Relaxed Multiple-Instance SVM with Application to Object Discovery

Xinggang Wang†, Zhuotun Zhu†, Cong Yao, Xiang Bai*
School of Electronic Information and Communications
Huazhong University of Science and Technology
xgwang@hust.edu.cn, zhuzhuotun@hust.edu.cn, yaocong2010@gmail.com, xbai@hust.edu.cn

Abstract

Multiple-instance learning (MIL) has served as an important tool for a wide range of vision applications, for instance, image classification, object detection, and visual tracking. In this paper, we propose a novel method to solve the classical MIL problem, named relaxed multiple-instance SVM (RMI-SVM). We treat the positiveness of instance as a continuous variable, use Noisy-OR model to enforce the MIL constraints, and jointly optimize the bag label and instance label in a unified framework. The optimization problem can be efficiently solved using stochastic gradient decent. The extensive experiments demonstrate that RMI-SVM consistently achieves superior performance on various benchmarks for MIL. Moreover, we simply applied RMI-SVM to a challenging vision task, common object discovery. The state-of-the-art results of object discovery on Pascal VOC datasets further confirm the advantages of the proposed method.

1. Introduction

Exploring big visual data is a new trend in computer vision in recent years [29, 9, 5]. Especially, with the development of deep learning, the performances of many large-scale visual recognition tasks have been significantly improved. However, the supervised deep learning methods, *e.g.*, deep convolutional neural networks (DCNN) [18], rely heavily on the huge number of human-annotated data that are non-trivial to get. Finely labeled images/videos, which have pixel-level labels and bounding-box labels, are very limited and expensive. However, there are hundreds times of weakly labeled visual data that have image-level labels or noisy labels. For example, we can extract image label from its text caption on Flickr [15]. How to use the weakly labeled visual data for object recognition is a quite important research problem.

† equal contribution; * corresponding author.

Figure 1. Iteratively discover the locations of objects using the proposed RMI-SVM algorithm. 1st row: The top 100 object proposals detected by Edgebox [38]. 2nd row: Randomly initialized object locations in iteration 0. 3rd - 6th rows: The detected object locations in iteration 100, 500, 1000, and 2000, respectively. The blue boxes show the object proposals, the red boxes show the detected objects that do not enough overlap with ground-truth, and the green boxes show the detected objects that own enough overlap with ground-truth. (Best viewed in color.)

The multiple-instance learning (MIL), proposed by Dietterich *et al.* [11] for the purpose of drug activity prediction, is a popular tool for exploring sematic information in weakly labeled visual data. In MIL, instead of being given the labels of each individual instance, the learner receives a set of labeled bags, each containing plenty of instances. In the binary-classification task, a bag may be labeled as positive if *at least* one instance is positive. On the other hand, a bag will be labeled as negative if *none* of the instances is positive. Typically, we can regard an image/video as a

bag, and a patch/cube inside as an instance. Objects of interest are considered as positive instances, and the rest are considered as negative instances. Besides of learning bag distribution, we expect MIL can infer the label of instance to find objects of interest. However, not all MIL algorithms can reach this goal; most of them only focus on bag classification [11, 36, 32].

Selecting positive instances and learning a discriminative/generative instance model to classify bag is a popular way for solving MIL problem in computer vision. For example, online multiple-instance Boosting was applied for robust visual tracking in [3]; multiple instance SVM [1] was used to learn deformable object detector [16], which is also called latent SVM; and, unsupervised multiple instance Boosting was developed for multi-class learning in [37]. However, these existing methods all treat instance selection and model learning as two separated procedures, and use EM-style algorithm for optimization. In this paper, we propose a unified framework to jointly optimize the label of instance and learn instance model by taking the advantage of relaxing the discrete instance label and stochastic gradient descent. The MIL constraints are formulated using a Noisy-OR model. The instance model is a simple linear SVM model which allows fast training and prediction. The optimization problem can be efficiently solved using stochastic gradient descend algorithm, and is very robust to initialization in practical applications.

As shown in Fig. 1, the proposed MIL algorithm can be applied to object discovery, which is also called weakly-supervised object location and object co-localization. At first, we obtain hundreds of object proposals using the Edgebox [38] and extract the deep feature for each proposal using DCNN [18] in each image. Then, The proposed RMI-SVM algorithm is able to gradually find the true object location from the initialization location which is randomly selected. In the procedure of training RMI-MIL, we get exact object locations; besides, the learned instance model (object model) can be even used for object detection in unseen images. Our object discovery method is clean, simple but effective. It uses the off-shelf Edgebox object proposals and DCNN features. After feature extraction is done, it takes about 35 minutes using a single CPU to discover all the 20 classes in the Pascal VOC 2007 dataset. In the experiments, RMI-SVM shows superior performance when compared to both other MIL algorithms and the state-of-the-art object discovery methods.

To summarize, our main contributions are three folds: 1) a novel MIL formulation that relaxes the MIL constraints into convex program; 2) a fast and robust MIL solution via SGD; 3) an effective weakly-supervised object discovery based on the proposed RMI-SVM, which can obtain the state-of-the-art performance on the challenging Pascal 2007 dataset.

2. Related Work

Multiple-instance learning was firstly proposed by Dietterich *et al.* [11] for drug activity prediction. After that, since it is very useful in both machine learning and computer vision, lots of MIL algorithms have been proposed. Some of the typical methods are briefly introduced as follows: The diverse density (DD) method [21] tackles MIL by finding regions in the instance space with instances from many different positive bags and few instances from negative bags. In [35], DD is refined using expectation maximization (EM). In DD-SVM [7], instance prototype is extracted based on DD function in the instance feature space, followed by a nonlinear mapping to project each bag to a point in the bag feature space. miSVM and MILBoost were proposed in [1] and [34] in which they train SVM and boosting classifier for instances respectively. Recent work on MIL includes: representing the bags as graphs and explicitly modeling the relationships between instances within a bag in [36], studying the problem if there are infinite number of instances in a bag in [2], mining key instances from a citer kNN graph for bag classification [20], building a deep learning framework in a weakly supervised setting [33], and using bag-of-word model to solve large-scale MIL problem [32].

MIL is highly related to and plays an important role in many visual recognition tasks, especially in weakly-supervised object discovery, for example, person head discovery [34], object part discovery [12, 16], object class discovery [37]. For generic object discovery in the wild, MIL also works very well. A generative and convex MIL algorithm was proposed in [31] for object discovery based salient object detection. Very recently, MIL is trained on the top of DCNN to discover object for automatically image captioning [15].

Object discovery has recently drawn lots of attentions. Top-down segmentation priors based object detector is combined for pixel-level object discovery in [5]. A part-based matching between object proposals is proposed for unsupervised object discovery in [8]. A multi-fold MIL is designed for object discovery in [9]. And, a joint box-image formulation is proposed in [29] and applied for large-scale object discovery on the ImageNet dataset. Different from the existing object discovery methods, our object discovery method utilizes the proposed novel RMI-SVM, Edgebox and off-the-shelf DCNN feature to construct an end-to-end system, in which all the components are very efficient and effective.

3. Relaxed Multiple-Instance SVM

3.1. MIL Relaxations

We first give notation of MIL as preliminaries. In MIL, we are given a set bags $X = \{X_1, \ldots, X_n\}$; each bag is consisted with a set of instance $X_i = \{\mathbf{x}_{i1}, \ldots, \mathbf{x}_{im_i}\}$,

where m_i denotes the number of instances in the bag X_i; and each instance is represented by a d-dimensional vector $\mathbf{x}_{ij} \in \mathbf{R}^{d\times 1}$. Each bag is associated with a bag label $Y_i \in \{0,1\}$; and each instance is associated with an instance label $y_{ij} \in \{0,1\}$ too. The relation between bag label and instance labels, which is also called *MIL constraints*, is interpreted in the following way:

- If $Y_i = 0$, then $y_{ij} = 0$ for all $j \in [1,\dots,m_i]$, *i.e.*, no instance in the bag is positive.
- If on the hand $Y_i = 1$, then at least one instance $\mathbf{x}_{ij} \in X_i$ is a positive instance of the underlying concept.

In RMI-SVM, we relax the instance label y_i to be a continues variable in the range of $[0,1]$, which is the probability of $\mathbf{x}_{ij}$ being positive, denoted as p_{ij}. Without loss of generality, we use a linear model as instance model. p_{ij} is given by a logistic function

$$p_{ij} = Pr(y_{ij} = 1|\mathbf{x}_{ij};\mathbf{w}) = \frac{1}{1+e^{-\mathbf{w}^T\mathbf{x}_{ij}}}, \quad (1)$$

where $\mathbf{w}$ is the weight vector of the linear model which needs to be optimized through in our formulation.

Only knowing the positive probability of instances is far from enough since the final goal of MIL is to predict whether a bag is positive. And we only know the bag-level label, but do not know the instance-level label. To bridge the gap between instance level and bag level, we adopt the Noisy-OR(NOR) model. The probability of bag regarded as positive is computed via

$$P_i = Pr(Y_i = 1|X_i;\mathbf{w}) = 1 - \prod_{j=1}^{m_i}(1-p_{ij}). \quad (2)$$

Assuming that one instance in the bag is predicted as positive, *e.g.*, $p_{ij} = 1$, then we can find $P_i = 1$ according to Eq.(2). If all the instances in the bag are predicted as zero, we can find $P_i = 0$. The NOR model is a relaxed version of the MIL constraints.

3.2. Objective Function

The above relaxations make the MIL problem more tractable, because there is no discrete variable and all parts in Eq. (3) are differentiable. Considering the instance-level loss, bag-level loss, and model regularization, we give our MIL objective function as follows:

$$\min_{\mathbf{w}} \frac{\lambda}{2}\|\mathbf{w}\|^2 + \frac{\beta}{n}\sum_{i=1}^{n}\mathcal{L}_{bag_i} + \frac{1}{n}\sum_{i=1}^{n}\frac{1}{m_i}\sum_{j=1}^{m_i}\mathcal{L}_{ins_{ij}}, \quad (3)$$

where the first regularization item is to avoid overfitting; $\mathcal{L}_{bag_i}$ denotes the cost item for i-th bag prediction and $\mathcal{L}_{ins_{ij}}$ denotes the cost item for ij-th instance prediction. More specifically, they are denoted as

$$\mathcal{L}_{bag_i} = -\{Y_i \log P_i + (1-Y_i)\log(1-P_i)\}, \quad (4)$$
$$\mathcal{L}_{ins_{ij}} = \max(0, [m_0 - \mathrm{sgn}(p_{ij}-p_0)\mathbf{w}^T\mathbf{x}_{ij}]). \quad (5)$$

where sgn is the sign function; m_0 is a crucial margin parameter used to separate the positive instances and negative instances distant from the hyper line in the feature space; p_0 is a threshold parameter to determine positive or instance.

The goal of RMI-SVM is to find an optimal instance model to determine the label of instances and bags. Thereby, the optimal instance model is given by:

$$\mathbf{w}^* = \operatorname*{argmin}_{\mathbf{w}} \frac{\lambda}{2}\|\mathbf{w}\|^2 + \frac{\beta}{n}\sum_{i=1}^{n}\mathcal{L}_{bag_i} + \frac{1}{n}\sum_{i=1}^{n}\frac{1}{m_i}\sum_{j=1}^{m_i}\mathcal{L}_{ins_{ij}}. \quad (6)$$

The positiveness of instance is given by $p_{ij} = \frac{1}{1+e^{-\mathbf{w}^{*T}\mathbf{x}_{ij}}}$. If $p_{ij} \geq p_0$, $y_{ij} = 1$; otherwise, $y_{ij} = 0$.

3.3. Derivations

The above optimization problem in Eq. (3) can be solved using stochastic gradient descent. Therefore, we derive the partial derivative of $\mathcal{L}_{bag_i}$ and $\mathcal{L}_{ins_{ij}}$ to the weight vector $\mathbf{w}$.

Using the chain rule of calculus in Eq. (4), the partial derivative of $\mathcal{L}_{bag_i}$ with respect to $\mathbf{w}$ is derived as

$$\frac{\partial \mathcal{L}_{bag_i}}{\partial \mathbf{w}} = \frac{\partial \mathcal{L}_{bag_i}}{\partial P_i}\cdot\sum_{j=1}^{m_i}\frac{\partial P_i}{\partial p_{ij}}\frac{\partial p_{ij}}{\partial \mathbf{w}}, \quad (7)$$

where $\frac{\partial \mathcal{L}_{bag_i}}{\partial P_i}$ and $\frac{\partial P_i}{\partial p_{ij}}$ is given by

$$\frac{\partial \mathcal{L}_{bag_i}}{\partial P_i} = -\{\frac{Y_i}{P_i} - \frac{(1-Y_i)}{1-P_i}\} = -\frac{Y_i - P_i}{P_i(1-P_i)}; \quad (8)$$
$$\frac{\partial P_i}{\partial p_{ij}} = \prod_{k=1,k\neq j}(1-p_{ik}) = \frac{\prod_{k=1}^{m_i}(1-p_{ik})}{(1-p_{ij})} = \frac{1-P_i}{1-p_{ij}}. \quad (9)$$

According to Eq. (1), we can find the partial derivative of p_{ij} to w is

$$\begin{aligned}\frac{\partial p_{ij}}{\partial \mathbf{w}} &= -(1+e^{-\mathbf{w}^T\mathbf{x}_{ij}})^{-2}\cdot e^{-\mathbf{w}^T\mathbf{x}_{ij}}\cdot(-\mathbf{x}_{ij}) \\ &= p_{ij}(1-p_{ij})\cdot\mathbf{x}_{ij}.\end{aligned} \quad (10)$$

Appying Eq.(8, 9, 10) to Eq. (7), the final expressoin of partial derivative of $\mathcal{L}_{bag_i}$ with respect to $\mathbf{w}$ is

$$\frac{\partial \mathcal{L}_{bag_i}}{\partial \mathbf{w}} = -\sum_{j=1}^{m_i}\frac{p_{ij}(Y_i - P_i)}{P_i}\mathbf{x}_{ij}. \quad (11)$$

As for the partial derivative of $\mathcal{L}_{ins_{ij}}$ with respect to $\mathbf{w}$, this expression is derived as

$$\frac{\partial \mathcal{L}_{ins_{ij}}}{\partial \mathbf{w}} = -\mathbf{1}[\mathrm{sgn}(p_{ij}-p_0)\mathbf{w}^T\mathbf{x}_{ij} < m_0]\cdot \mathrm{sgn}(p_{ij}-p_0)\mathbf{x}_{ij}, \tag{12}$$

where $\mathbf{1}[\mathrm{sgn}(p_{ij}-p_0)\mathbf{w}^T\mathbf{x}_{ij} < m_0]$ is an indicator function which equals one if its argument is true and zero otherwise.

3.4. SGD Optimization

We describe the optimization method in this subsection and also provide the pseudo-code. As mentioned in Sec. 3, our method performs SGD on the objective in Eq. (3) with a varied learning rate strategy. On a iteration t in our algorithm, we randomly choose a bag (X_{k_t}, Y_{k_t}) from the training sets $\mathcal{D}$ via picking an index $k_t \in \{1, 2, ..., n\}$ in a standard uniform distribution. Then we change the objection in Eq. (3) to an approximation based on the sample bag, obtaining

$$f(\mathbf{w}; X_{k_t}) = \frac{\lambda}{2}\|\mathbf{w}\|^2 + \beta\mathcal{L}_{bag_{k_t}} + \frac{1}{m_{k_t}}\sum_{j=1}^{m_{k_t}}\mathcal{L}_{ins_{k_tj}}. \tag{13}$$

Considering the gradient of the approximate function, given by

$$\nabla_t = \frac{\partial f(\mathbf{w}; X_{k_t})}{\partial \mathbf{w}} = \lambda\mathbf{w} - \sum_{j=1}^{m_{k_t}}\mathbf{x}_{k_tj}\{\beta\cdot\frac{p_{k_tj}(Y_{k_t}-P_{k_t})}{P_{k_t}} + \frac{\mathrm{sgn}(p_{k_tj}-p_0)}{m_{k_t}}\cdot\mathbf{1}[\mathrm{sgn}(p_{k_tj}-p_0)\mathbf{w}^T\mathbf{x}_{k_tj} < m_0]\}, \tag{14}$$

we update the weight vector using a varied learning rate $\eta_t = 1/[(t+1)\cdot\lambda]$, that is $\mathbf{w}_{t+1} \leftarrow \mathbf{w}_t - \eta_t\cdot\nabla_t$. When t reaches a predefined iteration T, we output the last weight $\mathbf{w}_T$. It is worth noting that after each gradient update, we employ a projection operation of $\mathbf{w}$ on the L_2 ball of radius $1/\sqrt{\lambda}$ just as mentioned in [24] via the following update,

$$\mathbf{w}_{t+1} \leftarrow \min\{1, \frac{1/\sqrt{\lambda}}{\|\mathbf{w}_{t+1}\|}\}\mathbf{w}_{t+1}. \tag{15}$$

This modification can significantly accelerate the rate of convergence in the optimization step.

In summary, the pseudo-code for solving RMI-SVM is given in Algorithm 1, which is granted to get a local optimal solution for the objective function Eq. (3). In practical application, it gives satisfactory accuracy and fast speed.

Algorithm 1: Pseudo-code for solving RMI-SVM.

Input: $\mathcal{D}$, λ, β, p_0, m_0, T
Output: $\mathbf{w}_{T+1}$
begin
  Initialize: Set $\mathbf{w}_1 = 0$
  for $t = 1, 2, ..., T$ **do**
    choose $k_t \in \{1, 2, ..., n\}$, uniform distribution
    Set $j^+ = \{j | \mathrm{sgn}(p_{k_tj} - p_0)\mathbf{w}_t\mathbf{x}_{k_tj} < m_0\}$
    Set $m_{k_t} = |X_{k_t}|$
    Set $\eta_t = \frac{1}{\lambda t}$
    Set $\mathbf{w}_{t+1} \leftarrow$
    $\{(1-\eta_t\lambda)\mathbf{w}_t + \beta\eta_t\sum_j \mathbf{x}_{k_tj}\frac{p_{k_tj}(Y_{k_t}-P_{k_t})}{P_{k_t}} +$
    $\frac{\eta_t}{m_{k_t}}\sum_{j^+}\mathrm{sgn}(p_{k_tj^+} - p_0)\mathbf{x}_{k_tj^+}\}$
    Set $\mathbf{w}_{t+1} \leftarrow \min\{1, \frac{1/\sqrt{\lambda}}{\|\mathbf{w}_{t+1}\|}\}\mathbf{w}_{t+1}$

4. Experiments on MIL Benchmarks

In this and the following section, we perform experiments to test RMI-MIL for bag classification on MIL benchmarks and object discovery in the wild, respectively. RMI-MIL is implemented in MATLAB and experiments are carried out on a desktop machine with Intel(R) Core(TM) i7-3930K CPU (3.20GHz) and 64GB RAM. The code will be released on publication. In the following subsections, three widely-used MIL benchmarks on different applications are tested.

4.1. Drug Activation Prediction

The task is to predict whether a new drug molecule can bind well to a target protein, which is mainly determined by the shape of the molecule. A "right" molecular shape can bind well to the target protein. Unfortunately, a molecule always exhibits multiple shapes. In this case, a good molecule will bind well if *at least* one of its shapes is right, while a poor molecule will not bind well if *none* of its shapes can bind. Therefore, the drug prediction task can be formulated as a MIL problem.

The widely-used MUSK datasets described in [11] for drug prediction are the benchmarks in nearly every previous MIL algorithm. Both of the datasets, MUSK1 and MUSK2, are composed of representations of molecules (bags) in multiple low-energy conformations (instances). Each conformation is described by a 166-dimensional feature vector derived from its surface properties. MUSK1 contains 476 instances divided into 47 positive bags and 45 negative bags, while MUSK2 owns approximately 6600 instances grouped into 39 positive bags and 63 negative bags. Another difference of these two datasets is that MUSK2 consists of more fraction of negative instances in a bag.

For this task, we set $\lambda = 0.05$, $\beta = 1.5$ and $m_0 = 0.5$ in the proposed algorithm. *For all our experiments including this and the following tasks, we fix the* p_0 *in Eq.* (5) *to* 0.5 *and the maximum iteration* T *to 2000 by default if we don't particularly point out.* We compare our results with miSVM and MISVM proposed in [1] in Table 1, which show that both MISVM and RMI-SVM achieve a similar accuracy on MUSK1 dataset and outperform miSVM by a few percent. Furthermore on MUSK2 dataset, RMI-SVM per-

forms marginally better than miSVM, which is susceptible to local minima. Note that the results of miSVM and MISVM are implemented via linear kernel for fair comparison with RMI-SVM.

Table 1. Average prediction accuracy (%) via ten times 10-fold cross validation on MUSK datasets. Please note that we all adopt linear kernels for fair comparison.

Dataset	MISVM	miSVM	RMI-SVM
MUSK1	80.4	78.0	**80.8**
MUSK2	77.5	70.2	**82.4**

4.2. Automatic Image Annotation

Widely applied to image retrieval systems, this task is the process by which an intelligent system automatically assigns context information in the form of *keywords* to digital images. An image (bag) contains a set of regions/segments (instances) which denote different visual objects. Assuming that a user is searching for a target object, an image is regarded as a relevant retrieval if only one of its regions is relevant, while other regions are relevant or not.

We perform three classification experiments on "elephant", "fox" and "tiger" classes in the Corel dataset [4]. More specifically, each image (bag) consists of plenty of segments (instances) and a 320-dimensional feature is extracted to represent the color, texture and shape characteristics of a segment. There are 100 positive/relevant images and 100 negative/irrelevant ones for each dataset. As for each image, the number of positive segments (instances) is approximately the same with that of negative ones.

In this task, all instance feature are preprocessed by $L2$ normalization as input. The parameters are given as $\lambda = 0.02, \beta = 5$ and $m_0 = 2$. We compare our method with miGraph and MIGraph in [36], miFV in [32], miSVM and MISVM in [11], EM-DD in [35], MILES [6], MIForests in [19] and PPMM in [30] via ten times 10-fold cross validation and report the average results and corresponding standard deviation in Table 2. The results of MI-Kernel was taken from [36]. Note that some standard deviations in former studies are not available. RMI-SVM achieves the best results on the three datasets.

4.3. Text Categorization

The task is to assign predefined categories to text documents. A document (bag) may be labeled as relevant to certain topic only if some unspecified paragraphs/keywords(instances) of it are relevant. In other words, a document is usually regarded as irrelevant if there are no relevant paragraphs/keywords. Therefore, document classification can be naturally formulated as a multiple instance problem.

[1]The results are reported in integer over 5 runs in [19].

Table 2. Average prediction accuracy (%) via ten times 10-fold cross validation on benchmarks. Some standard deviations in former approaches are not available.

Algorithm	Elephant	Fox	Tiger
RMI-SVM	**87.8±0.7**	**63.6±2.8**	**87.9±0.9**
MIGraph	85.1±2.8	61.2±1.7	81.9±1.5
miGraph	86.8±0.7	61.6±2.8	86.0±1.6
miFV	85.2±0.8	62.1±1.1	81.3±0.8
MI-Kernel	84.3±1.6	60.3±1.9	84.2±1.0
MISVM	81.4	57.8	84.0
miSVM	82.2	58.2	78.4
EM-DD	78.3	56.1	72.1
PPMM	82.4	60.3	82.4
MIForests[1]	84	**64**	82
MILES[1]	81	62	80

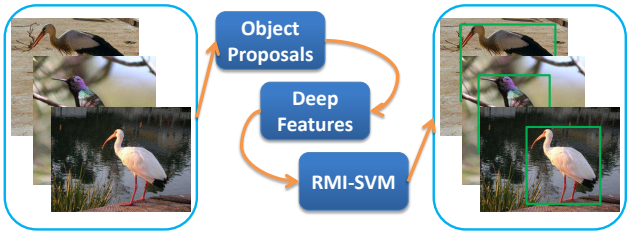

Figure 2. Illustration of our object discovery pipeline in the experiments. At first, the object proposal method Edgebox extracts candidate object regions. Then, for every candidate, its DCNN feature is extracted. At last, RMI-SVM identifies positive instances as object discovery results.

We test the proposed method on datasets from text categorization. The evaluated datasets are randomly split and subsampled from the original TREC9 dataset. Compared with those datasets used in Sec. 4.1 and 4.2, the representation is extremely sparse and high-dimensional with more than 66000 dimension but less than 32 non-zero values, which makes them challenging datasets.

In this task, we set parameters as $\lambda = 0.0003, \beta = 4$ and $m_0 = 2$ and all data is $L2$ normalized. During the experiments, we find that slight changes in the parameters make minor difference to the final average accuracy. Results of the proposed approach are reported in Table 3. We achieve the best results over the previous methods in all the seven subsets. The average classification accuracy is improved by more than 3 percent. Note that RMI-SVM with linear kernel consistently performs better than both miSVM and MISVM whatever the kernels they adopt.

5. Experiments of Object Discovery in the Wild

5.1. Datasets and Evaluation Criteria

In this section, we perform weakly-supervised object discovery in natural images following the pipeline shown

Table 3. Classification accuracy (%) of methods on seven subsets from TREC9. The standard deviations of other methods are not available.

Dataset Category	Dims Ins/Feat	EM-DD	miSVM			MISVM			RMI-SVM
			linear	poly	rbf	linear	poly	rbf	
TST1	3224/66552	85.8	93.6	92.5	90.4	93.9	93.8	93.7	$\mathbf{95.0 \pm 1.0}$
TST2	3344/66153	84.0	78.2	75.9	74.3	84.5	84.4	76.4	$\mathbf{86.3 \pm 0.8}$
TST3	3246/66144	69.0	87.0	83.3	69.0	82.2	85.1	77.4	$\mathbf{87.9 \pm 0.6}$
TST4	3391/68085	80.5	82.8	80.0	69.6	82.4	82.9	77.3	$\mathbf{85.3 \pm 1.0}$
TST7	3367/66823	75.4	81.3	78.7	81.3	78.0	78.7	64.5	$\mathbf{82.3 \pm 0.8}$
TST9	3300/66627	65.5	67.5	65.6	55.2	60.2	63.7	57.0	$\mathbf{71.2 \pm 0.7}$
TST10	3453/66082	78.5	79.6	78.3	52.6	79.5	81.0	69.1	$\mathbf{83.9 \pm 0.8}$
Average	3332/66638	77.0	81.4	79.2	70.3	80.1	81.4	73.6	$\mathbf{84.8 \pm 0.8}$

in Fig. 2. Given a set of images, we firstly utilize Edgebox [38] to capture plenty of windows/patches as object proposals. This strategy turns the object discovery problem into a well-defined MIL problem, in which an image is a bag, an object proposal is an instance, and image label is used as bag label. Then, a pre-trained DCNN is applied to extract the rich semantic feature for each object proposal. Here, we use the BVLC AlexNet model provide in Caffe Model Zoo [17]. Furthermore, we treat the images containing a shared object as the positive set and randomly select images from the remaining images as negative. At last, after that the models adopting the proposed method are learnt, we report the object proposals with maximal value predicted by RMI-SVM as the detected object. The final results evaluated via CorLoc measure [10], which is the percentage of the correct location of objects under the Pascal criteria (intersection over union (IoU) > 0.5 between detected bounding boxes and the ground truth).

The popular Pascal 2006 and 2007 datasets [13] are extremely challenging and have been widely used as the benchmarks to evaluate object discovery methods. Following the protocol of [10], two subsets are taken from Pascal 2006 and 2007 *train+val* dataset, which are then divided into various of class and view combinations. The two subsets are referred as *Pascal06-all* and *Pascal07-all* below, respectively. There are in total 2047 images divided into 45 class/viewpooint combinations in Pascal07-all while total 2184 images from 33 class/viewpoint in Pascal06-all. Besides of Pascal06-all and Pascal07-all, recent methods start to focus on the 20 classes Pascal 2007 training set (denoted as Pascal 2007) without considering view variations, which makes the object discovery task more challenging. Thus, in the experiments, we have three different testing sets: Pascal06-all, Pascal07-all, and Pascal 2007. Following the common setting [10], for the three sets, we use all images that contain at least one object instance not marked as truncated or difficult in the ground truth.

We utilize the Structured Edge Detection Toolbox in [38] to extract a large number of object proposals. The parameters are given via the step size of 0.65, Non-maximal suppression (NMS) threshold of 0.55, minimum score of boxes to detect of 0.1 and maximal number of boxes to detect of 400. For two object proposals in NMS, if the ratio of intersect area to union area is greater than a given threshold, then the proposal with the lower score is suppressed. As for the DCNN feature extraction, we adopt the exact output of the $fc6$ layer, whose dimension is 4096. On Pascal07-all dataset, we set $\lambda = 0.0015, \beta = 5$ and $m_0 = 1.2$, while $\lambda = 0.0015, \beta = 6$ and $m_0 = 0.2$ on Pascal06-all dataset.

Table 4. Object discovery results evaluated via CorLoc on Pascal06-all and Pascal07-all.

Dataset	Ours	bMCL [37]	ADMM [31]	MIForests [19]	WSDPM [22]	Deselaers *et.al.*[10]
Pascal06-all	**53**	45	43	36	N/A	49
Pascal07-all	**37**	31	27	25	30	28

5.2. Comparison to State-of-the-arts

5.2.1 Pascal06-all and Pascal07-all

The results of the proposed method on Pascal06-all and Pascal07-all are compared with the former state-of-the-art works and shown in Table 4. Our method consistently yields better performance than other former state-of-the-art approaches on the two datasets. The CorLoc measures have been improved by 4% and 9% on Pascal06-all and Pascal07-all respectively. Some object discovery results are shown in Fig. 4.

The CorLoc measure is not accurate enough since there may have more than one object of interests in image. To better characterize the discovery performance, we plot the detection v.s. the number of detections curve in Fig. 3, and compare our method to miSVM and Edgebox. In the compared classes, our RMI-SVM can consistently and significantly improves Edgebox; but miSVM failed. The curves also show that our RMI-SVM is more robust than miSVM.

5.2.2 Pascal 2007

The object discovery results on the 20 classes Pascal 2007 set measure by CorLoc are given in Table 5. Recent weakly-supervised detectors are compared, including the previous state-of-the-art method named Multi-fold MIL [9]. The average CorLoc of Multi-fold MIL is 38.8%. It uses the ad-

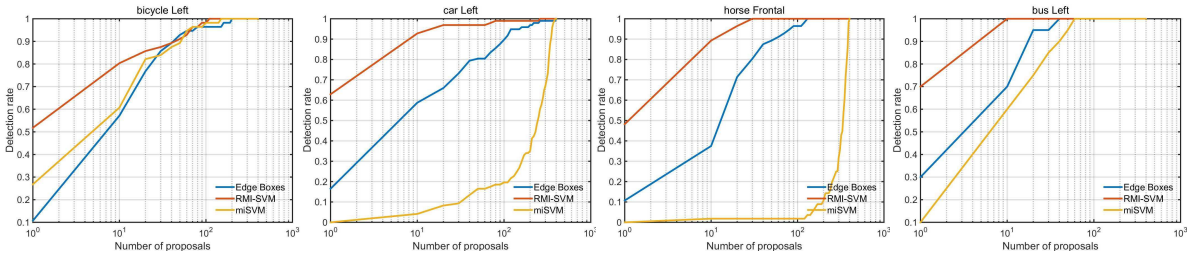

Figure 3. Detection rates when changing the number of detections/proposals on four class/viewpoint combinations. These combinations are, from left to right and top to bottom, Bicycle/Left, car/Left, House/Frontal, Bus/Left.

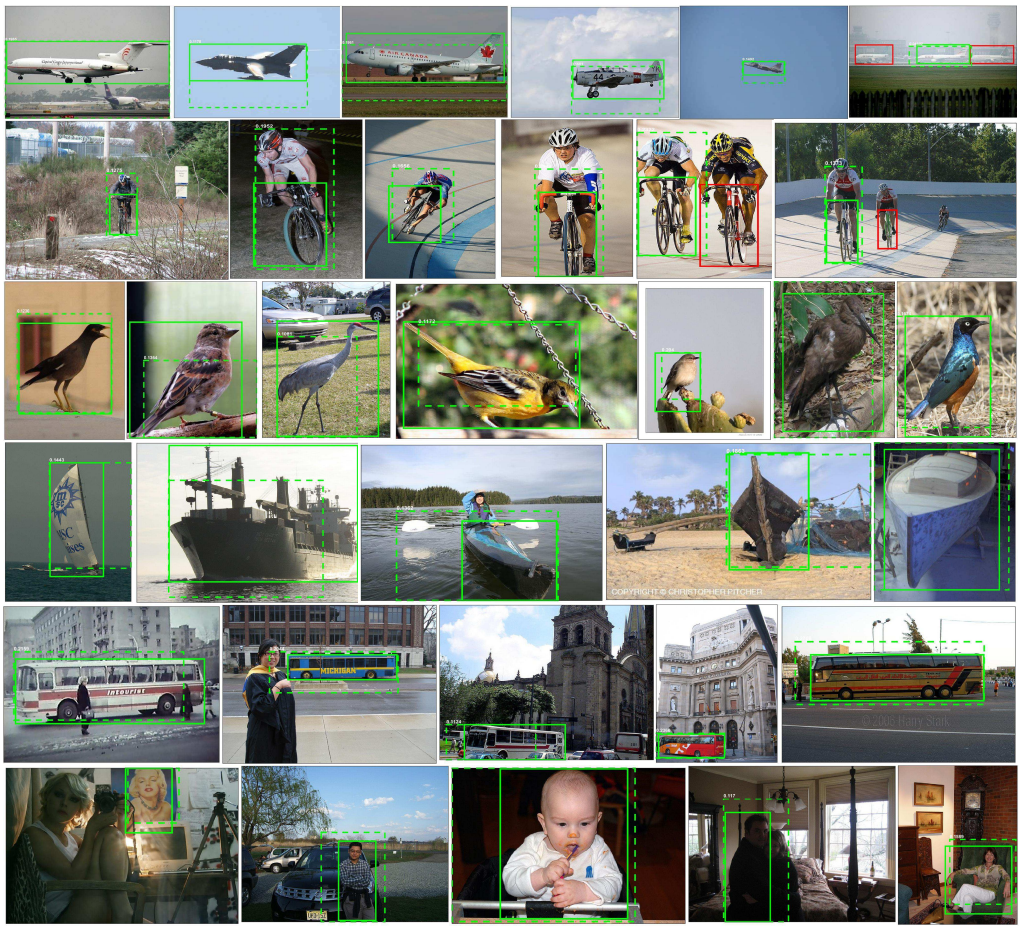

Figure 4. Results of object discovery on several class/viewpoint combinations on Pascal07-all set. Each row denotes one combination. These combinations are, from top row to bottom, Aeroplane/Left, Bicycle/Frontal, Bird/Right, Boat/Frontal, Bus/Left, Person/Frontal. It is worth noting that the solid green rectangle denotes the matched ground truth; the dashed green rectangle denotes the matched detection; and the solid red rectangle denotes the missed ground truth. Best viewed in color.

vanced fisher vector coding [23] to extract object feature. Our RMI-SVM based method improves the average CorLoc to 40.2% and wins in 7 out of 20 classes. The good results indicate that: (1) The proposed RMI-SVM is more robust and effective than other MIL algorithms, such as, the Multi-fold MIL and miSVM; (2) The DCNN feature used in our paper is very robust to view variation, since DCNN is learnt from the huge ImageNet dataset.

5.3. Improvement of detection performance over Edgebox

Edgebox is a method for generating object bounding box proposals using informative edges. However, it is imperative to extract a large number of proposals to reach a high detection rate. Given the labels of each image, we conduct experiments to demonstrate that RMI-SVM can assist

Table 5. Object discovery results evaluated via CorLoc of all 20 classes on Pascal 2007 training set. Note that the last column is the average CorLoc of all 20 classes. The best result of each class is emphasized in bold.

Algorithm	aero	bicy	bird	boa	bot	bus	car	cat	cha	cow	dtab	dog	hors	mbik	pers	plnt	she	sofa	trai	tv	Av.
Multi-fold MIL[9]	56.6	58.3	28.4	**20.7**	**6.8**	**54.9**	69.1	20.8	**9.2**	50.5	10.2	29.0	58.0	64.9	**36.7**	18.7	**56.5**	13.2	54.9	**59.4**	38.8
Shi *et al.*'13[25]	**67.3**	54.4	34.3	17.8	1.3	46.6	60.7	**68.9**	2.5	32.4	16.2	**58.9**	51.5	64.6	18.2	3.1	20.9	34.7	**63.4**	5.9	36.2
Siva *et al.*'13[27]	–	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	32.0
Siva&Xiang'11[28]	42.4	46.5	18.2	8.8	2.9	40.9	**73.2**	44.8	5.4	30.5	19.0	34.0	48.8	**65.3**	8.2	9.4	16.7	32.3	54.8	5.5	30.4
Siva *et al.*'12 [26]	45.8	21.8	30.9	20.4	5.3	37.6	40.8	51.6	7.0	29.8	27.5	41.3	41.8	47.3	24.1	12.2	28.1	32.8	48.7	9.4	30.2
Ours	37.7	**58.8**	**39.0**	4.7	4.0	48.4	70.0	63.7	9.0	**54.2**	**33.3**	37.4	**61.6**	57.6	30.1	**31.7**	32.4	**52.8**	49.0	27.8	**40.2**

Edgebox on detection task to a great margin even with a few number of object proposals. Under the criteria of IoU > 0.5, we show the detection rates on Pascal07-all when varying the number of detections. Several results of different class/viewpoint combinations are given in Fig. 3. We can find that the detection rates are greatly improved via using the weakly supervised information.

5.4. Comparison to miSVM

We compare the effectiveness of the proposed RMI-SVM with the conventional miSVM on the object discovery task. As stated in 5, we first make use of the Edgebox to extract object proposals. Then deep representation is captured using Convolutional Neural Network, which is finally the process of multiple instance learning. To keep fair comparison, we replace the RMI-SVM with miSVM to guarantee the exactly same features as input in the final learning step. Results are given in Table 6, which demonstrates that the learning ability of RMI-SVM is superior over miSVM to a great margin under the MIL constraints. The Liblinear [14] toolbox is chosen in the implementation of miSVM.

As shown in Fig. 3, the RMI-SVM is superior to miSVM when choosing the number of proposals as 1, which is exactly the CorLoc evaluation. It obviously shows that miSVM fails in learning the common attributes in the same class/viewpoint. In miSVM framework, all the instances in positive bag are initialized as positive, followed by updating instance labels in each iteration. This learning strategy seems reasonable in Sections 4.1, 4.2 and 4.3 since the ratio of positive instances to negative ones is approximately 1, except the MUSK2 dataset where the ratio is $\frac{1}{6}$. When the number of positive instances makes up to quite a large portion in a positive bag, miSVM can find a hyperplane that divides the negative instances as true negative. However, positive proposals in positive images hold a small portion after the NMS on object discovery, usually less than $\frac{1}{20}$. Even though miSVM wrongly classifies the negative instances in negative bag as positive, it considers little penalty in each iteration. Thus, miSVM which is always susceptible to local minima cannot distinguish background windows/patches well from the shared object proposals. However, RMI-SVM accounts for the penalty of false positive via the term in Eq. (8). Thus, RMI-SVM can well separate the background proposals from the common object proposals.

Table 6. Comparison between RMI-SVM and miSVM via CorLoc evaluation and running time on object discovery experiments.

Evaluation	RMI-SVM	miSVM
CorLoc (%) on Pascal06-all	**53**	30
CorLoc (%) on Pascal07-all	**37**	20
Running time (s) on Pascal07-all	**854**	4300

Furthermore, we experimentally compare the time complexity of RMI-SVM with miSVM. On Pascal07-all dataset for object discovery, it takes RMI-SVM around 854 seconds to learn 45 models for all combinations, while miSVM spends more than 4300 seconds. RMI-SVM is 5 times efficient than the conventional miSVM. The performance gain in running time should be owned to our novel formulation and the fast SGD. The SGD in RMI-SVM randomly uses only one bag. As for in every iteration of miSVM, it takes all instances of all bags as input, which is the crucial issue of time consuming. Other EM-style MIL methods, *e.g.*, MILBoost, have the same mechanism, and are less efficient than our RMI-SVM.

6. Conclusion

In this paper, we have proposed a novel formulation for MIL and applied it for robust weakly-supervised object discovery. Different from the traditional EM-style MIL solutions, we relax the highly combinatorial MIL optimization problem into a convex program and solve it efficiently using SGD. Our idea of solving MIL in a relaxed formulation is general. More complex discriminative model and model regularization method, *e.g.*, deep neural networks, can be adopted. Besides of object discovery, RMI-SVM can also be used to solve other recognition tasks, such as visual tracking, image classification, and learning part-based object detection model.

7. Acknowledgement

This work was supported by National Natural Science Foundation of China (NSFC) (NO. 61503145, NO. 61222308 and NO. 61573160), the Fundamental Research Funds for the Central Universities (HUST 0118181099), and CCF-Tencent RAGR (NO. 20140116).

References

[1] S. Andrews, I. Tsochantaridis, and T. Hofmann. Support vector machines for multiple-instance learning. In *NIPS*, 2002. 2, 4

[2] B. Babenko, N. Verma, P. Dollár, and S. J. Belongie. Multiple instance learning with manifold bags. In *ICML*, 2011. 2

[3] B. Babenko, M.-H. Yang, and S. Belongie. Visual tracking with online multiple instance learning. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, August 2011. 2

[4] C. Carson, M. Thomas, S. Belongie, J. M. Hellerstein, and J. Malik. Blobworld: A system for region-based image indexing and retrieval. In *Visual Information and Information Systems*, pages 509–517. Springer, 1999. 5

[5] X. Chen, A. Shrivastava, and A. Gupta. Enriching visual knowledge bases via object discovery and segmentation. In *CVPR*, pages 2035–2042, 2014. 1, 2

[6] Y. Chen, J. Bi, and J. Z. Wang. Miles: Multiple-instance learning via embedded instance selection. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 28(12):1931–1947, 2006. 5

[7] Y. Chen and J. Z. Wang. Image categorization by learning and reasoning with regions. *Journal of Machine Learning Research*, 5:913–939, 2004. 2

[8] M. Cho, S. Kwak, C. Schmid, and J. Ponce. Unsupervised object discovery and localization in the wild: Part-based matching with bottom-up region proposals. *arXiv preprint arXiv:1501.06170*, 2015. 2

[9] R. G. Cinbis, J. J. Verbeek, and C. Schmid. Multi-fold MIL training for weakly supervised object localization. In *CVPR*, 2014. 1, 2, 6, 8

[10] T. Deselaers, B. Alexe, and V. Ferrari. Weakly supervised localization and learning with generic knowledge. *International Journal of Computer Vision*, 100(3):275–293, 2012. 6

[11] 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, 1997. 1, 2, 4, 5

[12] P. Dollár, B. Babenko, S. Belongie, P. Perona, and Z. Tu. Multiple component learning for object detection. In *ECCV*, 2008. 2

[13] M. Everingham, L. Van Gool, C. K. Williams, J. Winn, and A. Zisserman. The pascal visual object classes (voc) challenge. *International Journal of Computer Vision*, 88(2):303–338, 2010. 6

[14] R.-E. Fan, K.-W. Chang, C.-J. Hsieh, X.-R. Wang, and C.-J. Lin. LIBLINEAR: A library for large linear classification. *Journal of Machine Learning Research*, 9:1871–1874, 2008. 8

[15] H. Fang, S. Gupta, F. Iandola, R. Srivastava, L. Deng, P. Dollr, J. Gao, X. He, M. Mitchell, J. Platt, C. Zitnick, and G. Zweig. From captions to visual concepts and back. In *CVPR*, 2015. 1, 2

[16] P. F. Felzenszwalb, R. B. Girshick, D. McAllester, and D. Ramanan. Object detection with discriminatively trained part-based models. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 32(9):1627–1645, 2010. 2

[17] Y. Jia, E. Shelhamer, J. Donahue, S. Karayev, J. Long, R. Girshick, S. Guadarrama, and T. Darrell. Caffe: Convolutional architecture for fast feature embedding. *arXiv preprint arXiv:1408.5093*, 2014. 6

[18] A. Krizhevsky, I. Sutskever, and G. E. Hinton. Imagenet classification with deep convolutional neural networks. In *NIPS*, 2012. 1, 2

[19] C. Leistner, A. Saffari, and H. Bischof. Miforests: Multiple-instance learning with randomized trees. In *ECCV*, 2010. 5, 6

[20] G. Liu, J. Wu, and Z.-H. Zhou. Key instance detection in multi-instance learning. In *ACML*, 2012. 2

[21] O. Maron and T. Lozano-Pérez. A framework for multiple-instance learning. *NIPS*, 1998. 2

[22] M. Pandey and S. Lazebnik. Scene recognition and weakly supervised object localization with deformable part-based models. In *ICCV*, 2011. 6

[23] F. Perronnin, J. Sánchez, and T. Mensink. Improving the fisher kernel for large-scale image classification. In *ECCV*. Springer, 2010. 7

[24] S. Shalev-Shwartz, Y. Singer, and N. Srebro. Pegasos: Primal estimated sub-gradient solver for SVM. In *ICML*, 2007. 4

[25] Z. Shi, T. M. Hospedales, and T. Xiang. Bayesian joint topic modelling for weakly supervised object localisation. In *ICCV*, 2013. 8

[26] P. Siva, C. Russell, and T. Xiang. In defence of negative mining for annotating weakly labelled data. In *ECCV*, 2012. 8

[27] P. Siva, C. Russell, T. Xiang, and L. de Agapito. Looking beyond the image: Unsupervised learning for object saliency and detection. In *CVPR*, 2013. 8

[28] P. Siva and T. Xiang. Weakly supervised object detector learning with model drift detection. In *ICCV*, 2011. 8

[29] K. Tang, A. Joulin, L.-J. Li, and L. Fei-Fei. Co-localization in real-world images. In *CVPR*, 2014. 1, 2

[30] H.-Y. Wang, Q. Yang, and H. Zha. Adaptive p-posterior mixture-model kernels for multiple instance learning. In *ICML*, 2008. 5

[31] X. Wang, Z. Zhang, Y. Ma, X. Bai, W. Liu, and Z. Tu. Robust subspace discovery via relaxed rank minimization. *Neural Computation*, 26(3):611–635, 2014. 2, 6

[32] X. Wei, J. Wu, and Z. Zhou. Scalable multi-instance learning. In *ICDM*, 2014. 2, 5

[33] J. Wu, Y. Yu, C. Huang, and K. Yu. Deep multiple instance learning for image classification and auto-annotation. In *CVPR*, 2015. 2

[34] C. Zhang, J. C. Platt, and P. A. Viola. Multiple instance boosting for object detection. In *NIPS*, 2005. 2

[35] Q. Zhang and S. A. Goldman. EM-DD: an improved multiple-instance learning technique. In *NIPS*, 2001. 2, 5

[36] Z. Zhou, Y. Sun, and Y. Li. Multi-instance learning by treating instances as non-i.i.d. samples. In *ICML*, 2009. 2, 5

[37] J. Zhu, J. Wu, Y. Xu, E. I. Chang, and Z. Tu. Unsupervised object class discovery via saliency-guided multiple class learning. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 37(4):862–875, 2015. 2, 6

[38] C. L. Zitnick and P. Dollár. Edge boxes: Locating object proposals from edges. In *ECCV*, 2014. 1, 2, 6