–71, Jan. 1997.

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

[9] M. Everingham, L. Van Gool, C. K. I. Williams, J. Winn, and A. Zisserman. The PASCAL Visual Object Classes Challenge 2011 (VOC2011) Results. http://www.pascal-network.org/challenges/VOC/voc2011/workshop/index.html.

[10] R.-E. Fan, K.-W. Chang, C.-J. Hsieh, X.-R. Wang, and C.-J. Lin. LIBLINEAR: A library for large linear classification. *JMLR*, 9:1871–1874, 2008.

[11] P. F. Felzenszwalb, R. B. Girshick, D. McAllester, and D. Ramanan. Object detection with discriminatively trained part-based models. *TPAMI*, 32(9):1627–1645, 2010.

[12] P. F. Felzenszwalb and D. P. Huttenlocher. Efficient graph-based image segmentation. *IJCV*, 59(2):167–181, Sept. 2004.

[13] S. Fidler, R. Mottaghi, A. L. Yuille, and R. Urtasun. Bottom-up segmentation for top-down detection. In *CVPR*, pages 3294–3301, 2013.

[14] R. Girshick, J. Donahue, T. Darrell, and J. Malik. Rich feature hierarchies for accurate object detection and semantic segmentation. In *CVPR*, 2014.

[15] B. Hariharan, P. Arbelaez, R. Girshick, and J. Malik. Simultaneous detection and segmentation. In *ECCV*, 2014.

[16] K. Järvelin and J. Kekäläinen. Cumulated gain-based evaluation of ir techniques. *TOIS*, 20(4):422–446, 2002.

[17] A. Krizhevsky, I. Sutskever, and G. E. Hinton. Imagenet classification with deep convolutional neural networks. In *NIPS*, pages 1106–1114, 2012.

[18] M. Lapin, M. Hein, and B. Schiele. Learning using privileged information: Svm+ and weighted svm. *Neural Networks*, 53:95–108, 2014.

[19] F. Li, J. Carreira, and C. Sminchisescu. Object recognition as ranking holistic figure-ground hypotheses. In *CVPR*, pages 1712–1719, 2010.

[20] Y. Liu, J. Liu, Z. Li, J. Tang, and H. Lu. Weakly-supervised dual clustering for image semantic segmentation. In *Computer Vision and Pattern Recognition (CVPR), 2013 IEEE Conference on*, pages 2075–2082. IEEE, 2013.

[21] A. Müller and S. Behnke. Multi-instance methods for partially supervised image segmentation. In *PSL*, pages 110–119, 2012.

[22] X. Ren, L. Bo, and D. Fox. Rgb-(d) scene labeling: Features and algorithms. In *CVPR*, June 2012.

[23] J. Shotton, M. Johnson, and R. Cipolla. Semantic texton forests for image categorization and segmentation. In *CVPR*, pages 1–8, 2008.

[24] J. Suykens, J. D. Brabanter, L. Lukas, and J. Vandewalle. Weighted least squares support vector machines: robustness and sparse approximation. *NEUROCOMPUTING*, 48:85–105, 2002.

[25] J. Uijlings, K. van de Sande, T. Gevers, and A. Smeulders. Selective search for object recognition. *IJCV*, 104(2):154–171, 2013.

[26] A. Vedaldi and A. Zisserman. Efficient additive kernels via explicit feature maps. *TPAMI*, 34(3):480–492, 2012.

[27] A. Vezhnevets, V. Ferrari, and J. Buhmann. Weakly supervised semantic segmentation with a multi image model. In *ICCV*, 2011.

[28] X. Yang, Q. Song, and A. Cao. Weighted support vector machine for data classification. In *IJCNN*, 2005.