IEEE Access
Multidisciplinary | Rapid Review | Open Access Journal

Received July 13, 2019, accepted August 8, 2019, date of publication August 13, 2019, date of current version September 5, 2019.

Digital Object Identifier 10.1109/ACCESS.2019.2935011

Logistic Regression Region Weighting for Weakly Supervised Object Localization

LIANTAO WANG[1], TINGWEI WANG[2], AND XUELEI HU[3]

[1]Key Laboratory of Sensor Networks and Environmental Sensing, Hohai University, Changzhou 213022, China
[2]School of Information Science and Engineering, University of Jinan, Jinan 250013, China
[3]School of Information Technology and Electrical Engineering, The University of Queensland, Brisbane, QLD 4072, Australia

Corresponding author: Liantao Wang (ltwang@hhu.edu.cn)

This work was supported in part by the National Natural Science Foundation of China under Grant 61703139, and in part by the Key Research and Development Program of Jiangsu Province (Social Development) under Grant BE2019649.

ABSTRACT In this paper, we address the problem of weakly supervised object localization using region weighting. For a weakly labelled image/video, the inside regions have different relevance to its semantic label. We first over-segment an image/video to get super-pixel/voxel regions, and assign each region with a latent weight to represent its support to the semantic label, then regress the weights to right values by optimizing the classification according to the weak labels. We adopt logistic regression as our base model due to its good performance in multiple-instance setting. The latent region weights are incorporated into the objective function as an interpretation of region combination at feature-level. The weights and the model parameters are optimized in an alternate manner. With the updates of the weights, the model is trained on the semantic regions independently of the background, therefore the learned model is capable of distinguishing object and non-object regions, and generating irregular-shape object localization. The method overcomes the limitations of applying multiple-instance learning to visual object localization. Experimental results on three datasets validates the effectiveness of the proposed method.

INDEX TERMS Region weighting, logistic regression, automatic annotation, irregular-shape object localization.

I. INTRODUCTION

Object localization is one of the fundamental challenges in computer vision. Bounding-box is the most common way to locate objects. It is much simpler than pixel-wise mask. However, it would include much background when the objects are non-boxy, which might mislead the training of object detectors. Pixel labelling can tackle this problem, but is too complicated. An alternative is segment annotation. One can first over-segment images/videos using unsupervised methods to get regions roughly homogeneous in color and texture, then distinguish which regions belong to objects. Based on the boundary detection of the over-segmentation algorithms, pixel-wise localization can be realized through easier segment classification. This intuition is first implemented in supervised way [1], [2], and further extended for weak supervision [3], [4].

In a weakly supervised scenario, labels only indicate whether there are objects of interest inside the images/vidoes, yet do not provide any information for their locations. When considering an image/video as a bag, and its inside segments as instances, this segment annotation can be naturally posed as a multiple-instance learning (MIL) problem [5]–[10]. However, MIL has two limitations when applied to this task: 1) Many MIL methods trigger the bag label with the single maximum-score instance, leading to incomplete annotation; 2) MIL methods disregard the spatial relations of the segments, leading to inaccurate annotation.

We aim to overcome these limitations for segment annotation in weakly labelled visual data. Considering a weak label indicating what concept (object or action) is contained in an image/video, it is only part of the image/video that accounts for the semantic label, and the others are background clutters. That means the regions in the image should have different relevance to that label. We consider the relevance intensity as latent weights, and learn them by optimizing the classification in terms of the weak labels. Fig. 1 illustrates how we address the limitations: 1) We consider that a combination of the over-segmented regions rather than a single one triggers the semantic label. 2) Spatial constraint is incorporated

The associate editor coordinating the review of this article and approving it for publication was K.C. Santosh.

into the segment classification to improve the localization performance.

Since logistic regression (LR) has been proved as a competitive model in MIL [11], [12], we adopt it as our base model for implementing region weighting. Our contributions in this paper include the following aspects: 1) We derive a region weighting formulation that has an interpretation of region combination at feature-level, while the traditional MILRs can be considered as the region combination at score-level. 2) The region weights and the LR model parameter are incorporated into a single objective function, and optimized in an alternate manner. 3) The region weighting generates irregular-shape object localization. By taking advantage of deep local features, it performs comparably to state-of-the-art methods.

The rest of this paper is organized as follows. We first review the works related to our methodology in Section II. Then give some preliminaries to our method in Section III. The logistic regression region weighting is detailed in Section IV. In Section V, we provide experimental results and analysis, and finally conclude this paper with Section VI.

II. LITERATURE REVIEW

A. REGION WEIGHTING IN VISUAL LEARNING

Traditional image classification methods do not explicitly handle background clutter, but rely on global image representations. Since image classification and localization are interdependent in visual learning, region weighting is employed to boost the joint performance. Single region selection can be considered as an extreme version of region weighting. One popular strategy is to treat the semantic region as a latent sub-window among the image, and apply a region classifier to localize it. Nguyen *et al.* [6] pick out the sub-window with the maximal classification score as the semantic object representing the image. While Siva *et al.* [7] choose the sub-window that is most different from the negative images as the semantic region. This idea can also be integrated into the deep CNN framework by leveraging the deep feature representations [13] in shallow methods, or by redesigning the last hidden layer to derive an end-to-end method [9]. However, computing CNN features is time consuming so the total number of proposal sub-windows is usually limited to ten or at most several hundred [14]; and CNN features [15] can only be computed on square regions of a certain size, requiring the semantic region to be roughly rectangular. This makes the single sub-window assumption do not usually hold in practice.

Instead of relying on a single sub-window to trigger an image label, Yakhnenko *et al.* [16] score an image using a weighted sum of all grid segments. They associate each grid with a latent weight that indicates whether it belongs to the object of interest or the background, then optimize the weights in linear SVM. Zhao *et al.* [17] extend the idea to non-linear kernel SVM based on multiple kernel learning. Expectation loss SVM [18] is proposed to weight each segment with a function of its positiveness, and is able to learn segment classifiers with image labels. Recently, Wei and Hoai [19] propose another way to overcome the limitation of single sub-window selection. They assume that the semantic region in an image is a weighted combination of multiple overlapping sub-windows. Compared to [17], spatial constraints on the weights are considered in our method to ensure better results. Compared to [19], we impose weights on a set of irregular-shape regions rather than sub-windows, which has the potential to provide more precise irregular-shape object localization.

Other related works include [20], [21]. Instead of localizing each object class independently from other classes, Shi *et al.* [20] propose Bayesian joint topic modelling for exploiting multiple object co-existence. It learns a single background shared across classes and deals with large scale data more efficiently than prior approaches. Kolesnikov and Lampert [21] introduce a scheme for weakly supervised semantic segmentation based on three guiding principles: seeding, expansion and constrain-to-boundary.

B. DEEP LOCAL FEATURES

Image descriptors using the last fully-connected layer in a convolutional neural network (CNN) have emerged as state-of-the-art features for visual recognition [15]. Afterwards, research attention shifted from the features extracted from the fully-connected layer to the features from the convolutional layers of a CNN, either for stronger global image description [22], [23] or for spatial information in object localization [24], [25]. These features have a natural interpretation as descriptors of local image regions, and can be extracted with any size and shape from an image. Cimpoi *et al.* [22] exploit Fisher vector constructed on deep local features to produce global image descriptors for classification. Babenko and Lempitsky [23] aggregate local deep features to produce compact global descriptors for image retrieval. Other researchers [24], [25] concatenate multiple-layer outputs at a pixel to construct hypercolumn for object localization. Zhang *et al.* [26] use the five convolutional layers in front of each pooling layer to extract useful information from the low-level contrast to high-level semantics for each pixel, and max-pool the hypercolumns to obtain the descriptors for each super-pixel. Wei *et al.* [27] select the deep descriptors with higher summed activation responses in an image, and concatenate aggregated feature maps with weights to construct useful deep descriptors. These findings provide a chance for us to take advantage of deep features in the region weighting.

C. LOGISTIC REGRESSION IN MULTIPLE-INSTANCE SETTING

Logistic regression is a discriminative probabilistic classifier that has achieved a great success in many fields, and has also been extended to multiple-instance setting. By assuming that all instances contribute equally or independently to a bag label, Xu and Frank [28] predict the bag label by simply averaging or maximizing individual instance probabilities.

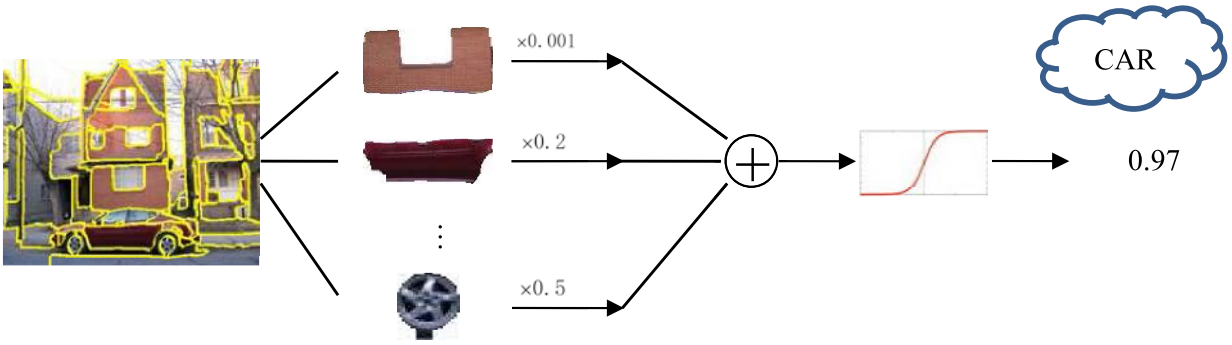


FIGURE 1. Illustration of logistic regression for region weighting. Given a weak label for an image, the regions in the image have different relevance to the semantic label. We impose a latent weight variable on each region, and assume the weighted combination of the regions triggers the semantic label. Spatial constraint is also considered to ensure the region smoothness.

Ray and Craven [11] generalize the Diverse Density [29] framework to design a multiple-instance logistic regression (MILR) algorithm. They first encode the multiple-instance assumption through the softmax combining function. This general setting allows any model that can learn class conditional probabilities in a supervised setting to be used in a multiple instance setting as well. Then adopt the logistic regression model to estimate conditional probabilities for each instance. The algorithm is competitive for text classification and CBIR tasks with other MIL algorithms. Settles *et al.* [12] extend the MILR to develop a multiple-instance active learning method. The works [5], [30] model the posterior probability for each instance using logistic sigmoid function, and combine them using noisy-OR for representing the bag probability. Furthermore, Raykar *et al.* [30] introduce a prior on the model parameter and develop a Bayesian MILR algorithm. Chen *et al.* [31] incorporate the LASSO approach into the proposed MILR and provide an efficient computer algorithm for variable selection and estimation.

III. PRELIMINARIES

A. LOGISTIC REGRESSION

We first review the supervised logistic regression for binary classification. Given a collection of training data $\{\mathrm{x}_1, \mathrm{x}_2, \cdots, \mathrm{x}_N\}$, $\mathrm{x}_i \in \mathbb{R}^{D\times 1}$ with labels $t_i \in \{0, 1\}$. For an instance x_i, we express its posterior probability of belonging to the positive class as a logistic sigmoid acting on a linear function of its feature vector:

$$y_i = \sigma(\mathrm{w}^\top \mathrm{x}_i) = \frac{1}{1+\exp(-\mathrm{w}^\top \mathrm{x}_i)}. \tag{1}$$

The posterior probability of belonging to the negative class is just $1 - y_i$. For the training data set with N instances, the likelihood function can be written as:

$$p(\mathrm{t}|\mathrm{w}) = \prod_{i=1}^{N} y_i = \prod_{i=1}^{N} y_i^{t_i}(1-y_i)^{1-t_i}, \tag{2}$$

where $\mathrm{t} = (t_1, t_2, \cdots, t_N)^\top$.

For computational convenience, an error function can be defined by taking the negative logarithm of the likelihood, which gives the cross entropy error function in the form:

$$\begin{aligned} E(\mathrm{w}) &= -\log p(\mathrm{t}|\mathrm{w}) \\ &= -\sum_{i=1}^{N}\Big(t_i \log(y_i) + (1-t_i)\log(1-y_i)\Big). \end{aligned} \tag{3}$$

Taking the gradient with respect to w, we obtain:

$$\nabla E(\mathrm{w}) = -\sum_{i=1}^{N}(t_i - y_i)\mathrm{x}_i. \tag{4}$$

where we have used

$$\frac{\partial y_i}{\partial \mathrm{w}} = y_i(1-y_i)\mathrm{x}_i. \tag{5}$$

Then the model parameter w can be estimated by iterative gradient descent $\mathrm{w}^{(\mathrm{new})} = \mathrm{w}^{(\mathrm{old})} - \rho\nabla E(\mathrm{w}^{(\mathrm{old})})$, where ρ is the learning rate that can be determined by line search, and the label of a new instance x is decided by the larger posterior probability.

B. QUADRATIC PROGRAMMING SOLUTION

One more efficient strategy to minimize the cross entropy loss is Newton-Raphson scheme [32]. Inspired by [32], we rewrite the Newton-Raphson update to derive a quadratic programming solution, which will play a key role in our region weighting optimization where there are constraints on the parameter.

The Newton-Raphson scheme uses a local quadratic approximation to the log likelihood function. It takes the form

$$\mathrm{w}^{(\mathrm{new})} = \mathrm{w}^{(\mathrm{old})} - H^{-1}(\mathrm{w}^{(\mathrm{old})})\nabla E(\mathrm{w}^{(\mathrm{old})}), \tag{6}$$

where H is the Hessian matrix that comprises the second derivative of the error function $E(\mathrm{w})$ (3) with respect to w.

$$\nabla E(\mathrm{w}^{(\mathrm{old})}) = -\sum_{i=1}^{N}(t_i - y_i)\mathrm{x}_i, \tag{7}$$

$$\begin{aligned} H(\mathrm{w}^{(\mathrm{old})}) &= \nabla\nabla E(\mathrm{w}^{(\mathrm{old})}) \\ &= \sum_{i=1}^{N}\mathrm{x}_i y_i(1-y_i)\mathrm{x}_i^\top \\ &= X\Lambda X^\top, \end{aligned} \tag{8}$$

where

$$X = (\mathrm{x}_1, \mathrm{x}_2, \cdots, \mathrm{x}_N), \tag{9}$$

and

$$\Lambda = diag\big(y_1(1-y_1), y_2(1-y_2), \cdots, y_N(1-y_N)\big). \tag{10}$$

Rewrite $\nabla E(\mathrm{w}^{(\mathrm{old})})$ of (7) to have

$$\begin{aligned}\nabla E(\mathrm{w}^{(\mathrm{old})}) &= -\sum_{i=1}^{N}(t_i - y_i)\mathrm{x}_i \\ &= -\sum_{i=1}^{N}\mathrm{x}_i y_i(1-y_i)\Big(\frac{(t_i-y_i)}{y_i(1-y_i)} + \mathrm{x}_i^\top \mathrm{w}^{(\mathrm{old})} - \mathrm{x}_i^\top \mathrm{w}^{(\mathrm{old})}\Big) \\ &= -X\Lambda(\mathrm{z} - X^\top \mathrm{w}^{(\mathrm{old})}) \end{aligned} \tag{11}$$

where z is a N-dimensional column vector with elements

$$z_i = \frac{(t_i - y_i)}{y_i(1-y_i)} + \mathrm{x}_i^\top \mathrm{w}^{(\mathrm{old})}. \tag{12}$$

Substituting (11) and (8) into (6), we have

$$\mathrm{w}^{(\mathrm{new})} = (X\Lambda X^\top)^{-1} X\Lambda \mathrm{z}. \tag{13}$$

Observing the form of (13), it is easy to know $\mathrm{w}^{(\mathrm{new})}$ is the solution to the following weighted least squares problem:

$$\min_{\mathrm{w}} \|(\Lambda^{\frac{1}{2}} X^\top)\mathrm{w} - \Lambda^{\frac{1}{2}}\mathrm{z}\|_2^2 \tag{14}$$

We can further rewrite it to a standard quadratic programming problem

$$\min_{\mathrm{w}} \ \frac{1}{2}\mathrm{w}^\top (X\Lambda X^\top)\mathrm{w} + (-X\Lambda \mathrm{z})^\top \mathrm{w}. \tag{15}$$

This reformulation makes the optimization feasible when the parameter is constrained, which we will see for our region weighting in Section IV-D.

IV. LOGISTIC REGRESSION FOR REGION WEIGHTING

We now formulate the region weighting intuition in the logistic regression framework, and propose our logistic regression region weighting (LRRW) method. Without loss of generality, we consider a collection of images $\{\mathcal{I}_1, \mathcal{I}_2, \cdots, \mathcal{I}_N\}$ with binary weak labels $t_i \in \{0, 1\}$ indicating whether they contain an object of interest. Using unsupervised over-segmentation methods, each image $\mathcal{I}_i$ is converted into a set of regions $\{\mathcal{R}_{i1}, \mathcal{R}_{i2}, \cdots, \mathcal{R}_{in_i}\}$, where n_i represents the number of regions in the image, and $\mathrm{x}_{ij} \in \mathbb{R}^{D\times 1}$ is the feature vector describing each region.

A. REGION WEIGHTING FORMULATION AND INTERPRETATION

Since different regions possess different intensity of relevance to the semantic label of the image, we impose a set of weight variables $\mathrm{s}_i = (s_{i1}, s_{i2} \cdots, s_{in_i})^\top$ on each region in $\mathcal{I}_i$ to reflect the intensity, where $|\mathrm{s}_i| = 1, \mathrm{s}_{ij} \geq 0, \forall j$. Suppose $X_i \in \mathbb{R}^{D\times n_i}$ is a matrix arranging the feature vectors as $X_i = (\mathrm{x}_{i1}, \mathrm{x}_{i2}, \cdots, \mathrm{x}_{in_i})$. We formulate the region weighting for image $\mathcal{I}_i$ as $X_i\mathrm{s}_i$, which is actually a weighted sum of the region feature vectors. Note that the mathematical result of the region combination $\Phi_i \mathrm{s}_i$ is still a D-dimensional column vector. s_i can be considered as hidden variables, and different assignment of s_i means selecting different combination of the regions to represent the semantic label of the image.

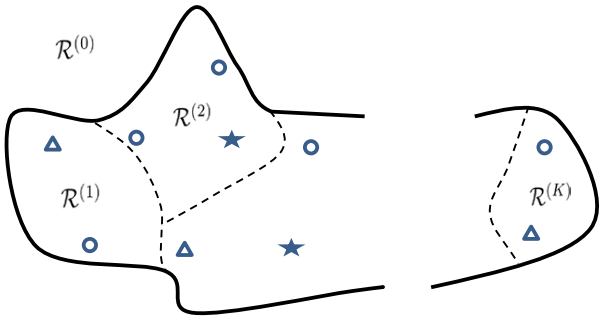


FIGURE 2. Illustration of region combination at feature-level. With normalized histogram feature representation, the feature vector for a larger image region is a weighted sum of feature vectors for its sub-regions, which is the basis of our region weighting formulation.

The region weighting has a theoretical interpretation of region combination at feature-level (in contrast to the score-level combination in traditional MILR). With the bag-of-words histogram feature representation, the feature vector of a larger region is the sum of the feature vectors for each sub-region therein. Suppose we already have a visual vocabulary $\{\mathrm{c}_1, \mathrm{c}_2, \ldots, \mathrm{c}_D\}$, which are cluster centers obtained by applying clustering to densely sampled training feature-points. Then for a region $\mathcal{R}^{(0)}$ containing M local feature-points, a histogram $\mathrm{h}^{(0)} = (h_1^{(0)}, h_2^{(0)}, \ldots, h_D^{(0)})^\top$ can be obtained as the feature representation, where $\{h_1^{(0)}, h_2^{(0)}, \ldots, h_D^{(0)}\}$ are the numbers of the M feature-points mapped to each visual word respectively, and obviously $\sum_{i=1}^{D} h_i^{(0)} = M$. If we divide the region into K sub-regions $\{\mathcal{R}^{(1)}, \mathcal{R}^{(2)}, \ldots, \mathcal{R}^{(K)}\}$ as shown in Fig. 2, then all of the feature-points will scatter into each region. By counting the number of feature-points similarly in each sub-region, we also have the histograms for each sub-region, $\mathrm{h}^{(k)} = (h_1^{(k)}, h_2^{(k)}, \ldots, h_D^{(k)})^\top$. Obviously $\mathrm{h}^{(0)} = \sum_{k=1}^{K} \mathrm{h}^{(k)}$, since the total numbers of the feature-points mapped to each visual word never change, $h_j^{(0)} = \sum_{k=1}^{K} h_j^{(k)}, \forall j$.

If we normalize a histogram representation h to get p by $p_i = \frac{h_i}{\sum_{j=1}^{D} h_j}$, then for the original region $\mathrm{p}^{(0)}$, $p_i^{(0)} = \frac{h_i^{(0)}}{\sum_{j=1}^{D} h_j^{(0)}}$, and for each sub-region $\mathrm{p}^{(k)}$, $p_i^{(k)} = \frac{h_i^{(k)}}{\sum_{j=1}^{D} h_j^{(k)}}$. The relation between the normalized histograms of the original region $\mathcal{R}^{(0)}$ and its sub-regions $\{\mathcal{R}^{(1)}, \mathcal{R}^{(2)}, \ldots, \mathcal{R}^{(K)}\}$ is

$$\begin{aligned} p_i^{(0)} &= \frac{h_i^{(0)}}{\sum_{j=1}^{D} h_j^{(0)}} \\ &= \frac{\sum_{k=1}^{K} h_i^{(k)}}{\sum_{j=1}^{D} h_j^{(0)}} \\ &= \sum_{k=1}^{K} \frac{\sum_{j=1}^{D} h_j^{(k)}}{\sum_{j=1}^{D} h_j^{(0)}} p_i^{(k)}. \end{aligned} \tag{16}$$

If we denote $m_k = \sum_{j=1}^{D} h_j^{(k)}$ representing the feature-points falling into the k-th sub-region, and $s_k = \frac{m_k}{M}$, then (16) can be rewritten as $p_i^{(0)} = \sum_{k=1}^{K} s_k p_i^{(k)}$. Consequently the relation between features of the original region $\mathcal{R}^0$ and its sub-regions $\{\mathcal{R}^{(1)}, \mathcal{R}^{(2)}, \ldots, \mathcal{R}^{(K)}\}$ can be expressed as:

$$\mathrm{p}^{(0)} = P\mathrm{s}, \tag{17}$$

where $P = (\mathrm{p}^{(1)}, \mathrm{p}^{(2)}, \cdots, \mathrm{p}^{(K)})$, and $\mathrm{s} = (s_1, s_2, \cdots, s_K)^\top$. Therefore, with the normalized histogram feature representation, the feature for a region is the weighted sum of its features for each sub-region, which coincides the assumption of our region weighting.

Difference from MILR: The MILR methods mentioned in the literature review use combining function to explicitly encode the MI assumption for a bag. If the model finds an instance likely to be positive, the output of the combining function should find its corresponding bag likely to be positive as well. Concretely, these MILR methods first model conditional probabilities for each instance using logistic sigmoid of a linear function of the feature vector, then combine the instance probabilities in a bag to represent the bag probability by strategies such as softmax and noisy-OR. When applied to segment-based object localization, these methods can be considered as region combination at score-level: They combine the conditional probability scores of the regions in an image using certain function to represent the image probability. Nevertheless, as is shown above, our region weighting is a region combination at feature-level. One of the benefits to do so is that we can easily integrate spatial constraint for region weights, which is important for object localization.

B. REGION WEIGHTING IN LOGISTIC REGRESSION

We integrate this region weighting formulation into the logistic regression framework. Suppose $\{\mathrm{s}_1, \mathrm{s}_2, \cdots, \mathrm{s}_N\}$ are hidden variables that can select the semantic region out of the background. We model the posterior probability of the object belonging to the semantic category as a logistic sigmoid function acting on a linear function of the region weighting:

$$y_i = \sigma(\mathrm{w}^\top X_i \mathrm{s}_i) = \frac{1}{1 + \exp(-\mathrm{w}^\top X_i \mathrm{s}_i)}. \tag{18}$$

In conventional LR, the bias for the linear combination is usually expressed implicitly for computational convenience. One can extend the feature vector by add one element $x_0 = 1$, then w_0 will play the role of a bias. In (18), this trick still works because we restrict $|\mathrm{s}_i| = 1$.

For an image/video set $\{\mathcal{I}_1, \mathcal{I}_2, \cdots, \mathcal{I}_N\}$ with weak labels $t_i \in \{0, 1\}$, the likelihood function in the form of region weighting can be written as

$$p(\mathrm{t}|\mathrm{w}, \mathrm{s}_1, \mathrm{s}_2, \cdots, \mathrm{s}_N) = \prod_{i=1}^{N} y_i = \prod_{i=1}^{N} y_i^{t_i}(1 - y_i)^{1-t_i}. \tag{19}$$

We then define an error function by taking the negative logarithm of the likelihood, which gives the cross entropy error function in the form:

$$\begin{aligned} &E(\mathrm{w}, \mathrm{s}_1, \cdots, \mathrm{s}_N) \\ &\quad = -\log p(\mathrm{t}|\mathrm{w}, \mathrm{s}_1, \mathrm{s}_2, \cdots, \mathrm{s}_N) \\ &\quad = -\sum_{i=1}^{N} \Big(t_i \log(y_i) + (1 - t_i) \log(1 - y_i) \Big). \end{aligned} \tag{20}$$

Note that (20) is different from (3) because y_i here is a function of w as well as s_i.

The objective function can be expressed as:

$$\begin{aligned} &\min_{\mathrm{w}, \mathrm{s}_1, \cdots, \mathrm{s}_N} E(\mathrm{w}, \mathrm{s}_1, \cdots, \mathrm{s}_N) \\ &\text{s.t. } |\mathrm{s}_i| = 1, \mathrm{s}_i \geq 0, \quad i = 1, \cdots, N. \end{aligned} \tag{21}$$

There are two types of unknown variables in (21), the weighting parameter and the model parameter. Moreover, the two types of unknown variables are coupled with each other and not jointly convex w.r.t. the objective function, thus it is difficult to get an analytical solution or optimize the two types of unknown variables simultaneously. We adopt the block coordinate descent approach to dealing with the coupled optimization problem. First we fix the weighting parameter s_i, $i = 1, 2, \cdots, N$, and update the model parameter w; Then fix the model parameter, and update the weighting parameter. Repeat the above two steps iteratively until the relative change of the objective function between two successive iterations is less than a predefined threshold.

C. UPDATE MODEL PARAMETER

When the weighting parameter s_i is fixed, the optimization of (21) becomes

$$\min_{\mathrm{w}} E(\mathrm{w}). \tag{22}$$

Taking the gradient with respect to w, we obtain:

$$\nabla E(\mathrm{w}) = -\sum_{i=1}^{N} (t_i - y_i) X_i \mathrm{s}_i, \tag{23}$$

where we have used

$$\frac{\partial y_i}{\partial \mathrm{w}} = y_i(1 - y_i) X_i \mathrm{s}_i. \tag{24}$$

This provides an iterative gradient descent direction, and we can use line search to determine the distance to move along this direction.

$$\mathrm{w}^{(\mathrm{new})} = \mathrm{w}^{(\mathrm{old})} - \rho \nabla E(\mathrm{w}^{(\mathrm{old})}). \tag{25}$$

D. UPDATE WEIGHTING PARAMETER

When the model parameter w is fixed, the objective function becomes:

$$\begin{aligned} &\min_{\mathrm{s}_1, \cdots, \mathrm{s}_N} E(\mathrm{s}_1, \cdots, \mathrm{s}_N) \\ &\text{s.t. } |\mathrm{s}_i| = 1, \mathrm{s}_i \geq 0, i = 1, \cdots, N. \end{aligned} \tag{26}$$

and this optimization can be equivalently decomposed into the following sub-optimization problems with $i = 1, 2, \cdots, N$:

$$\min_{\mathrm{s}_i} E_i(\mathrm{s}_i) \quad \text{s.t. } |\mathrm{s}_i| = 1, \mathrm{s}_i \geq 0. \tag{27}$$

where

$$E_i(\mathrm{s}_i) = -t_i \log(y_i) - (1 - t_i) \log(1 - y_i) \tag{28}$$

is a single term corresponding to the i-th image in the summation of (20).

The optimization of (27) would be similar to the original logistic regression since the symmetry of w and s_i in y_i (18), if there were not the constraints. However, the constraints makes the optimization not straightforward. Following the derivation in Section III-B, we apply the Newton-Raphson method to the update of the weight parameter s_i, and transform it to a standard quadratic programming.

Let us ignore the constraints in (27) for now. The Newton-Raphson update that uses quadratic approximation for minimizing a cross entropy error function of (28) should be

$$\mathrm{s}_i^{(\mathrm{new})} = \mathrm{s}_i^{(\mathrm{old})} - H^{-1}(\mathrm{s}_i^{(\mathrm{old})}) \nabla(\mathrm{s}_i^{(\mathrm{old})}). \tag{29}$$

It is easy to compute the gradient with respect to s_i

$$\nabla E_i(\mathrm{s}_i^{(\mathrm{old})}) = -(t_i - y_i) X_i^\top \mathrm{w}, \tag{30}$$

where we have used

$$\frac{\partial y_i}{\partial \mathrm{s}_i^{(\mathrm{old})}} = y_i(1 - y_i) X_i^\top \mathrm{w}, \tag{31}$$

and the Hessian matrix

$$H(\mathrm{s}_i^{(\mathrm{old})}) = \nabla\nabla E_i(\mathrm{s}_i^{(\mathrm{old})}) = X_i^\top \mathrm{w} y_i (1 - y_i) (X_i^\top \mathrm{w})^\top. \tag{32}$$

We now show how the Newton-Raphson update can be replaced by solving a quadratic programming, for the purpose of including the constraints into the optimization. Substituting (30) and (32) into (29), we have

$$\begin{aligned} \mathrm{s}_i^{(\mathrm{new})} &= \mathrm{s}_i^{(\mathrm{old})} - H^{-1}(\mathrm{s}_i^{(\mathrm{old})}) \nabla(\mathrm{s}_i^{(\mathrm{old})}) \\ &= \left((\Phi_i \alpha_i)^\top (\Phi_i \alpha_i) \right)^{-1} (\Phi_i \alpha_i)^\top (\lambda_i), \end{aligned} \tag{33}$$

where we have denoted

$$\Phi_i = \mathrm{w}^\top X_i, \tag{34}$$

$$\alpha_i = (y_i(1 - y_i))^{\frac{1}{2}}, \tag{35}$$

$$\lambda_i = \alpha_i (\Phi_i \mathrm{s}_i^{(\mathrm{old})} + \frac{1}{y_i}). \tag{36}$$

Therefore, $\mathrm{s}_i^{(\mathrm{new})}$ is just the solution of the following least squares:

$$\min_{\mathrm{s}_i} \|\Phi_i \alpha_i \mathrm{s}_i - \lambda_i\|_2^2 \tag{37}$$

We only do region weighting for the images/videos that indeed contain objects of interest. In other words, we update s_i only when $t_i = 1$, and keep s_i constant for $t_i = 0$.

Based on this conclusion and reconsidering the constraints, we can transform the optimization of (27) into an iterative quadratic programming:

$$\min_{\mathrm{s}_i} \frac{1}{2} \mathrm{s}_i^\top (\alpha_i^2 \Phi_i^\top \Phi_i) \mathrm{s}_i + (-\alpha_i \lambda_i \Phi_i^\top)^\top \mathrm{s}_i \quad \text{s.t. } |\mathrm{s}_i| = 1, \mathrm{s}_i \geq 0, \tag{38}$$

and it can be solved with a typical optimization package.

E. SPATIAL CONSTRAINT

The above strategy determines the region weights to select the best region combination that best corresponds to the semantic label, from the feature viewpoint. In a visual learning task, the region weights should be reconsidered from the viewpoint of spatial relation. The semantic region in an image usually corresponds to an object, and an object must be a set of connected regions. Therefore adjacent regions should be encouraged to have similar weight values because they are more likely to belong to an identical semantic category. To this end, we add a regularization term into the s_i optimization (38) to achieve spatial constraint.

We adopt the Laplacian prior term [26], [33], [34],

$$g_i(\mathrm{s}_i) = \mathrm{s}_i^\top L_i \mathrm{s}_i, \tag{39}$$

where L_i denotes the Laplacian matrix for the i-th image/video, and is computed as follows. We first compute the adjacency matrix A_i for the image/video by

$$(A_i)_{jk} = \begin{cases} \exp\left(-\frac{\|\mathrm{x}_{ij} - \mathrm{x}_{ik}\|_2^2}{2\sigma}\right), & \text{if } \mathcal{R}_{ij}\mathcal{R}_{ik} \text{ adjacent;} \\ 0, & \text{otherwise.} \end{cases} \tag{40}$$

σ is set to the mean squared Euclidean distance between instances in our experiments. Then the diagonal degree matrix is computed by summing up the columns of A_i

$$(E_i)_{jj} = \sum_{k=1}^{n_i} (A_i)_{kj}. \tag{41}$$

Then the Laplacian matrix is computed as

$$L_i = E_i - A_i. \tag{42}$$

The regularization term is included into the objective function of (38) to obtain:

$$\min_{\mathrm{s}_i} \frac{1}{2} \mathrm{s}_i^\top (\alpha_i^2 \Phi_i^\top \Phi_i + \gamma L_i) \mathrm{s}_i + (-\alpha_i \lambda_i \Phi_i^\top)^\top \mathrm{s}_i \quad \text{s.t. } |\mathrm{s}_i| = 1, \mathrm{s}_i \geq 0, \tag{43}$$

where γ is the hyper-parameter to control the relevant importance of the spatial constraint. We will show in the experiments that this regularization term plays an important role in the region weighting.

Algorithm 1 LRRW

Input: A set of images/videos $\{\mathcal{I}_1, \mathcal{I}_2, \cdots, \mathcal{I}_N\}$ with binary labels $t_i \in \{0, 1\}$; The hyper-parameters γ;
Output: Region weights $\{\mathrm{s}_1, \mathrm{s}_2, \cdots, \mathrm{s}_N\}$, and model parameter w.

```
Over-segment each image/video into regions;
Initialize model parameter w by training conventional LR using ambiguous instances;
for i=1:N do
    Initialize s_i = (1/n_i)1;
end
while not converged do
    Update w using (25);
    for i=1:N do
        if t_i==1 then
            Solve the quadratic programming (43) to obtain s_i;
        end
    end
end
Return {s_1, s_2, ..., s_N} and w.
```

F. ALGORITHM

Both w and $\{\mathrm{s}_1, \mathrm{s}_2, \cdots, \mathrm{s}_N\}$ need to be initialized to start the iterative optimization. For w initialization, we train a conventional LR using all the instances in the negative images/videos and all the instances in the positive images/videos as negative and positive training data respectively. For the weighting parameter, we initialize each region with equal values, i.e., $\mathrm{s}_i = \frac{1}{n_i}\mathbf{1}$. For positive images/videos that really contain objects of interest, we update the region weights in every iteration. For negative ones, we keep $\mathrm{s}_i = \frac{1}{n_i}\mathbf{1}$ throughout the process. The hyper-parameter γ is set to 1. The pseudo code is described in Algorithm 1. The block-coordinate descent algorithm is guaranteed to converge because each procedure does not increase the objective value.

With the iterative updates, the weights will concentrate to the semantic regions, and the LR model parameter w will be learned from the semantic regions independently of the background. As a result it is able to distinguish object regions and non-object regions. Consequently we use it to locate the objects in the positive images/videos by classifying regions using the following function:

$$sgn(\sigma(\mathrm{w}^\top \mathrm{x}) - 0.5). \tag{44}$$

The computational time is mainly spent on the two loops in the alternate optimization, which therefore can approximate the overall time complexity of the algorithm. In the inner loop, we need to update weighting parameter for each positive image/video. For the i-th image/video, the time complexity of quadratic programming is $O(n_i^3)$ if Interior Point method is used, hence the overall time for each inner loop is $O(n_+\overline{n}^3)$, where n_+ is the number of positive images, $\overline{n}$ is the average number of segments in positive images. In each outer loop, we still need to do gradient descent besides the inner loop. The time complexity of the gradient descent is $O(ND)$. Let us assume the average numbers of the inner loop and the outer loop are n_{in} and n_{out} respectively, then the overall time complexity can be approximated as $O\Big(n_{out}\big((n_{in}n_+\overline{n}^3) + ND\big)\Big)$.

V. EXPERIMENTS

Baselines. We compare our LRRW with methods that realize region detection: MILBoost [5], CRANE [3], SVM-RS [17], OBoW [4], and weakly supervised object localization methods LCL [35], CC [36], WSDNN [37] and MFMIL [10]. We consider MILBoost [5] as the representative of applying MILR to weakly supervised object detection. It uses logistic sigmoid to model conditional probability for instances, and combine them using noisy-OR for the bag probability. CRANE [3] extends negative mining to annotate the segments in weakly labelled videos, and is considered as a baseline for videos. SVM-RS [17] proposes a latent SVM to realize region selection in image classification, and OBoW [4] incorporates a learning distribution into markov random field to realize weakly supervised segment annotation. These two are considered as state-of-the-art region selection/detection methods. Since our region weighting leading to weakly supervised object localization (WSOL), we also compare with recently proposed WSOL methods [10], [35]–[37].

Datasets. We evaluate the performances on Pittsburgh Car(PittCar), PASCAL VOC 2007 and YouTube-Object(YTO) datasets. PittCar dataset is a relatively simple dataset for car detection, which we used to visualize and analysis the process of our LRRW. PittCar contains 400 images of street scenes. There are 200 images containing cars, and the others do not. The appearance of the cars in the images varies in shape, size, grayscale intensity and location. In addition, the cars occupy only a small portion of the images and may be partially occluded by other objects.

PASCAL VOC 2007 is a more convincing dataset, which consists of 9963 images. There are 20 object categories, with some images containing multiple objects. This dataset has been previously split into training and testing sets, which contained 5011 and 4952 images respectively. For irregular-shape object localization, it is better to use pixel-wise ground truth to calculate mean overlap for performance evaluation [1]. In [1], the authors manually annotate the 'cat' and 'dog' classes to generate pixel-wise ground truth. In order to evaluate our method on all of the classes, we use the subset with pixel-wise ground truth for evaluating segmentation tasks.

In order to evaluate the region weighting performance for videos, we also consider the (YTO) dataset used in [3]. YTO contains ten classes of videos collected from YouTube. Tang *et al.* [3] generated a groundtruthed set by manually annotating the segments for 151 selected shots. The segment-level ground truth is well suitable for the evaluation of irregular-shape object localization.

Set up and implementations We first need unsupervised over-segmentation methods to get exclusive regions

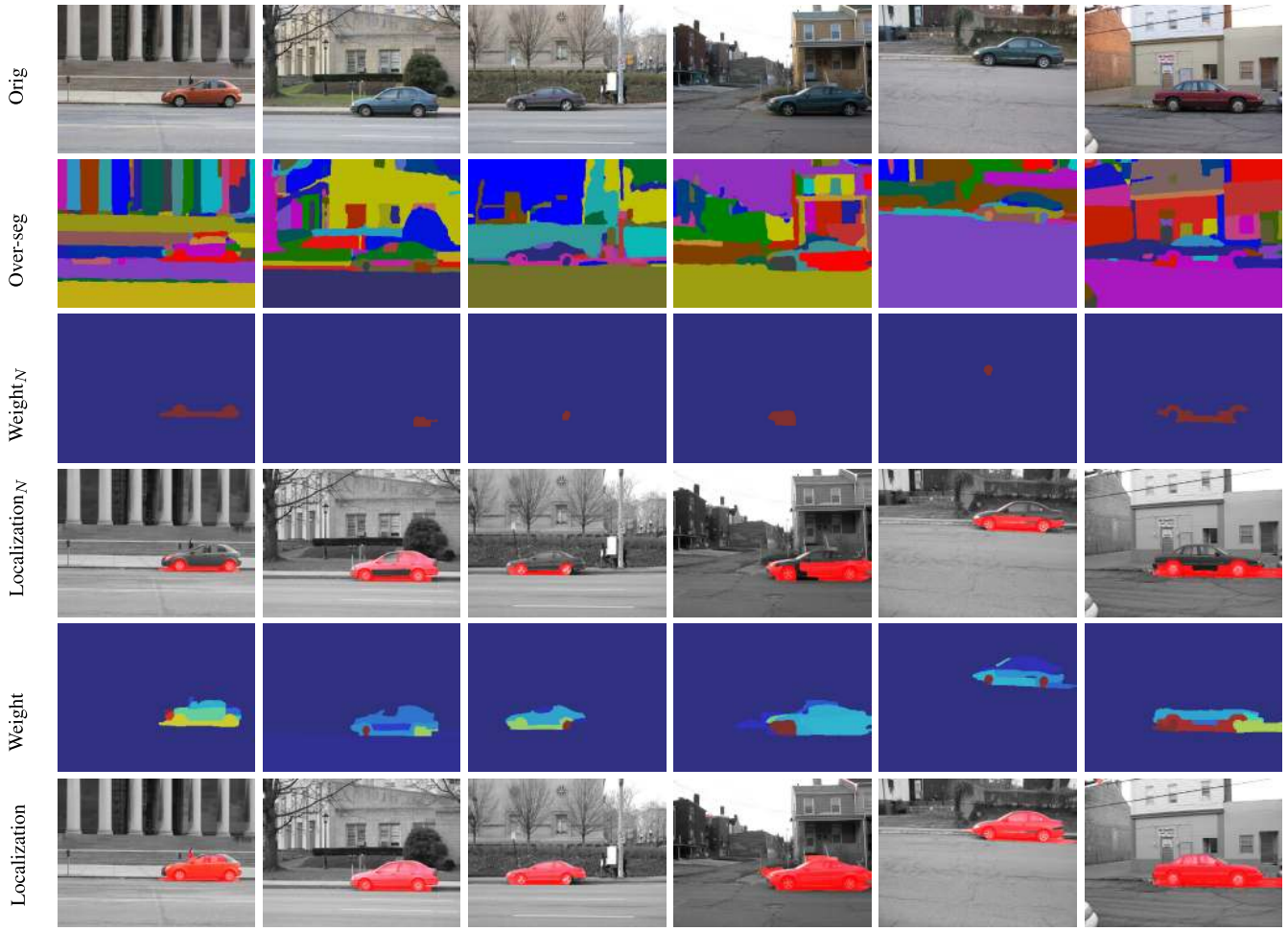


FIGURE 3. Visualization of LRRW results. Row 1 shows the original images. Row 2 visualizes the over-segmentation. Rows 3 and 4 plot the region weights and the predicted semantic regions without spatial constraints. Rows 5 and 6 plot the region weights and the predicted semantic regions with spatial constraints.

for images [38] and videos [39]. For each image/frame, we densely extract SIFT features [40] and apply clustering to a set of randomly selected training descriptors to obtain 1000 visual words for each dataset. Note that we ignore the temporal information in videos and process them similarly to images just as in [3]. Then for each segment, a 1000-dimensional histogram vector can be generated based on the vocabulary to describe the feature. Although the formulation of region combination at feature-level is derived under L_1 normalized histogram representation, in practice, we use L_2 normalization that has been widely proven much stronger in visual learning.

Results analysis. We visualize some results on PittCar in Fig. 3. The first row shows the original images, and the second row plots the over-segmentations, where different colors represent different segments. We plot the learned weighting parameters and the predicted labels for each region from LRRW without regularization in rows 3 and 4, and that with regularization in rows 5 and 6. The warmer color represents the larger weight values. The regions with predicted positive labels are stained with red.

It can be seen from the figure that the regularization term of spatial constraints plays an important role in the LRRW. Without the term, the weights tend to concentrate to sparse parts of the cars such as wheels, and the semantic localization tends to be incomplete. By contrast, if regularization is added, the region weights will get a trade-off between the attributive discrimination and the spatial smoothness, and the car localization is much better.

We use (44) to obtain irregular-shape object localization, and compare the performance with MILBoost [5], CRANE [3], SVM-RS [17], OBoW [4] on PittCar, PASCAL VOC 2007 and YTO datasets. The quantitative comparison is shown in Table 1. Following [41]–[43], we use average precision (AP), which is actually the ratio of truly detected objects to all the objects, to evaluate the performance. The LRRW outperforms MILBoost, the representative of applying MILR to object localization, because the MILRs learn the region classification parameter through relating the image/video probability to the regions' using noisy-OR/softmax, but do not really consider an object as a combination of regions. CRANE scores the ambiguous regions by their difference to

TABLE 1. Quantitative comparison with other region-based methods. For PittCar and YTO datasets, the performance is evaluated in terms of the mean average precision. For PASCAL dataset, the performance is evaluated in terms of the mean overlap.

Methods	Pitt	PASCAL VOC 2007 dataset									
		aero	bicy	bird	boat	bott	bus	car	cat	chai	cow
LRRW	**0.700**	**0.287**	**0.124**	**0.246**	0.106	**0.117**	0.376	**0.211**	0.356	**0.161**	**0.223**
SVM-RS[17]	0.525	0.130	0.084	0.134	**0.124**	0.105	0.355	0.197	**0.410**	0.106	0.173
OBoW[4]	0.465	0.172	0.092	0.190	0.108	0.093	**0.378**	0.165	0.298	0.115	0.176
CRANE [3]	0.205	0.117	0.071	0.143	0.101	0.093	0.298	0.107	0.282	0.105	0.170
MILBoost [5]	0.345	0.053	0.045	0.029	0.019	0.032	0.124	0.047	0.169	0.040	0.034

Methods	PASCAL VOC 2007 dataset										
	dini	dog	hors	moto	pers	pott	shee	sofa	trai	TV	ave
LRRW	0.208	**0.297**	**0.191**	0.385	0.268	**0.136**	**0.218**	0.146	**0.372**	**0.237**	**0.233**
SVM-RS[17]	0.198	0.172	0.159	**0.437**	**0.274**	0.079	0.210	0.177	0.324	0.191	0.202
OBoW[4]	**0.210**	0.182	0.106	0.266	0.203	0.072	0.204	**0.199**	0.357	0.120	0.185
CRANE [3]	0.187	0.140	0.123	0.181	0.184	0.065	0.186	0.154	0.284	0.132	0.156
MILBoost [5]	0.026	0.113	0.063	0.180	0.090	0.044	0.191	0.007	0.097	0.023	0.071

Methods	YouTube-Objects dataset										
	aero	bird	boat	car	cat	cow	dog	hors	moto	trai	ave
LRRW	**0.67**	**0.83**	**0.59**	0.75	0.56	**0.65**	**0.63**	**0.65**	**0.54**	**0.39**	**0.62**
SVM-RS[17]	0.44	0.67	0.47	0.75	0.44	0.60	**0.63**	0.53	0.36	**0.39**	0.53
OBoW[4]	0.22	0.50	0.35	**0.88**	**0.56**	**0.65**	0.56	0.41	0.27	0.33	0.47
CRANE [3]	0.00	0.33	0.35	0.75	0.33	0.45	0.26	0.24	0.27	0.17	0.32
MILBoost [5]	0.11	0.33	0.24	0.38	0.28	0.15	0.30	0.12	0.10	0.17	0.22

TABLE 2. Quantitative comparison for weakly supervised object localization.The performance is evaluated in terms of the mean average precision.

Methods	PASCAL VOC 2007 dataset									
	aero	bicy	bird	boat	bott	bus	car	cat	chai	cow
LRRW	0.654	0.630	0.523	0.469	0.289	0.612	0.742	0.446	0.235	0.667
MFMIL[10]	0.653	0.550	0.524	0.483	0.182	0.664	0.778	0.356	0.265	0.670
WSDNN[37]	0.651	0.634	0.597	0.459	0.385	0.694	0.770	0.507	0.301	0.688
CC[36]	0.664	0.593	0.427	0.204	0.213	0.634	0.743	0.596	0.211	0.582
LCL[35]	0.801	0.639	0.515	0.149	0.210	0.557	0.742	0.435	0.262	0.534

Methods	PASCAL VOC 2007 dataset										
	dini	dog	hors	moto	pers	pott	shee	sofa	trai	TV	ave
LRRW	0.452	0.480	0.642	0.688	0.259	0.405	0.680	0.532	0.668	0.620	0.535
MFMIL[10]	0.469	0.484	0.705	691	0.352	0.352	0.696	0.434	0.646	0.437	0.520
WSDNN[37]	0.340	0.373	0.610	0.829	0.251	0.429	0.792	0.594	0.682	0.641	0.561
CC[36]	0.140	0.385	0.495	0.600	0.198	0.392	0.417	0.301	0.502	0.441	0.437
LCL[35]	0.163	0.567	0.583	0.695	0.141	0.383	0.588	0.472	0.491	0.609	0.485

the definite negative ones, yet ignores the intra-class similarity and structural information of object regions, leading to severe misclassifications. LRRW is also better than SVM-RS and OBoW, resulting from the spatial constraints.

Deep features. The basis of our method is BoW histogram feature representation, however, it has been outperformed by DCNN especially in the field of object localization [15]. Therefore we also take advantage of deep features in LRRW to improve the performance, and compare it on PASCAL VOC 2007 with recently proposed weakly supervised object localization methods LCL [35], CC [36], WSDNN [37] and MFMIL [10].

Inspired by the recently developed pixel-wise DCNN representation [24], [25], we explore the deep feature from the VGG-NET [14] pre-trained on the ImageNet. The evaluation is executed on PASCAL VOC 2007 *trainval* set. All of the data are used for fine-tuning the pre-trained VGG model with bag-level labels.

For each image, we first resize it to 224 $\times$ 224 and feed it into the VGG model. We extract the feature maps of the convolutional layers in front of each pooling layer. Since the convolution and pooling operations in CNN reduce the spatial resolution, we up-sample each feature map to the scale of the original image. Each up-sampled feature map is considered as

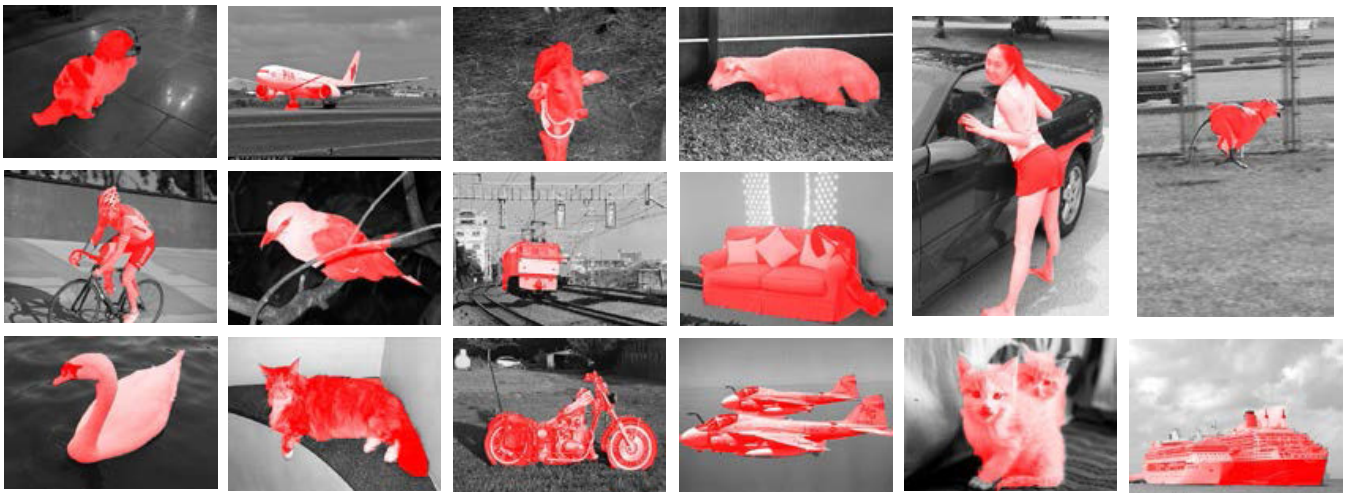


FIGURE 4. Visualization of some localization results using deep local features on PASCAL VOC 2007. The detected semantic regions are stained with red color.

a feature representation for pixels from low level to high level. We do L_2 normalization for each feature map and concatenate them with coefficients to construct a 1472-dimensional hyper-column for each pixel. Based on the findings in [27], we empirically set the concatenation coefficients for each layer to 1, 0.8, 0.5, 0.3 and 0.1, respectively. Then clustering is applied to the training points to obtain 1000 words, and therefore each segment can be represented by a 1000-bin histogram. L_2 normalization for histogram is used before feeding to the classifier.

We directly compare the irregular-shape object localization results with the bounding-box groundtruth. A localization is considered correct if its overlap with the ground truth is larger than 0.5. Then the percentage of the correctly detected objects is counted.

The quantitative comparison is listed in Table 2, and LRRW achieves comparable performance with state-of-the-art methods on average. Some visual localization results are shown in Fig. 4. The detected regions are stained with red color. The irregular-shape object localization is dependent on the over-segmentation quality. When the over-segmentation retains the boundaries, the localization tends to be satisfactory. Please note that our LRRW works in a binary classification setting. It performs object localization for each class separately as a one-versus-the-rest problem. When there are multiple categories in an image, they would be detected by different models. We visualize the single category localization in Fig. 4, where the person and the car cannot be detected simultaneously. One issue with LRRW is that it does not distinguish object individuals when there are multiple objects of identical category in one image, as shown in the last three images in Fig. 4.

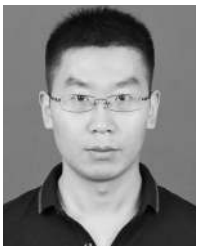
VI. CONCLUSION

In this paper, we focus on the visual region weighting problem when only weak labels are available. Since different visual regions possess different relevance to the semantic label, we propose to use weights to measure it. We also derive an interpretation of region combination at feature-level when BoW histogram feature representation used. LR is adopted as the base model and the region weights are incorporated into the cross entropy loss function. The region weights and the model parameter are optimized using block-coordinate descent algorithm. With the weights update, the LR model is trained with the semantic regions independently of the background, and therefore is able to distinguish object and non-object regions. The LRRW generates irregular-shape object localization, and overcomes the limitations of applying MIL to the task, resulting from its region combination assumption and spatial constraints. When taking advantage of deep local features, LRRW performs comparably to state-of-the-art methods. In future work we will explore region weighting in end-to-end deep neural networks.

REFERENCES

[1] S. Vijayanarasimhan and K. Grauman, "Efficient region search for object detection," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2011, pp. 1401–1408.
[2] C.-Y. Chen and K. Grauman, "Efficient activity detection with max-subgraph search," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2012, pp. 1274–1281.
[3] K. D. Tang, R. Sukthankar, J. Yagnik, and F.-F. Li, "Discriminative segment annotation in weakly labeled video," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2013, pp. 2483–2490.
[4] L. Wang, D. Meng, X. Hu, J. Lu, and J. Zhao, "Instance annotation via optimal BoW for weakly supervised object localization," *IEEE Trans. Cybern.*, vol. 47, no. 5, pp. 1313–1324, May 2017.
[5] P. A. Viola, J. C. Platt, and C. Zhang, "Multiple instance boosting for object detection," in *Proc. Adv. Neural Inf. Process. Syst.*, 2005, pp. 1–8.
[6] M. H. Nguyen, L. Torresani, L. de la Torre, and C. Rother, "Weakly supervised discriminative localization and classification: A joint learning process," in *Proc. IEEE Int. Conf. Comput. Vis.*, Sep. 2009, pp. 1925–1932.
[7] P. Siva, C. Russell, and T. Xiang, "In defence of negative mining for annotating weakly labelled data," in *Proc. Eur. Conf. Comput. Vis.*, 2012, pp. 594–608.
[8] R. G. Cinbis, J. J. Verbeek, and C. Schmid, "Multi-fold MIL training for weakly supervised object localization," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2014, pp. 2409–2416.
[9] J. Wu, Y. Yu, C. Huang, and K. Yu, "Deep multiple instance learning for image classification and auto-annotation," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Boston, MA, USA, Jun. 2015, pp. 3460–3469.

[10] R. G. Cinbis, J. Verbeek, and C. Schmid, "Weakly supervised object localization with multi-fold multiple instance learning," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 39, no. 1, pp. 189–203, Jan. 2017.
[11] S. Ray and M. Craven, "Supervised versus multiple instance learning: An empirical comparison," in *Proc. 2nd Int. Conf. Mach. Learn. (ICML)*, Bonn, Germany, Aug. 2005, pp. 697–704.
[12] B. Settles, M. Craven, and S. Ray, "Multiple-instance active learning," in *Proc. Adv. Neural Inf. Process. Syst.*, Vancouver, BC, Canada, vol. 20, Dec. 2007, pp. 1289–1296.
[13] H. O. Song, Y. J. Lee, S. Jegelka, and T. Darrell, "Weakly-supervised discovery of visual pattern configurations," in *Proc. Adv. Neural Inf. Process. Syst.*, 2014, pp. 1637–1645.
[14] K. Simonyan and A. Zisserman, "Very deep convolutional networks for large-scale image recognition," 2014, *arXiv:1409.1556*. [Online]. Available: https://arxiv.org/abs/1409.1556
[15] K. Chatfield, K. Simonyan, A. Vedaldi, and A. Zisserman, "Return of the devil in the details: Delving deep into convolutional nets," in *Proc. Brit. Mach. Vis. Conf. (BMVC)*, Nottingham, U.K., Sep. 2014, pp. 1–11.
[16] O. Yakhnenko, J. Verbeek, and C. Schmid, "Region-based image classification with a latent SVM model," INRIA, Paris, France, Tech. Rep. inria-00605344, 2011, pp. 1–13.
[17] J. Zhao, L. Wang, R. Cabral, and F. De la Torre, "Feature and region selection for visual learning," *IEEE Trans. Image Process.*, vol. 25, no. 3, pp. 1084–1094, Mar. 2016.
[18] J. Zhu, J. Mao, and A. L. Yuille, "Learning from weakly supervised data by the expectation loss SVM (e-SVM) algorithm," in *Proc. Adv. Neural Inf. Process. Syst.*, 2014, pp. 1125–1133.
[19] Z. Wei and M. Hoai, "Region ranking SVM for image classification," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Las Vegas, NV, USA, Jun. 2016, pp. 2987–2996.
[20] Z. Shi, T. M. Hospedales, and T. Xiang, "Bayesian joint modelling for object localisation in weakly labelled images," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 37, no. 10, pp. 1959–1972, Oct. 2015.
[21] A. Kolesnikov and C. H. Lampert, "Seed, expand and constrain: Three principles for weakly-supervised image segmentation," in *Proc. 14th Eur. Conf. Comput. Vis. (ECCV)*, Amsterdam, The Netherlands, Oct. 2016, pp. 695–711.
[22] M. Cimpoi, S. Maji, and A. Vedaldi, "Deep filter banks for texture recognition and segmentation," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Boston, MA, USA, Jun. 2015, pp. 3828–3836.
[23] A. Babenko and V. S. Lempitsky, "Aggregating local deep features for image retrieval," in *Proc. IEEE Int. Conf. Comput. Vis. (ICCV)*, Santiago, Chile, Dec. 2015, pp. 1269–1277.
[24] B. Hariharan, P. A. Arbeláez, R. B. Girshick, and J. Malik, "Hypercolumns for object segmentation and fine-grained localization," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Boston, MA, USA, Jun. 2015, pp. 447–456.
[25] C. Ma, J. Huang, X. Yang, and M. Yang, "Hierarchical convolutional features for visual tracking," in *Proc. IEEE Int. Conf. Comput. Vis. (ICCV)*, Santiago, Chile, Dec. 2015, pp. 3074–3082.
[26] D. Zhang, D. Meng, and J. Han, "Co-saliency detection via a self-paced multiple-instance learning framework," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 39, no. 5, pp. 865–878, May 2017.
[27] X.-S. Wei, J.-H. Luo, J. Wu, and Z.-H. Zhou, "Selective convolutional descriptor aggregation for fine-grained image retrieval," *IEEE Trans. Image Process.*, vol. 26, no. 6, pp. 2868–2881, Jun. 2017.
[28] X. Xu and E. Frank, "Logistic regression and boosting for labeled bags of instances," in *Proc. 8th Pacific-Asia Conf. Adv. Knowl. Discovery Data Mining (PAKDD)*, Sydney, NSW, Australia, May 2004, pp. 272–281.
[29] O. Maron and T. Lozano-Pérez, "A framework for multiple-instance learning," in *Proc. Adv. Neural Inf. Process. Syst.*, Denver, CO, USA, vol. 10, 1997, pp. 570–576.
[30] V. C. Raykar, B. Krishnapuram, J. Bi, M. Dundar, and R. B. Rao, "Bayesian multiple instance learning: Automatic feature selection and inductive transfer," in *Proc. 25th Int. Conf. Mach. Learn. (ICML)*, Helsinki, Finland, Jun. 2008, pp. 808–815.
[31] R.-B. Chen, K.-H. Cheng, S.-M. Chang, S.-L. Jeng, P.-Y. Chen, C.-H. Yang, and C.-C. Hsia, "Multiple-instance logistic regression with lasso penalty," 2016, *arXiv:1607.03615*. [Online]. Available: http://arxiv.org/abs/1607.03615
[32] S. Lee, H. Lee, P. Abbeel, and A. Y. Ng, "Efficient L1 regularized logistic regression," in *Proc. 21st Nat. Conf. Artif. Intell. 18th Innov. Appl. Artif. Intell. Conf.*, Boston, MA, USA, Jul. 2006, pp. 401–408.
[33] C. C. Aytekin, S. Kiranyaz, and M. Gabbouj, "Automatic object segmentation by quantum cuts," in *Proc. 22nd Int. Conf. Pattern Recognit. (ICPR)*, Stockholm, Sweden, Aug. 2014, pp. 112–117.
[34] K. Fu, I. Y. H. Gu, C. Gong, and J. Yang, "Robust manifold-preserving diffusion-based saliency detection by adaptive weight construction," *Neurocomputing*, vol. 175, pp. 336–347, Jan. 2016.
[35] C. Wang, W. Ren, K. Huang, and T. Tan, "Weakly supervised object localization with latent category learning," in *Proc. Eur. Conf. Comput. Vis.*, 2014, pp. 431–445.
[36] H. Bilen, M. Pedersoli, and T. Tuytelaars, "Weakly supervised object detection with convex clustering," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Boston, MA, USA, Jun. 2015, pp. 1081–1089.
[37] H. Bilen and A. Vedaldi, "Weakly supervised deep detection networks," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2016, pp. 2846–2854.
[38] P. Arbeláez, M. Maire, C. Fowlkes, and J. Malik, "Contour detection and hierarchical image segmentation," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 33, no. 5, pp. 898–916, May 2011.
[39] C. Xu, C. Xiong, and J. J. Corso, "Streaming hierarchical video segmentation," in *Proc. Eur. Conf. Comput. Vis.*, 2012, pp. 626–639.
[40] D. G. Lowe, "Distinctive image features from scale-invariant keypoints," *Int. J. Comput. Vis.*, vol. 60, no. 2, pp. 91–110, 2004.
[41] T. Deselaers, B. Alexe, and V. Ferrari, "Localizing objects while learning their appearance," in *Proc. Eur. Conf. Comput. Vis.*, 2010, pp. 452–466.
[42] M. Pandey and S. Lazebnik, "Scene recognition and weakly supervised object localization with deformable part-based models," in *Proc. IEEE Int. Conf. Comput. Vis.*, Nov. 2011, pp. 1307–1314.
[43] O. Russakovsky, Y. Lin, K. Yu, and F.-F. Li, "Object-centric spatial pooling for image classification," in *Proc. Eur. Conf. Comput. Vis.*, 2012, pp. 1–15.



LIANTAO WANG received the B.S. degree in mechanical engineering and the Ph.D. degree in pattern recognition and intelligent system from the Nanjing University of Science and Technology, Nanjing, China, in 2004 and 2015, respectively.

From 2012 to 2014, he was a Visiting Scholar with the Robotics Institute, Carnegie Mellon University, Pittsburgh, PA, USA. He is currently an Assistant Professor with the College of Internet of Things Engineering, Hohai University, Changzhou, China. His current research interest includes weakly supervised learning methods for object classification and localization.

TINGWEI WANG received the B.S. degree from the China University of Mining and Technology and the Ph.D. degree from the Nanjing University of Science and Technology, in 2002 and 2018, respectively.

He is currently an Assistant Professor with the School of Information Science and Engineering, University of Jinan. His research interests include computer vision and pattern recognition, with a focus on learning effective video representations for action understanding.

XUELEI HU received the B.E. degree in computer science and engineering from the Nanjing University of Science and Technology, Nanjing, China, the M.Phil. degree in pattern recognition and intelligence system, and the Ph.D. degree in computer science and engineering from the Chinese University of Hong Kong, Hong Kong.

From 2009 to 2010, she was a Visiting Postdoctoral Fellow with the Robotics Institute, Carnegie Mellon University, Pittsburgh, PA, USA. She is currently an Associate Professor with the School of Computer Science and Engineering, Nanjing University of Science and Technology, where she has been a Faculty Member, since 2005, and a Postdoctoral Research Fellow with the School of Information Technology and Electrical Engineering, The University of Queensland, Brisbane, QLD, Australia. Her current research interests include machine learning, ranging from theory to application, with a recent focus on developing novel machine learning, and data mining methods for real-world problems.