This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TPAMI.2016.2552172, IEEE Transactions on Pattern Analysis and Machine Intelligence

Learning from Weak and Noisy Labels for Semantic Segmentation

Zhiwu Lu, Zhenyong Fu, Tao Xiang, Peng Han, Liwei Wang, and Xin Gao

Abstract—A weakly supervised semantic segmentation (WSSS) method aims to learn a segmentation model from weak (image-level) as opposed to strong (pixel-level) labels. By avoiding the tedious pixel-level annotation process, it can exploit the unlimited supply of user-tagged images from media-sharing sites such as Flickr for large scale applications. However, these 'free' tags/labels are often noisy and few existing works address the problem of learning with both weak and noisy labels. In this work, we cast the WSSS problem into a label noise reduction problem. Specifically, after segmenting each image into a set of superpixels, the weak and potentially noisy image-level labels are propagated to the superpixel level resulting in highly noisy labels; the key to semantic segmentation is thus to identify and correct the superpixel noisy labels. To this end, a novel L_1-optimisation based sparse learning model is formulated to directly and explicitly detect noisy labels. To solve the L_1-optimisation problem, we further develop an efficient learning algorithm by introducing an intermediate labelling variable. Extensive experiments on three benchmark datasets show that our method yields state-of-the-art results given noise-free labels, whilst significantly outperforming the existing methods when the weak labels are also noisy.

Index Terms—Semantic segmentation, weakly supervised learning, label noise reduction, sparse learning

1 Introduction

Semantic image segmentation has long been the focus of computer vision research [1]. Given a set of labelled training images, the objective is to parse a test image into regions with a set of semantic labels (e.g. sky, roads, cars, and people). Early works have been dominated by fully supervised methods [2]–[8] which require pixel-level annotation during training. This tedious labelling process thus hinders those fully supervised methods from being applied to large scale problems. To overcome this limitation, recently weakly supervised approaches have become popular [9]–[19]. Taking a weakly supervised approach, each training image is annotated only at the image level. The annotation is weak in the sense that one only knows if an object class is present in the image but not where. This significantly reduces the annotation cost. More importantly, it is possible now to utilise the almost unlimited number of images on media-sharing sites such as Flickr. Such images have 'free' user-provided tags which can be used as image-level labels for model training. Nevertheless, these labels are much noisier than those obtained from annotators.

The weakly supervised semantic segmentation (WSSS) problem is challenging because weak labels lead to label noise. Specifically, most existing WSSS methods start with over-segmenting each training image into superpixels and assigning image-level labels to each superpixel. The initially assigned superpixel labels inevitably contain noise, which thus affects the learning of the subsequent segmentation model. This problem is further compounded when the image-level weak labels also contain noise – particularly when the user-provided tags (e.g. those from Flickr) are used as labels for model training [20], [21]. In addition, many existing methods [11], [12], [17]–[19] need to use predicted image-level labels for the test images as model input, which again are noisy due to imperfect prediction. Solving the WSSS problem thus boils down to how to effectively deal with the superpixel label noise which exists even if the image-level labels are clean.

Existing approaches to weakly supervised semantic segmentation only provide a partial and indirect solution to the label noise problem. Specifically, various object class appearance modelling and smoothing constraints are exploited to infer more accurate superpixel labels. However, their ability for noise reduction is severely hampered when the image-level labels are noisy to begin with (e.g. Flickr tags). This is because that without explicitly and directly addressing the superpixel label noise problem, existing methods have limited ability to deal with the larger amount of superpixel label noise propagated from image-level noisy labels. As a result, few works tackle the most challenging task of learning from both weak and noisy labels.

In this paper, we propose to cast the WSSS problem into a label noise reduction problem and develop a novel approach to explicitly identify and correct noisy labels. Specifically, given potentially noisy image-level labels, and a set of over-segmented superpixels from an image with inherited noisy labels from the image-level, we aim to infer the unknown true superpixel labels. To this end,

Z. Lu and P. Han are with the Beijing Key Laboratory of Big Data Management and Analysis Methods, School of Information, Renmin University of China, Beijing 100872, China. E-mail: zhiwu.lu@gmail.com.
Z. Fu and T. Xiang are with the School of Electronic Engineering and Computer Science, Queen Mary University of London, Mile End Road, London E1 4NS, United Kingdom. E-mail: t.xiang@qmul.ac.uk.
L. Wang is with the School of Electronics Engineering and Computer Science, Peking University, Beijing 100871, China.
X. Gao is with the King Abdullah University of Science and Technology (KAUST), Computational Bioscience Research Center (CBRC), CEMSE Division, Thuwal, 23955-6900, Saudi Arabia.

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TPAMI.2016.2552172, IEEE Transactions on Pattern Analysis and Machine Intelligence

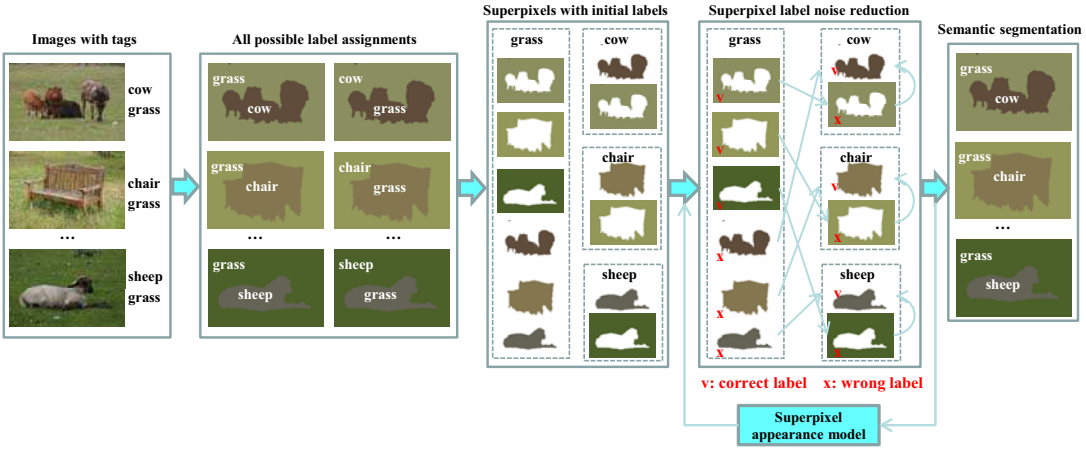


Fig. 1: The pipeline of our weakly supervised semantic segmentation model.

inspired by the successful use of sparse learning for noise reduction in other vision problems [22]–[24], a novel L_1-optimisation [25], [26] based sparse learning model is formulated. By modelling the label noise explicitly in the formulation and estimating the noise-free labels by solving an L_1-optimisation problem, this model aims to directly identify the noisy superpixel labels and correct them. However, solving the L_1-optimisation problem is non-trivial and conventional solvers are computationally expensive; to make them tractable, approximation often has to be made meaning performance is sacrificed in exchange for tractability. Our solution is to introduce an intermediate labelling variable in our formulation to drastically improve the efficiency of the model learning algorithm without compromising the performance. After addressing the superpixel label noise problem, the cleaned superpixel labels are used to learn superpixel appearance models for each label. These models are then deployed to perform semantic segmentation on unseen images without labels. The pipeline of our framework is illustrated in Figure 1.

Our contributions are as follows: 1) We tackle the semantic segmentation problem by learning from both weak and noisy labels, as opposed to most previous works which employ either strong, or weak but clean labels for semantic segmentation. 2) The WSSS problem is solved from a new perspective as a label noise reduction (or label denoising) problem. This direct and explicit approach to superpixel label noise detection and correction is superior to the existing indirect and implicit approaches in dealing with more severe label noise caused by the noisy image-level labels. 3) A novel L_1-optimisation based sparse learning model with an efficient algorithm is developed for label noise detection and correction. A theoretical analysis of this algorithm is also provided. 4) A new dataset is introduced which is based on the PASCAL VOC dataset but contains the original noisy user-provided tags from Flickr as weak labels. It is thus perfectly suited for studying the effects of weak and noisy labels for semantic segmentation. Extensive experiments on three benchmark datasets show that our method achieves state-of-the-art results given noise-free labels, whilst significantly outperforming the existing methods when the weak labels are also noisy.

2 Related Work

Semantic segmentation or scene parsing is one of the most widely studied computer vision problems [1]. Most semantic segmentation methods can be categorised into either fully supervised [2]–[8], [27]–[31] or weakly supervised [9]–[19], [32] ones depending on whether pixel-level or image-level labels are required for learning. In addition, a number of methods use labels that are weaker than pixel-level but stronger than image-level, ranging from a mixture of both [33] to a mixture of pixel-level and bounding-box-level labels [30]. Among various semantic segmentation methods, the weakly supervised ones are clearly more scalable and thus the focus of this paper.

Existing weakly supervised semantic segmentation (WSSS) methods employ a variety of models including latent topic model [9], conditional random field (CRF) [17], [19], linear SVM [18], label propagation [13], multi-instance learning [10], [11], clustering [14], and sparse reconstruction [15]. They also differ in whether superpixel-based object appearance is explicitly modelled and whether they operate under an inductive or transductive setting, or both. Despite these differences, all of them begin with decomposing each image into a set of superpixels with noisy labels inherited from image-level labels. This is followed by inferring the true (noise-free) superpixel labels. For this purpose various methods are developed but all are based on the same assumptions that visually similar superpixels across images should have same labels and vice versa. Similar to the existing methods, our method starts with noisy superpixel labels and exploits the same correlation between superpixel visual appearance similarity and label similarity. Nevertheless, there is a vital difference: our method explicitly models the superpixel label noise and thus has the potential to be more robust against higher level of label noise. This leads to another difference, that is, few existing semantic segmentation works consider the more realistic yet more challenging setting of learning from both weak and noisy image-level labels due to their limited ability to deal with label noise.

As far as we know, the only existing work that targets at learning a semantic segmentation model from both weak and noisy labels is the study of Zhang et al [19]. Although

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TPAMI.2016.2552172, IEEE Transactions on Pattern Analysis and Machine Intelligence

the problem identified in [19] is new, the proposed solution is based on conditional random field which has been exploited previously in both supervised [34] and weakly supervised [17] segmentation methods. To deal with the more severe superpixel label noise problem caused by noisy image-level labels, contextual information is exploited in the form of inter-label correlation and co-occurrence statistics. However, without explicitly modelling the label noise and solving it as a denoising problem, their performance is much weaker than that of ours as demonstrated in our experiments (see Section 5). Another existing work that is worth mentioning here is that of Xu et al [18]. Formulating the WSSS problem as a large-margin clustering method, this work is related to ours in that both methods iteratively estimate the superpixel labels and update superpixel appearance models[1]. Despite its simple formulation, its performance on various benchmark datasets is the state-of-the-art among existing WSSS methods published so far. Our extensive comparative experiments show that our method's performance is marginally better overall than theirs given noise-free image-level labels whilst being significantly better when the image-level labels contain noise.

It is noted that as in most other computer vision areas, deep learning models, particularly deep convolutional neural networks (CNNs) [35] start to gain popularity in semantic segmentation. Most efforts are focused on the supervised setting. These include a number of earlier works that adapt the models learned for the ImageNet classification tasks to detection and segmentation [36]–[38]. These approaches rely on preprocessing steps such as superpixels or object proposals and post-processing based on random fields or local appearance classifiers for pixel label refinements. More recently state-of-the-art semantic segmentation results are achieved by an end-to-end, and pixels-to-pixels trained CNN model called fully convolutional network (FCN) [31] which does not need those pre- and post- processing steps. The most relevant works along this line of research are [32], [39]. These CNN-based WSSS methods either directly learn a FCN model for semantic segmentation, or first learn object classifiers and then learn the weights of pixels within each image with respect to the object classifiers to obtain the final pixel-level segmentation. The reported results are clearly higher than any non-CNN based WSSS methods on the VOC datasets. Note that in order to learn the CNN model and avoid overfitting, they use 700,000 ImageNet images, each containing one of the 20 object classes in VOC, and 60,000 background images. In this paper, although we also focus on semantic segmentation with the pre-trained ImageNet features, we additionally show that it is possible to obtain good results with a superpixel appearance model based on CNNs without any pre-training or additional data apart from a handful (100s/1000s) of weakly labelled training images.

Our work on semantic segmentation is closely related to multi-class image co-segmentation [40]–[42], which aims to segment salient objects from a set of weakly-labelled images. The key difference is that the main objective of our model is to segment a set of test images without any labels rather than segmenting the weakly-labelled training images. Moreover, our work is also related to image tagging works such as [20], [21], [43] which learn a model from noisy user-provided image-level labels as well. However, the problem tackled here is harder as we aim to simultaneously segment images and label each segment, rather than labelling the whole images.

Beyond specific computer vision problems, learning from noisily labelled data has been studied extensively in machine learning. A comprehensive review of this field can be found at [44]. Among the existing noise-robust models and noise-cleaning algorithms, the sparse learning based ones [20], [26] are the most related which have been successfully applied to other computer vision problems [22]–[24]. Apart from formulating the sparse learning model specifically for WSSS with a problem-specific constraint in the cost function, another novelty in our method is the introduction of an intermediate labelling variable enabling the development of a more efficient as well as effective algorithm to solve the L_1-optimisation problem. Our experimental results suggest that this new formulation and optimisation algorithm lead to improvements (on both performance and speed) over a number of conventional sparse learning based noise reduction models [20], [26], as well as our own formulation without the intermediate labelling variable.

3 Superpixel Label Noise Reduction

We first introduce our sparse learning model for superpixel label noise reduction, which is the most important step in our pipeline (see Figure 1). The rest of the pipeline will be described in Section 4.

3.1 Problem Definition

Given a set of training images with weak image-level labels and a set of unlabelled test images, we assume that each training and test image has been over-segmented into a set of superpixels. Each superpixel is assigned an object class label based on the potentially noisy image-level labels; these superpixel labels thus inevitably contain large amount of noise. The objective of superpixel label noise reduction is to identify and correct the noisy superpixel labels.

Formally, we are given a large set of superpixels $\mathcal{X} = \{x_1, ..., x_N\}$ and their initial labels $Y = \{y_{ij} : y_{ij} \in \{0, 1\}\}_{N \times C}$, where N is the total number of superpixels and C is the number of object categories. These superpixels are represented as feature vectors x_i $(i = 1, ..., N)$ which capture the visual appearance of each superpixel region (see more details in Section 4.1), while the initial superpixel labels $\{y_{i1}, y_{i2}, ..., y_{iC}\}$ of them are inferred from the image-level labels. As illustrated in Figure 2, the initial superpixel labels $Y = \{y_{ij}\}_{N \times C}$ are estimated as: $y_{ij} = 1$ if the superpixel x_i belongs to an image which is labelled with category j, and $y_{ij} = 0$ otherwise.

Note that the initial superpixel labels Y cannot be accurately estimated by such a simple inference method.

1. This similarity means that our model can also utilise various forms of weak supervision as theirs does.

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TPAMI.2016.2552172, IEEE Transactions on Pattern Analysis and Machine Intelligence

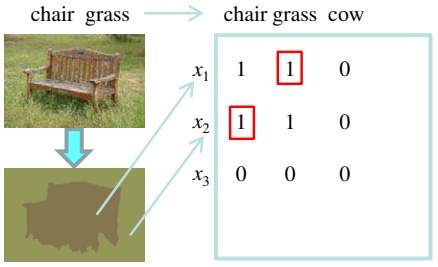


Fig. 2: Superpixel initial label assignment from image-level labels. The wrongly assigned labels are marked in red.

The noise issue becomes even severer when the image-level labels are noisy to begin with, e.g. Flickr user-provided tags. To address this noise issue, we will formulate a superpixel label noise reduction model in the following.

3.2 Formulation

Our main goal is estimating the unknown true labels of the superpixel set, denoted as $\hat{Y} \in \mathcal{R}^{N\times C}$, by reducing the noise in Y. Before formulating such noise reduction problem, we first introduce two constraints for inferring $\hat{Y}$. In particular, to reduce the noise in Y as much as possible, we only consider L_1-norm constraints in our objective function, because L_1-norm constraints on latent label variables typically lead to better robustness against noise compared to their L_2-norm counterparts.

The first constraint is a *visual similarity constraint*, which is based on the assumption that visually similar superpixels should have same labels and vice versa. This constraint is explored in various forms in all existing weakly supervised semantic segmentation works. To formulate this constraint, we first construct a graph $\mathcal{G} = \{\mathcal{V}, W\}$ with its vertex set $\mathcal{V} = \mathcal{X}$ and weight matrix $W = \{w_{ij}\}_{N\times N}$, where w_{ij} denotes the similarity between superpixel feature vectors x_i and x_j, computed using a Gaussian heat kernel. The normalised Laplacian matrix $\mathcal{L}$ of $\mathcal{G}$ is given by

$$\mathcal{L} = I - D^{-\frac{1}{2}} W D^{-\frac{1}{2}}, \tag{1}$$

where I is an $N \times N$ identity matrix, and D is an $N \times N$ diagonal matrix with its i-th diagonal element being $\sum_j w_{ij}$. We derive a new matrix $B \in \mathcal{R}^{N\times N}$ from $\mathcal{L}$:

$$B = \Sigma^{\frac{1}{2}} V^T, \tag{2}$$

where V is an $N \times N$ orthonormal matrix with each column being an eigenvector of $\mathcal{L}$, and Σ is an $N \times N$ diagonal matrix with its diagonal element Σ_{ii} being an eigenvalue of $\mathcal{L}$ (sorted as $0 \leq \Sigma_{11} \leq ... \leq \Sigma_{NN}$). Denoting the eigen-decomposition of $\mathcal{L}$ as $\mathcal{L} = V\Sigma V^T$, $\mathcal{L}$ can be represented in a symmetrical decomposition form:

$$\mathcal{L} = (\Sigma^{\frac{1}{2}} V^T)^T \Sigma^{\frac{1}{2}} V^T = B^T B. \tag{3}$$

Given the matrix B, the visual similarity constraint becomes a graph smoothness constraint with which the optimal $\hat{Y}$ will minimise $||B\hat{Y}||_1$, that is, label similarity needs to agree with visual similarity. This constraint is closely related to graph Laplacian regularisation [45], [46]. Note that the conventional graph Laplacian terms is written as a trace norm term. Here L_1-norm is used to promote sparsity on the inferred label (because by default each superpixel should only have one label), as well as to suppress the negative impacts of outlying superpixels.

The second constraint is a *noise sparsity constraint* which enforces noise sparsity in Y. It can be formulated as a L_1-norm sparsity regularisation term $||\hat{Y} - Y||_1$. This is a commonly used constraint for data noise [24], [47], and has been proven to be effective even when the data noise is not sparse, e.g. over 50% of the data are corrupted by noise.

Combining these two constraints, the superpixel label noise reduction problem becomes the optimisation problem:

$$\min_{\hat{Y}\geq 0} ||B\hat{Y}||_1 + \gamma||\hat{Y} - Y||_1, \tag{4}$$

where γ is the weight balancing the two constraints.

Note that if our superpixel label noise reduction is considered as a compressed sensing process, the graph smoothness constraint $||B\hat{Y}||_1$ will induces sparsity in the compressed domain spanned by the eigenvectors of $\mathcal{L}$ (see more discussions in Section 3.3). Therefore our formulation induces not only the smoothness sparsity $||B\hat{Y}||_1$ in the compressed domain but also the noise sparsity $||\hat{Y} - Y||_1$ in the original space for superpixel label noise reduction.

3.3 An Efficient Optimisation Algorithm

Although the L_1-optimisation problem in Eq. (4) looks rather simple, its solution is anything but. This is because both constraints in Eq. (4) are L_1-norm terms and solving such a cost function is notoriously hard [48]. In particular, to make the solver tractable, approximation has to be made which means sacrifice in the learning performance.

To overcome this problem, our strategy is to introduce additional non-L_1-norm terms in the formulation. Specifically, inspired by [49] we introduce an intermediate labelling variable $F \in \mathcal{R}^{N\times C}$ as an auxiliary variable for $\hat{Y}$ in our formulation. With this variable, we avoid solving a 'pure' L_1-norm optimisation problem, which makes it possible to develop more efficient solvers. After introducing F, the original problem in Eq. (4) becomes:

$$\min_{\hat{Y}\geq 0, F} \frac{\lambda}{2}||\hat{Y} - F||_F^2 + ||BF||_1 + \gamma||\hat{Y} - Y||_1, \tag{5}$$

which is equivalent to Eq. (4) when $\lambda \to +\infty$. To be consistent with the standard formulation of sparse learning, we rewrite Eq. (5) to the following equivalent form:

$$\min_{\hat{Y}\geq 0, F} \frac{1}{2}||\hat{Y} - F||_F^2 + \lambda||BF||_1 + \gamma||\hat{Y} - Y||_1. \tag{6}$$

Note that since the auxiliary variable F has the same meaning as $\hat{Y}$, their relationship is enforced using a Frobenius norm term. Now the cost function contains more than just L_1-norm terms, and the L_1-optimisation problem in Eq. (6) can be solved by the following alternate optimisation steps each containing an easier subproblem:

$$F^* = \arg\min_F \frac{1}{2}||F - \hat{Y}^*||_F^2 + \lambda||BF||_1, \tag{7}$$

$$\hat{Y}^* = \arg\min_{\hat{Y}\geq 0} \frac{1}{2}||\hat{Y} - F^*||_F^2 + \gamma||\hat{Y} - Y||_1, \tag{8}$$

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TPAMI.2016.2552172, IEEE Transactions on Pattern Analysis and Machine Intelligence

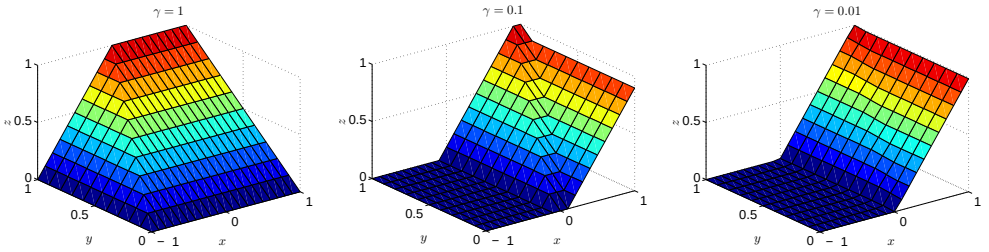


Fig. 3: Typical examples of the soft-thresholding function $z = \text{soft\_thr}(x, y, \gamma)$. Here, γ is set to 1, 0.1, or 0.01.

where $\hat{Y}^* = Y$ initially.

As a basic L_1-optimisation problem, the second subproblem in Eq. (8) has a closed-form and exact solution:

$$\hat{Y}^* = \text{soft\_thr}(F^*, Y, \gamma), \tag{9}$$

where $\text{soft\_thr}(\cdot, \cdot, \gamma)$ is a soft-thresholding function. Here, we define $z = \text{soft\_thr}(x, y, \gamma)$ as:

$$z = \begin{cases} z_1 = \max(x - \gamma, y), & f_1 \leq f_2 \\ z_2 = \max(0, \min(x + \gamma, y)), & f_1 > f_2 \end{cases}, \tag{10}$$

where $f_1 = \frac{1}{2}(z_1 - x)^2 + \gamma|z_1 - y|$ and $f_2 = \frac{1}{2}(z_2 - x)^2 + \gamma|z_2 - y|$. This piecewise function can be obtained by solving an optimisation problem that derives from Eq. (8): $z = \arg\min_{\hat{y} \geq 0} \frac{1}{2}(\hat{y} - x)^2 + \gamma|\hat{y} - y|$. The key lies in how to remove the operator $|\cdot|$ from the objective function. Some typical examples are shown in Figure 3.

In the following, we focus on developing an efficient algorithm to solve the first L_1-optimisation subproblem in Eq. (7) which is trickier than the second subproblem. In particular, directly solving the first L_1-optimisation subproblem is computationally intractable primarily due to the dimension of the B matrix ($N \times N$ where N is the number of superpixels in a training set), which is derived from $\mathcal{L}$ (Eq. (2)). Fortunately, the dimension of our superpixel label noise reduction can be reduced dramatically by working only with a small subset of eigenvectors of $\mathcal{L}$. Specifically, similar to [50], we significantly reduce the dimension of F by decomposing it to $F = V_m A$, where $A = \{a_{ij}\}_{m \times C}$ is an $m \times C$ matrix that collects the reconstruction coefficients and V_m is an $N \times m$ matrix whose columns are the m eigenvectors with the smallest eigenvalues (i.e. the first m columns of V). The first L_1-optimisation subproblem in Eq. (7) can now be formulated as follows:

$$\begin{aligned} &\arg\min_A \frac{1}{2}||V_m A - \hat{Y}^*||_F^2 + \lambda||BV_m A||_1 \\ &= \arg\min_A \sum_{j=1}^{C} \frac{1}{2}||V_m A_{\cdot j} - \hat{Y}_{\cdot j}^*||_2^2 + \lambda||BV_m A_{\cdot j}||_1, \end{aligned} \tag{11}$$

where $\hat{Y}_{\cdot j}^*$ and $A_{\cdot j}$ denote the j-th column of $\hat{Y}^*$ and A, respectively. We prove in Appendix 1 that solving the smaller-scale problem in Eq. (11) is equivalent to solving the original intractable problem in Eq. (7) under an easy condition. The above L_1-optimisation problem can be further decomposed into the following C independent L_1-optimisation subproblems:

$$\begin{aligned} &\arg\min_{A_{\cdot j}} \frac{1}{2}||V_m A_{\cdot j} - \hat{Y}_{\cdot j}^*||_2^2 + \lambda||BV_m A_{\cdot j}||_1 \\ &= \arg\min_{A_{\cdot j}} \frac{1}{2}||V_m A_{\cdot j} - \hat{Y}_{\cdot j}^*||_2^2 + \lambda||\sum_{i=1}^{m} \Sigma^{\frac{1}{2}} V^T V_{\cdot i} a_{ij}||_1 \\ &= \arg\min_{A_{\cdot j}} \frac{1}{2}||V_m A_{\cdot j} - \hat{Y}_{\cdot j}^*||_2^2 + \lambda \sum_{i=1}^{m} \Sigma_{ii}^{\frac{1}{2}}|a_{ij}|, \end{aligned} \tag{12}$$

where the orthonormality of V is used to simplify the L_1-norm term $||BV_m A_{\cdot j}||_1$. The first term in Eq. (12) denotes the linear reconstruction error just as that in the standard sparse coding formulation [25], while the second term denotes the weighted L_1-norm sparsity regularisation over the reconstruction coefficients. That is, the first L_1-optimisation problem in Eq. (12) is transformed into a generalised sparse coding problem. Many off-the-shelf solvers exist; in this paper, the L1General toolbox[2] is employed.

The decomposition $F_{\cdot j} = V_m A_{\cdot j}$ ($m \ll N$) used in Eq. (12) has two distinct advantages. Firstly, we can transform the original L_1-optimisation problem in Eq. (7) into a generalised sparse coding problem, which can then be solved at a linear cost with respect to N. Secondly, we do not need to compute the full matrix B (with a large time cost), which is especially beneficial for our problem of semantic segmentation where a large set of superpixels are used to compute B. To further improve the computational efficiency, we choose to compute the Laplacian matrix $\mathcal{L}$ over a k-nearest neighbour (k-NN) graph. Given a k-NN graph ($k \ll N$), finding m smallest eigenvectors of the sparse matrix $\mathcal{L}$ has a time complexity of $O(m^3 + m^2 N + kmN)$, which scales well to the data. In addition, to cope with extremely large data ($N > 1$M), we also provide a much more efficient approach to finding m smallest eigenvectors of $\mathcal{L}$ in the supplementary material.

The complete algorithm for superpixel label noise reduction is outlined in Algorithm 1. Note that although each superpixel is assumed to belong to a single object category, we only take soft label assignment into account during the iterative optimisation process and only enforce the single label constraint in the final step (Step 7). In practice, it is noted that the L_1-norm sparsity constraints in Eq. (4) would encourage the inferred labels for each superpixel to be as sparse as possible. This means that it is not necessary to enforce the single-label-per-superpixel constraint explicitly in each iteration of the optimisation

2. http://www.cs.ubc.ca/~schmidtm/Software/L1General.html

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TPAMI.2016.2552172, IEEE Transactions on Pattern Analysis and Machine Intelligence

Algorithm 1: Superpixel Label Noise Reduction

Input: Superpixels $\mathcal{X} = \{x_1, ..., x_N\}$
Initial labels of superpixels Y
Parameters k, m, λ, γ.
Output: Labels of superpixels
1. Construct a k-NN graph with its weight matrix W being defined over all the superpixels $\mathcal{X}$;
2. Compute the normalised Laplacian matrix $\mathcal{L}$ (Eq. (1));
3. Find m smallest eigenvectors of $\mathcal{L}$ and store them in V_m;
4. Initialise $\hat{Y}^*$ as $\hat{Y}^* = Y$;
while *a stopping criterion is not met* **do**
  5. Find the best solution A^* of the L_1-optimisation problem in Eq. (11) using the solver in L1General;
  6. Compute $F^* = V_m A^*$ and update $\hat{Y}^*$ with Eq. (9);
end
7. With the estimated $\hat{Y}^* = \{\hat{y}^*_{ij}\}_{N \times C}$, label each superpixel x_i with object category $\arg\max_j \hat{y}^*_{ij}$.

algorithm. In our experiments, we found that the proposed algorithm not only converges to a solution (i.e. A^*) as sparse as possible, but also converges within very limited number of iterations (< 5).

4 Semantic Segmentation with Weak and Noisy Labels

In this section, the other steps in our pipeline (Figure 1) for weakly supervised semantic segmentation are detailed.

4.1 Superpixel Segmentation and Representation

For fair comparison, we follow the superpixel segmentation and feature extraction method in [18]. Specifically, for each image, we compute the Ultrametric Contour Map (UCM) [51], and threshold it at 0.4 to extract superpixels. Since UCM tends to produce superpixels of very small sizes, we thus adopt a local search algorithm [52] to merge adjacent tiny superpixels. Moreover, for each obtained superpixel, visual features are extracted to represent its appearance. In particular, first a bounding box is fit to each superpixel; R-CNN [53] features are then extracted within the bounding box as well as within the superpixel region. These two sets of features (8,192-dimensional in total) thus capture the local context and the shape of the superpixel. To also capture the global context and superpixel size/location in the image, we enlarge the bounding box to cover the whole image and compute another two sets of R-CNN features. That is, R-CNN features are extracted from the whole image as well as an image of the same size which is filled with the ImageNet mean image except for the superpixel region. By concatenating the four sets of R-CNN features, we obtain a 16,384-dimensional feature vector for each superpixel. This feature representation will be deployed to compute the Laplacian matrix $\mathcal{L}$ (Eq. (1)) as well as act as input to a superpixel appearance model to be detailed later.

4.2 Superpixel Appearance Modelling and Iterative Label Refinement

Given a training set of weakly labelled images, after superpixel decomposition, initial label assignment and applying the proposed superpixel noise reduction method in Algorithm 1, we can now obtain a semantic segmentation of the training images. After that, a superpixel appearance model is learned for two purposes: to iteratively refine the semantic segmentation, and to predict superpixel labels of (i.e. to segment) an unseen test image.

The powerful R-CNN features used for superpixel representation makes it an easy task for appearance model selection. We simply learn C linear one-versus-all SVMs for the C object categories given the superpixel labels inferred by Algorithm 1. The learned model is then used to predict a new set of superpixel labels for the training images. These labels enforce the superpixel appearance consistency for each object category globally across the whole training set, thus having the potential to further reduce the superpixel label noise. Therefore, to improve the performance of semantic segmentation, the superpixel labels are iteratively refined. That is, we feed the predicted labels into Algorithm 1 for further noise reduction. With the 'cleaned' labels, a new set of appearance models are learned. This iterative superpixel label prediction/ appearance model updating procedure has been widely adopted in existing WSSS methods [14], [17], [18]. It is noted that if the superpixel labels are treated as latent variables, this procedure is similar in spirit to the EM (Expectation-Maximisation) algorithm.

4.3 An Alternative Appearance Model

The R-CNN model deployed for superpixel feature extraction is learned using millions of images from ImageNet [53]. Although this is different from the deep learning model deployed in [32] which uses additional images of only the targeted object categories, one still wonders how much the final segmentation performance benefits from the additional images used for learning the R-CNN model. In other words, how well can our method perform when learned from the weakly labelled training data only? To answer this question, an alternative appearance model is considered in the following.

First of all, instead of the powerful R-CNN features, a conventional hand-crafted visual feature representation is employed for our superpixel label noise reduction model (Algorithm 1) to produce the initial labels of superpixels. These labels are then used as the training labels for an appearance model to be detailed later. Specifically, an 137-dimensional hand-crafted feature vector is computed for each superpixel by concatenating color and texture visual features. These features include: three mean color features with their standard deviations (6-dimensional), three mean texture features with their standard deviations (6-dimensional), and color histogram (125-dimensional). Obviously these features are not learned thus do not require additional images to compute.

Since the feature representation used for our superpixel label noise reduction model is weaker (compared to the R-CNN features), the appearance model needs to be stronger (compared to the linear SVM). To this end, an 8-layers deep CNN model is developed for modelling superpixel appear-

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TPAMI.2016.2552172, IEEE Transactions on Pattern Analysis and Machine Intelligence

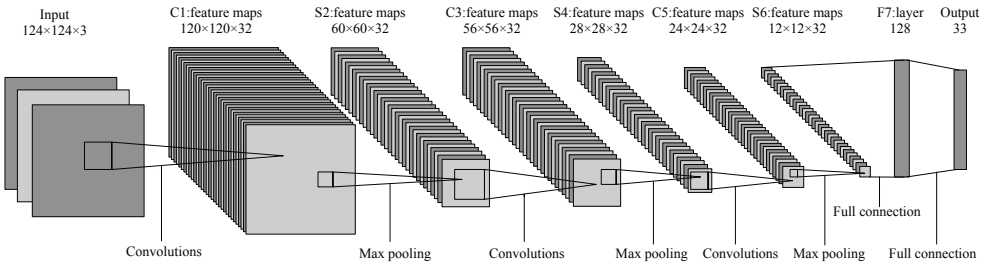


Fig. 4: The architecture of our deep CNN appearance model. Our model contains 8 layers, excluding the input layer.

ance and predicting superpixel labels[3]. The architecture of our CNN is illustrated in Figure 4. It can be seen that the architecture is similar to AlexNet [35] with the difference mainly on the image/filter sizes and the number of layers. To learn this CNN model, the bounding-box image of each superpixel is scaled to 124×124 pixels and the initial label of each superpixel is provided by our label noise reduction model (Algorithm 1) as the training labels of the CNN. Note that this model is trained from scratch using the weakly labelled training images only. With this CNN-based model, the same iterative label refinement procedure is performed, that is, we alternate between label noise reduction and CNN training/label prediction. In our experiments (Section 5.4.3), we compare the R-CNN features + Linear SVM model with the hand-crafted features + CNN model, and find that they yield similar performance.

Algorithm 2: The Complete WSSS Algorithm

Input: Training and test images
Tags of training images
Parameters k, m, λ, γ.
Output: The semantic segmentations of test images
1. Predict the tags of test images (Section 4.4) if operating under a transductive setting;
2. Over-segment each image into superpixels and then extract superpixel features (Section 4.1);
3. Assign the initial labels of superpixels from the image tags (Section 3.1);
while *a stopping criterion is not met* **do**
4. Run our superpixel label noise reduction algorithm (Algorithm 1) for semantic segmentation;
5. Train a linear SVM (Section 4.2) or a CNN model (Section 4.3) over superpixels using the outputs of Algorithm 1 as training labels;
6. Predict the labels of superpixels based on the trained superpixel appearance model.
end

4.4 Transductive vs. Inductive Learning

As in most existing weakly supervised semantic segmentation methods [14], [16], [17], our method can also operate under a transductive setting, that is, we assume that all the test images are available at once and utilise them for joint label noise deduction and refinement together with the training images. To this end, the test image labels/tags[4] need to be estimated by learning a multi-label image classification model from weakly labelled training images. Similar to [17], we first extract 4,096-dimensional deep CNN features [54] using the implementation of [55]; this is followed by training a linear SVM classifier on the training images. The classifier is then used to estimate the tags of the test images. Note that even when the powerful CNN features are used, the estimated test image tags still contain a large amount of noise as shown in our experiments, resulting in a higher level of initial superpixel label noise than that in the training images. We note from our experiments that when the test images are not used for model training (i.e. inductive learning), the performance of our method only degrades slightly compared to its transductively learned counterpart. The complete WSSS algorithm is summarised in Algorithm 2.

5 Experiments

5.1 Datasets and Settings

5.1.1 Datasets

The three most widely used benchmark datasets for WSSS are selected for performance evaluation.

Pascal VOC [56]: This dataset was originally used for the PASCAL Visual Object Category (VOC) segmentation contest 2007. Here, only the 'train-val' split is used (the 'train' set for training and the 'val' set for test), which includes 632 images downloaded from Flickr containing 20 object categories. Each image is provided with both pixel and image level labels by annotators, which makes it suitable for the evaluation of both fully and weakly supervised semantic segmentation methods. The original training set tags are clean (i.e. with annotator-provided ground-truth tags), and we call this version **VOC_T**. For evaluation under a more challenging setting, we collect the user-provided tags[5] from Flickr for the training images. Standard natural language processing (NLP) techniques are applied to remove some irrelevant tags and keep those tags that are only related to the 20 object categories[6]. This version of the Pascal VOC dataset is denoted as **VOC_N**. Note that these Flickr user-provided tags are far from perfect – as shown in Table 1, compared with the ground truth, about 10% of the present tags are wrong and 60% of the true tags are missing. This is understandable: without instruction on what should be tagged, the Flickr

3. Note that the CNN model takes raw pixel values as input, whilst the hand-crafted features are used for computing the Laplacian matrix to be used in Algorithm 1 for generating a set of denoised labels as training labels for the CNN model.

4. We use tags in the rest of the paper to refer to the image-level labels to distinguish from superpixel labels.

5. Available at: http://lear.inrialpes.fr/people/guillaumin/data.php

6. These NLP techniques include removing stop-words, finding synonyms and parasynonyms etc.

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TPAMI.2016.2552172, IEEE Transactions on Pattern Analysis and Machine Intelligence

TABLE 1: The quality of the tags of both the training and test images for the two versions of the Pascal VOC dataset.

Version	Training images			Test Images		
	Metrics	recall (%)	precision (%)	Metrics	recall (%)	precision (%)
VOC_T	True tags (noise)	100.0 (missing=0%)	100.0 (wrong=0%)	Predicted tags (noise)	62.6 (missing>30%)	66.5 (wrong>30%)
VOC_N	Flickr tags (noise)	39.6 (missing≈60%)	90.4 (wrong≈10%)	Predicted tags (noise)	49.6 (missing≈50%)	49.9 (wrong≈50%)

users would provide tags that they think are relevant to the image content, therefore missing most of the 20 pre-defined object tags (e.g. why does one want to label the potted plant in the background?). These original Flickr object tags together with our model code will be made available for download from the first author's website. In addition, the VOC 2012 dataset is selected for performance evaluation in the supplementary material due to space constraint.

MSRC [57]: This dataset contains 591 images, manually labelled with 21 object categories. Pixels on the boundaries of objects are usually labelled as background and not taken into account. To test the noise-robustness of our model, we randomly add noise (wrong/missing) to the tags of training images (see more details in Section 5.2). We use the standard training/test (276/256) split.

SIFT-Flow [58]: This dataset consists of 2,688 outdoor images, densely labelled with 33 object categories using the LabelMe online annotation tool. The standard training/test (2,488/200) split is used. Similar to MSRC, we also randomly add noise to the tags of the training images to simulate noisy image-level labels.

Note that these three datasets differ in a number of aspects. First, the definition of objects differs: MSRC and SIFT-Flow objects include sky, road and building which are normally considered as stuffs/background rather than things/foreground, whilst the VOC objects only contain things and some of them such as bottle and potted plant can be quite small in the images, thus harder to segment. Second, overall the VOC dataset is much harder than the other two for the segmentation task due to the smaller objects, co-existence of multiple objects in each image and more cluttered background. In that sense, SIFT-Flow is also harder than MSRC. Finally, as mentioned early VOC_N contains the original noisy user-provided tags from Flickr as weak labels, whilst for MSRC and SIFT-Flow, we can only simulate the noisy tags by randomly adding noise.

5.1.2 Evaluation Metrics

Most existing semantic segmentation works report results using two metrics: total per-pixel accuracy which measures the percentage of correctly labelled pixels in the test images, and average per-class accuracy which is the percentage of correctly labelled pixels for a class averaged over all object classes. Past results on the MSRC and SIFT-Flow datasets typically report results in both metrics. This is because some model parameters can normally be tuned so that one metric gets higher at the price of lowing the other metric. Therefore, for easy comparison between different methods, we also use the harmonic mean of the two metrics to measure their overall performance over the two metrics. Note that as mentioned above, compared with MSRC and SIFT-Flow, the VOC images typically contain larger portions of background (stuffs) and relatively smaller objects. The per-class accuracy is thus more appropriate and adopted by most published results. In addition, recently the intersection-over-union (IOU) score is also used for semantic segmentation performance evaluation on Pascal VOC, which is a standard measure for many other PASCAL VOC challenges. We therefore report results in both per-class accuracy and IOU for the two versions of VOC.

5.1.3 Settings

We over-segment each dataset (see Section 4.1) into a total of 15,000, 7,500 and 33,000 superpixels for VOC, MSRC, and SIFT-Flow respectively. Note that since the ground-truth pixel-level labels of all the images are unknown under the weakly supervised setting, it is not possible to tune the model parameters by cross-validation. In this paper, we thus fix the parameters of our algorithm as $k = 200$ (for the k-NN graph for computing the Laplacian matrix $\mathcal{L}$), $m = 50$ (the number of eigenvectors of $\mathcal{L}$ used in Eq. (11)), $\lambda = 0.01$ and $\gamma = 0.12$ (both are weights in Eq. (6)) for all the three datasets. We found that our model is insensitive to different values of these parameters (see Section 5.4.6 for a detailed analysis). Similar to [18], experiments are conducted under both transductive and inductive settings (see Section 4.4). Our method under the transductive setting is denoted as "Ours (trans.)", whilst under the inductive setting, it is simply "Ours". Unless otherwise stated, the appearance model described in Section 4.2 (R-CNN features+linear SVM) is adopted. The alternative appearance models used in our method are compared in Section 5.4.1.

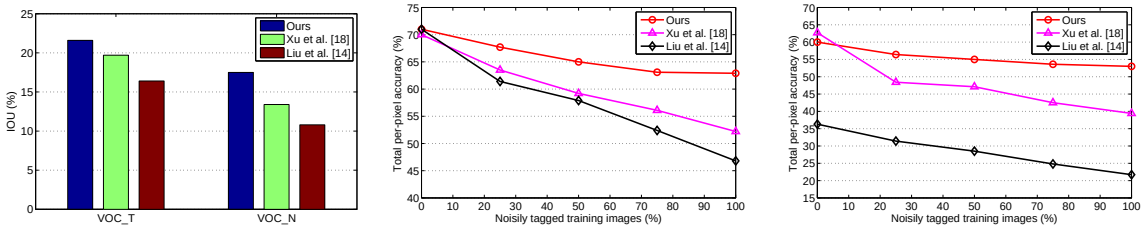
5.1.4 Compared Methods

We conduct three groups of experiments for evaluation and choose competitors to compare accordingly: (1) Our main focus is the evaluation of segmentation given weak and noisy image tags. However, most existing WSSS methods only report results obtained using clean tags. The comparison is thus mainly limited to methods [14], [18] that we have code. (2) To compare with a wider range of segmentation methods, we also carry out experiments with clean image tags and compare with the published results on the three benchmark datasets. These include the state-of-the-art weakly supervised as well as fully supervised methods. (3) Beyond WSSS, we compare our sparse learning model to alternative sparse learning based denoising models.

5.2 Segmentation with Noisy Tags

In this experiment, the training image are annotated with weakly and noisy tags, and the objective is to evaluate how different methods behave given different levels of image tag noise. In particular, for VOC, we compare the segmentation performance on both VOC_T and VOC_N in

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TPAMI.2016.2552172, IEEE Transactions on Pattern Analysis and Machine Intelligence



(a) VOC (b) MSRC (c) SIFT-Flow

Fig. 5: Comparison of different semantic segmentation methods under various noise settings. IOU is used as metrics for VOC, whilst total per-pixel accuracy is used for MSRC and SIFT-Flow.

TABLE 2: Comparison of our method to [19] under the same noise model on the SIFT-Flow dataset.

Noise (%)	0	10	25	50	75
Ours (16K R-CNN features)	**40.8**	**37.2**	**35.6**	**33.9**	**31.0**
Ours (4K features of [19])	37.9	36.4	34.3	32.8	30.6
Zhang et al. [19] (4K features)	32.3	32.8	32.4	29.8	22.3

order to examine how robust different methods are against the naturally present tag noise. As for MSRC and SIFT-Flow, we simulate the label noise by randomly selecting various percentages of training images to be nosily tagged. A "noisily tagged" training image means that a wrong tag is randomly added to the list of ground-truth tags and in the meantime, a random true tag is removed from the list. That is, each "noisily tagged" training image contains two tag errors: one missing tag and one wrong tag. This represents a significant amount of label noise for that image as on average there are only 2.5 ground-truth tags per image in MSRC and 4.4 per image in SIFT-Flow.

The results of our method, [14] and [18] on the three datasets are shown in Figure 5. It is clear that when increasing levels of image-level label noise is present in the training data, as expected, all compared methods perform worse. However, the performance of our method degrades much more gracefully; as a result, the advantage of our method over those of [14] and [18] gets bigger. In particular, for VOC, given significant amount of label noise in VOC_N (see Table 1), Figure 5(a) shows that the performance gain of our method over [18] increases from 1.9% on VOC_T to 4.1% on VOC_N. On the less challenging MSRC dataset (Figure 5(b)), all three compared methods are at par given clean tags. However, when 100% of the training images are corrupted with noisy tags, a large gap of 10.7% appears between ours and [18], and an even bigger gap exists between ours and [14]. As for SIFT-Flow (Figure 5(c)), the result of [18] is actually better than ours given clean tags. However, when more and more training images are corrupted with noisy tags, that advantage soon disappeared and when all training images contain noisy tags, our method has already built up a 13.6% advantage over that of [18]. Note that all three methods have exactly the same model input (superpixel segmentation and representation are identical); their appearance models are also similar (linear models). These results thus provide clear evidence that our method is more robust against label noise thanks to its explicit and direct noise reduction model.

TABLE 3: Results on the VOC_T dataset. Whether the results are obtained under a transductive setting is also indicated. The reported results for [18] and [53] were obtained by rerunning the released codes.

Method	Supervision	Trans.?	per-class (%)	IOU (%)
Upper bound (ours)	full	N	49.2	23.6
Ours (trans.)	weak	Y	**48.9**	**21.6**
Ours	weak	N	48.1	20.8
Xu et al. [18] (trans.)	weak	Y	48.5	19.7
Xu et al. [18]	weak	N	47.8	18.3
Liu et al. [14]	weak	Y	29.8	16.4
Zhang et al. [19]	weak	Y	44.6	N/A
Xie et al. [59]	weak	Y	42.0	N/A
Zhang et al. [15]	weak	N	24.0	N/A
Liu et al. [13]	weak	Y	38.0	N/A
Liu et al. [60]	weak	Y	32.0	N/A
Ladicky et al. [4]	full	N	30.0	N/A
Larlus et al. [61]	full	N	37.2	N/A
Shotton et al. [2]	full	N	42.0	N/A
Girshick et al. [53]	full	N	**43.0**	**26.7**

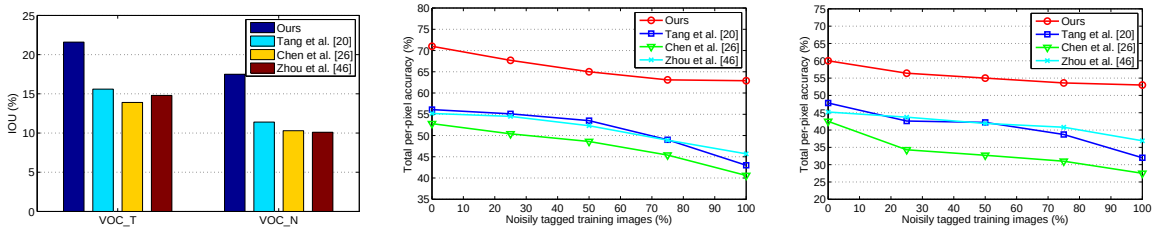
TABLE 4: Results on MSRC with clean tags

Method	Supervision	Trans.?	per-pixel	per-class	harmonic
Upper bound (ours)	full	N	80.0	75.7	77.8
Ours (trans.)	weak	Y	**71.0**	**74.7**	**72.8**
Ours	weak	N	69.8	71.5	70.6
Xu et al. [18] (trans.)	weak	Y	70.0	73.0	71.5
Xu et al. [18]	weak	N	68.3	70.6	69.4
Liu et al. [14]	weak	Y	**71.0**	70.0	70.5
Zhang et al. [15]	weak	N	N/A	69.0	N/A
Vezhnevets et al. [11]	weak	Y	67.0	67.0	67.0
Shotton et al. [2]	full	N	72.0	67.0	69.4
Ladicky et al. [4]	full	N	86.0	75.0	80.1
Lucchi et al. [28]	full	N	79.0	78.0	78.5
Boix et al. [62]	full	N	83.0	**80.0**	81.5
Lucchi et al. [63]	full	N	82.0	76.0	78.9
Yao et al. [64]	full	N	**86.2**	79.3	**82.6**

We further compare our method to the only existing method [19] for WSSS with noisy tags. Since we have no access to the code of this method, we make comparison against their reported results on SIFT-Flow under their noise model[7], and use the same evaluation metric – average per-class accuracy. For fair comparison, the features of [19] are also used for our model. The results are shown in Table 2. It can be seen that no matter what features are used, our method is clearly superior to their method. This suggests that the improvement comes mainly from the model itself rather than features used. Note that the gap is particularly big given extremely noisy labels (e.g., 75%).

7. Note that missing and wrong labels are introduced to each existing label randomly without selecting which image to add noise first.

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TPAMI.2016.2552172, IEEE Transactions on Pattern Analysis and Machine Intelligence



(a) VOC (b) MSRC (c) SIFT-Flow

Fig. 6: Comparison of different sparse learning models under various noisy settings.

TABLE 5: Results on SIFT-Flow with clean tags

Method	Supervision	Trans.?	per-pixel	per-class	harmonic
Upper bound (ours)	full	N	77.9	42.8	55.2
Ours (trans.)	weak	Y	60.0	40.8	48.6
Ours	weak	N	58.8	37.7	45.9
Xu et al. [18] (trans.)	weak	Y	62.7	41.4	**49.9**
Xu et al. [18]	weak	N	**65.0**	35.0	45.5
Caesar et al. [65]	weak	N	N/A	**44.8**	N/A
Zhang et al. [19]	weak	Y	N/A	32.3	N/A
Xu et al. [17]	weak	Y	N/A	27.9	N/A
Liu et al. [14]	weak	Y	36.3	26.3	30.5
Vezhnevets et al. [12]	weak	Y	51.0	21.0	29.8
Vezhnevets et al. [11]	weak	Y	N/A	14.0	N/A
Rubinstein et al. [66]	weak+full	Y	63.3	29.5	40.2
Tighe et al. [67]	full	N	76.9	29.4	42.5
Liu et al. [58]	full	N	76.7	24.0	36.6
Tighe et al. [29]	full	N	77.0	30.1	43.3
Tighe et al. [68]	full	N	78.6	39.2	52.3
Yang et al. [6]	full	N	79.8	48.7	60.5
Long et al. [31]	full	N	**85.2**	51.7	**64.4**
Caesar et al. [65]	full	N	N/A	**59.2**	N/A

5.3 Segmentation with Noise-Free Tags

In this experiment, we compare our method with the state-of-the-art fully and weakly supervised methods given clean ground-truth tags provided for the training data. The results on the three benchmark datasets are shown in Tables 3–5, respectively. Here, we also show *the upper bounds of our method* (non-transductive) obtained by directly initialising the superpixel labels Y with the ground-truth segmentations of training images. We can make the following observations: (1) Even though our method is designed for the more challenging case where image tags are both weak and noisy, given clean tags, our method remains competitive – it achieves the state-of-the-art results on VOC and MSRC among the compared WSSS methods, and is only slightly worse than [18] on SIFT-Flow under the transductive setting (harmonic mean of 48.6% vs. 49.9%), and marginally better under the inductive setting (45.9% vs. 45.5%). Admittedly given clean tags, the gaps between the results of the best WSSS methods are small. (2) The difference between transductive and inductive learning seems to be small for our method although it does benefit from the access to the whole test set at once. It is also noted that for [18], jointly learning the model with all the test data together even hampers the performance slightly on the SIFT-Flow dataset (see Table 5). This is because that under the transductive setting, the predicted tags on the test image will be used for model learning, which themselves contain a significant amount of noise as shown in Table 1. The benefit of using more data for training is thus out-weighted by the presence of more label noise when noise deduction is not solved explicitly. (3) Comparing with the fully supervised methods, there are still clear gaps between theirs and the results of the best WSSS methods when measured using the total per-pixel accuracy. This is particularly true for the more recent fully supervised methods, such as the fully convolutional network [31] (see Table 5). However, when the harmonic mean of the per-pixel and per-class accuracies is used, the performance is much closer and sometimes comparable. This shows that with the added benefit of scalability, a WSSS approach is indeed a promising solution to large-scale semantic segmentation. (4) Our method consistently yields strong results on the three datasets, whilst [14] only performs well on MSRC. That is, the performance of [14] is more likely to be affected by the dataset itself, due to the special definition of the loss function in [14].

5.4 Further Evaluations

5.4.1 Alternative Sparse Learning Models

We also compare our sparse learning model for superpixel label noise reduction (Algorithm 1) to two relevant alternatives [20], [26] which also aim to correct noisy tags following a sparse learning formulation. A key difference between these two works with ours is that they do not exploit the graph Laplacian constraint to enforce the smoothness of label assignment following the intrinsic data structure in the low-level feature space. In this aspect, the work of [46] is related to ours which also uses the Laplacian regularisation. Although it is designed for semi-supervised learning rather than explicit label noise reduction due to not having the L_1-optimisation terms in their cost function, it has the potential to be applied to the superpixel label assignment problem. These three models using exactly the same superpixels and features as ours are compared in Figure 6. It can be seen that our sparse learning model is clearly superior to these related weakly-supervised/noise deduction models, due to (a) the combination of the noise sparsity constraint and the visual similarity constraint, and (b) the novel optimisation algorithm developed to solve the L_1-optimisation problem. In addition, we compare our sparse learning model with its three variants whereby one or both constraints use Frobenious norm instead of L_1-norm to show the importance of using L_1-norm in both constraints. The results can be found in the supplementary material.

5.4.2 Effectiveness of Intermediate Labelling Variable

One of the key ideas in the proposed L_1-optimisation problem solver in Section 3.3 is the introduction of an intermediate labelling variable F to turn a cost function

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TPAMI.2016.2552172, IEEE Transactions on Pattern Analysis and Machine Intelligence

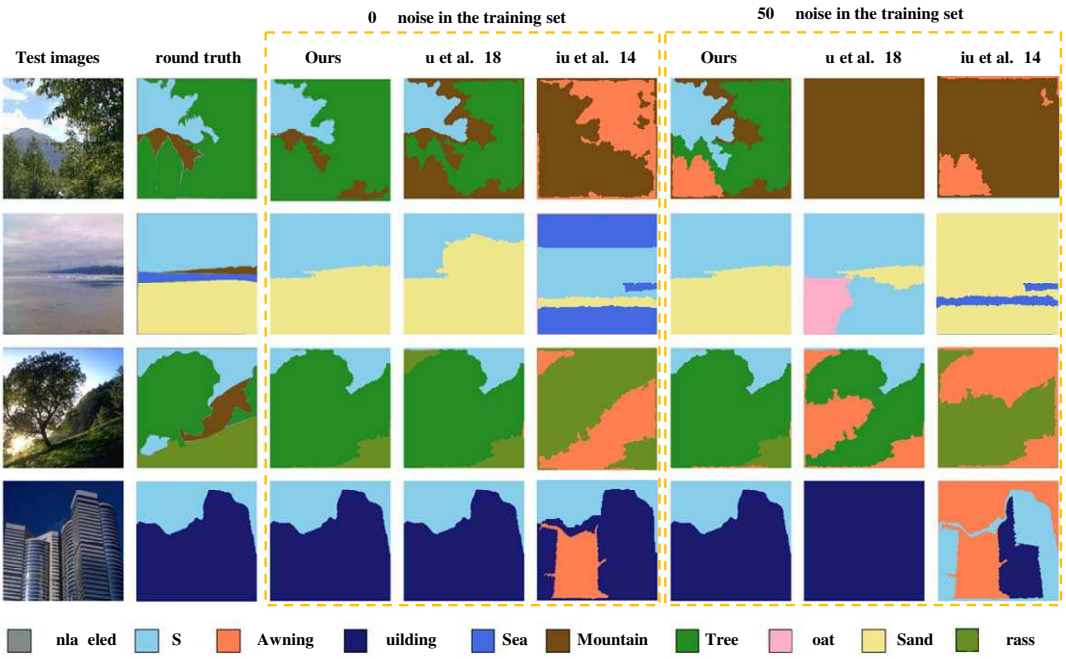


Fig. 7: Qualitative results of semantic segmentation with clean and noisy tags on the SIFT-Flow dataset

TABLE 6: Comparison of our algorithm to alternative noise reduction models on the MSRC dataset. The runtime was obtained on a PC platform with a 3.9GHz CPU and 32GB RAM with Matlab implementations.

Method	per-pixel (%)	per-class (%)	harmonic	time (sec.)
Ours	71.0	74.7	72.8	175
Ours (orig.)	70.2	71.5	70.8	383

TABLE 7: Comparison on different superpixel representations and appearance models used in our method on the SIFT-Flow dataset.

Representation and Model	per-pixel (%)	per-class (%)
R-CNN superpixel features + Algorithm 1 + linear SVM	58.8	37.7
handcrafted superpixel features + Algorithm 1 + linear SVM	51.9	30.1
handcrafted superpixel features + Algorithm 1 + deep CNN	60.2	38.6

consisting of only L_1-norm terms (Eq. (4)) to one with a hybrid of L_1-norm and Frobenious norm terms (Eq. (5)). To validate its effectiveness, we compare our algorithm to a variant that directly solves the original problem in Eq. (4) using the YALL1 toolbox [48] with the same dimensionality reduction $\hat{Y} = V_m A$ for fair comparison (denoted as Ours (orig.)). The two algorithms are compared both in terms of segmentation performance and runtime in Table 6. The results indicate that by introducing an intermediate labelling variable, our algorithm achieves better segmentation accuracy and is twice as faster. This is because directly solving the original L_1-optimisation problem in Eq. (4) relies on making approximations to be tractable which leads to performance degradation.

5.4.3 Alternative Superpixel Appearance Models

So far, the results present are based on the superpixel appearance model described in Section 4.2, that is R-CNN features + label noise reduction (Algorithm 1) + linear SVM. Here, we also evaluate two alternatives: (1) the hand-crafted superpixel features + label noise reduction + deep CNN appearance model described in Section 4.3. (2) the same hand-crafted superpixel features + label noise reduction + linear SVM. Both models do not require any additional data to pre-train. The results on the SIFT-Flow dataset are shown in Table 7. We can see that the deep CNN based appearance model, although slower to run than the linear SVM, performs slightly better[8]. But when both the feature representation and the appearance model are weak (hand-crafted features + label noise reduction + linear SVM), the result is clearly inferior. This suggests that our model can benefit from the stronger deep learning based appearance modelling without requiring any additional data for pre-training or feature extraction.

5.4.4 Qualitative Results

In Figure 7, we compare the methods of [14], [18] with ours given noisy and clean labels on the SIFT-Flow dataset. It is observed that when the training image labels are clean, both our and [18]'s methods produce good segmentation, whilst the method of [14] failed to learn the appearance model for certain categories correctly (e.g. awning) resulting in large regions being mis-labelled. When 50% of the training images are corrupted with label noise, all three models are affected adversely. However, our model clearly is more robust against the noisy labels than the other two. More qualitative results on the VOC and MSRC datasets can be found in the supplementary material.

5.4.5 Runtime Comparison

We compare our method with [14], [18] in terms of runtime for both training and testing in Table 8. We run all three

8. The performance of the CNN model on VOC and MSRC is slightly worse because these two datasets have significantly less superpixels for training leading to model overfitting.

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TPAMI.2016.2552172, IEEE Transactions on Pattern Analysis and Machine Intelligence

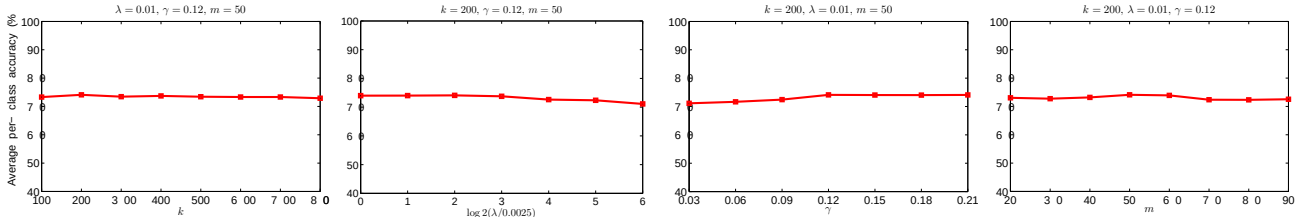


Fig. 8: Illustration of the effect of different parameters on our semantic segmentation algorithm for the MSRC dataset.

TABLE 8: Comparison of different semantic segmentation methods in terms of runtime on the MSRC dataset. Training involves processing 276 images and testing 256 images.

Method	training time (sec.)	test time (sec.)
Ours	173	2
Xu et al. [18]	159	2
Liu et al. [14]	245	2

methods (implemented in Matlab) on a PC platform with a 3.9GHz CPU and 32GB RAM. The results in Table 8 show that: (1) The three methods have the same test efficiency (around 2 seconds to process 256 images), because they all adopt a linear superpixel appearance model (in fact, most time is spent on feature extraction). (2) During training, our method and [18]'s are comparable and both are much more efficient than [14]'s.

5.4.6 Sensitivity to Parameter Settings

Note that since the ground-truth pixel-level labels of all the images are unknown under the weakly supervised setting, it is not possible to tune the model parameters by cross-validation. In this paper, we thus fix the four parameters of our algorithm as $k = 200$, $m = 50$, $\lambda = 0.01$, and $\gamma = 0.12$ for all the three datasets. In Figure 8, we investigate on how sensitive the method is to different values of these parameters. The results show that the influence of different parameter settings is very small.

6 Conclusions

In this paper, we have proposed a novel approach to learning a semantic segmentation model from both weak and noisy labels. The weakly supervised semantic segmentation problem is cast into a noise reduction problem and a superpixel label noise reduction model is developed based on a novel sparse learning model with an efficient optimisation algorithm. Extensive experiments are carried out to demonstrate that the proposed method is superior to the state-of-the-art methods and alternative sparse learning based label denoising models, particularly when the weak labels are also noisy. A number of directions are worth further investigation. First, in the current framework, our method alternates between the learning of the superpixel appearance model and the label noise reduction model. It is possible to integrate the two into a single model [14], [18], even though solving it often involves an alternating processing similar to ours. More efforts are needed to investigate the connection between the two approaches (one vs. two models) and their pros and cons. Second, in a practical large-scale learning scenario, new classes often need to be added dynamically to an existing model so that one does not have to retrain the model from scratch. An incremental learning variant of the current method is thus part of the ongoing work. Finally, it is worth pointing out that the label noise reduction model introduced in Section 3 is by no means restricted to the WSSS problem – many other vision problems need to deal with label noise when image labels are increasingly being harvested from social media sites. Current efforts thus also include the generalisation of the proposed model to solve a wider range of computer vision problems.

Appendix 1: Proof on Solving Eq. (11) Providing an Exact Solution to Eq. (7)

In Section 3.3, instead of solving the intractable L_1-optimisation subproblem in Eq. (7), we proposed to solve a much smaller-scale problem in Eq. (11). Here we provide a proof that these two problems are equivalent under an easy-to-satisfy condition.

Since V is an orthonormal matrix, any $F \in \mathcal{R}^{N\times C}$ can be denoted as $F = VA$, where $A = \{a_{ij}\}_{N\times C}$. Considering the orthonormality of V, we reformulate Eq. (7) as:

$$\begin{aligned}
&\arg\min_A \frac{1}{2}||VA - \hat{Y}^*||_F^2 + \lambda||BVA||_1 \\
&= \arg\min_A \sum_{j=1}^{C} \frac{1}{2}||VA_{.j} - \hat{Y}^*_{.j}||_2^2 + \lambda||\Sigma^{\frac{1}{2}}V^TVA_{.j}||_1 \\
&= \arg\min_A \sum_{j=1}^{C} \frac{1}{2}||VA_{.j} - \hat{Y}^*_{.j}||_2^2 + \lambda\sum_{i=1}^{N}\Sigma_{ii}^{\frac{1}{2}}|a_{ij}| \\
&= \arg\min_A \sum_{j=1}^{C}\sum_{i=1}^{N} \frac{1}{2}a_{ij}^2 - (V_{.i}^T\hat{Y}^*_{.j})a_{ij} + \lambda\Sigma_{ii}^{\frac{1}{2}}|a_{ij}|. \quad (13)
\end{aligned}$$

This means that we can solve Eq. (7) by solving the following $N \times C$ subproblems independently:

$$\arg\min_{a_{ij}} \frac{1}{2}a_{ij}^2 - (V_{.i}^T\hat{Y}^*_{.j})a_{ij} + \lambda\Sigma_{ii}^{\frac{1}{2}}|a_{ij}|. \quad (14)$$

Based on the above problem decomposition, we come to the following proposition about the dimension reduction of F used in Eq. (11).

Proposition 1: If $\lambda > \Sigma_{mm}^{-\frac{1}{2}} \cdot \max_{1\le j\le C, i>m}|V_{.i}^T\hat{Y}^*_{.j}| = \lambda^*(m)$, the solution $A^* = \{a^*_{ij}\}_{N\times C}$ of Eq. (13) satisfies that: $a^*_{ij} = 0$ $(1 \le j \le C, i > m)$.

Proof: Since $\lambda > \lambda^*(m)$ and $\Sigma_{ii}^{\frac{1}{2}} \ge \Sigma_{mm}^{\frac{1}{2}}$ $(i > m)$, we have: $\lambda > \Sigma_{mm}^{-\frac{1}{2}} \cdot \max_{1\le j\le C, i>m}|V_{.i}^T\hat{Y}^*_{.j}| \ge \Sigma_{ii}^{-\frac{1}{2}} \cdot \max_{1\le j\le C, i>m}|V_{.i}^T\hat{Y}^*_{.j}| \ge \Sigma_{ii}^{-\frac{1}{2}} \cdot |V_{.i}^T\hat{Y}^*_{.j}|$, i.e.,

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TPAMI.2016.2552172, IEEE Transactions on Pattern Analysis and Machine Intelligence

$\lambda\Sigma_{ii}^{\frac{1}{2}} > |V_{.i}^T \hat{Y}_{.j}^*|$ $(i > m)$. We thus have: $\lambda\Sigma_{ii}^{\frac{1}{2}}|a_{ij}| \geq |V_{.i}^T \hat{Y}_{.j}^*||a_{ij}| \geq (V_{.i}^T \hat{Y}_{.j}^*)a_{ij}$ $(i > m)$. Hence, $\frac{1}{2}a_{ij}^2 - (V_{.i}^T \hat{Y}_{.j}^*)a_{ij} + \lambda\Sigma_{ii}^{\frac{1}{2}}|a_{ij}| \geq 0$. We can readily obtain the solution of Eq. (14) exactly as: $a_{ij}^* = 0$ $(i > m)$. This means that the solution $A^* = \{a_{ij}^*\}_{N\times C}$ of Eq. (13) satisfies: $a_{ij}^* = 0$ $(1 \leq j \leq C, i > m)$. □

According to Proposition 1, if we set $\lambda > \lambda^*(m)$ initially, the solution of Eq. (7) is equal to that of Eq. (11). That is, Proposition 1 provides a theoretical guarantee for the dimension reduction of F used in Eq. (11).

ACKNOWLEDGEMENTS

This work was partially supported by National Natural Science Foundation of China (61573363 and 61573026), 973 Program of China (2014CB340403 and 2015CB352502), the Fundamental Research Funds for the Central Universities and the Research Funds of Renmin University of China (15XNLQ01), IBM Global SUR Award Program, European Research Council FP7 Project SUNNY (313243), and the funding from KAUST.

REFERENCES

[1] H. Zhu, F. Meng, J. Cai, and S. Lu, "Beyond pixels: A comprehensive survey from bottom-up to semantic image segmentation and cosegmentation," *CoRR*, vol. abs/1502.00717, 2015.

[2] J. Shotton, M. Johnson, and R. Cipolla, "Semantic texton forests for image categorization and segmentation," in *CVPR*, 2008, pp. 1–8.

[3] P. Kohli, L. Ladicky, and P. Torr, "Robust higher order potentials for enforcing label consistency," *International Journal of Computer Vision*, vol. 82, no. 3, pp. 302–324, 2009.

[4] L. Ladicky, C. Russell, P. Kohli, and P. Torr, "Associative hierarchical CRFs for object class image segmentation," in *ICCV*, 2009, pp. 739–746.

[5] ——, "Graph cut based inference with co-occurrence statistics," in *ECCV*, 2010, pp. 239–253.

[6] J. Yang, B. Price, S. Cohen, and M.-H. Yang, "Context driven scene parsing with attention to rare classes," in *CVPR*, 2014, pp. 3294–3301.

[7] F. Tung and J. J. Little, "Collageparsing: Nonparametric scene parsing by adaptive overlapping windows," in *ECCV*, 2014, pp. 511–525.

[8] S. Gould, J. Zhao, X. He, and Y. Zhang, "Superpixel graph label transfer with learned distance metric," in *ECCV*, 2014, pp. 632–647.

[9] J. Verbeek and B. Triggs, "Region classification with Markov field aspect models," in *CVPR*, 2007, pp. 1–8.

[10] A. Vezhnevets and J. Buhmann, "Towards weakly supervised semantic segmentation by means of multiple instance and multitask learning," in *CVPR*, 2010, pp. 3249–3256.

[11] A. Vezhnevets, V. Ferrari, and J. Buhmann, "Weakly supervised semantic segmentation with a multi-image model," in *ICCV*, 2011, pp. 643–650.

[12] ——, "Weakly supervised structured output learning for semantic segmentation," in *CVPR*, 2012, pp. 845–852.

[13] S. Liu, S. Yan, T. Zhang, C. Xu, J. Liu, and H. Lu, "Weakly supervised graph propagation towards collective image parsing," *IEEE Trans. Multimedia*, vol. 14, no. 2, pp. 361–373, 2012.

[14] Y. Liu, J. Liu, Z. Li, J. Tang, and H. Lu, "Weakly-supervised dual clustering for image semantic segmentation," in *CVPR*, 2013, pp. 2075–2082.

[15] K. Zhang, W. Zhang, Y. Zheng, and X. Xue, "Sparse reconstruction for weakly supervised semantic segmentation," in *IJCAI*, 2013, pp. 1889–1895.

[16] S. Liu, J. Feng, C. Domokos, H. Xu, J. Huang, Z. Hu, and S. Yan, "Fashion parsing with weak color-category labels," *IEEE Trans. Multimedia*, vol. 16, no. 1, pp. 253–265, 2014.

[17] J. Xu, A. Schwing, and R. Urtasun, "Tell me what you see and I will show you where it is," in *CVPR*, 2014, pp. 3190–3197.

[18] J. Xu, A. G. Schwing, and R. Urtasun, "Learning to segment under various forms of weak supervision," in *CVPR*, 2015, pp. 3781–3790.

[19] W. Zhang, S. Zeng, D. Wang, and X. Xue, "Weakly supervised semantic segmentation for social images," in *CVPR*, 2015, pp. 2718–2726.

[20] J. Tang, S. Yan, R. Hong, G.-J. Qi, and T.-S. Chua, "Inferring semantic concepts from community-contributed images and noisy tags," in *ACM Multimedia*, 2009, pp. 223–232.

[21] Z. Feng, S. Feng, R. Jin, and A. K. Jain, "Image tag completion by noisy matrix recovery," in *ECCV*, 2014, pp. 424–438.

[22] M. Elad and M. Aharon, "Image denoising via sparse and redundant representations over learned dictionaries," *IEEE Trans. Image Processing*, vol. 15, no. 12, pp. 3736–3745, 2006.

[23] J. Mairal, M. Elad, and G. Sapiro, "Sparse representation for color image restoration," *IEEE Trans. Image Processing*, vol. 17, no. 1, pp. 53–69, 2008.

[24] J. Wright, A. Yang, A. Ganesh, S. Sastry, and Y. Ma, "Robust face recognition via sparse representation," *IEEE Trans. Pattern Analysis and Machine Intelligence*, vol. 31, no. 2, pp. 210–227, 2009.

[25] H. Lee, A. Battle, R. Raina, and A. Ng, "Efficient sparse coding algorithms," in *Advances in Neural Information Processing Systems 19*, 2007, pp. 801–808.

[26] X. Chen, Q. Lin, S. Kim, J. Carbonell, and E. Xing, "Smoothing proximal gradient method for general structured sparse learning," in *UAI*, 2011, pp. 105–114.

[27] G. Csurka and F. Perronnin, "An efficient approach to semantic segmentation," *International Journal of Computer Vision*, vol. 95, no. 2, pp. 198–212, 2011.

[28] A. Lucchi, Y. Li, X. Boix, K. Smith, and P. Fua, "Are spatial and global constraints really necessary for segmentation?" in *ICCV*, 2011, pp. 9–16.

[29] J. Tighe and S. Lazebnik, "Superparsing - scalable nonparametric image parsing with superpixels," *International Journal of Computer Vision*, vol. 101, no. 2, pp. 329–349, 2013.

[30] F.-J. Chang, Y.-Y. Lin, and K.-J. Hsu, "Multiple structured-instance learning for semantic segmentation with uncertain training data," in *CVPR*, 2014, pp. 360–367.

[31] J. Long, E. Shelhamer, and T. Darrell, "Fully convolutional networks for semantic segmentation," in *CVPR*, 2015, pp. 3431–3440.

[32] P. H. O. Pinheiro and R. Collobert, "From image-level to pixel-level labeling with convolutional networks," in *CVPR*, 2015, pp. 1713–1721.

[33] L.-J. Li, R. Socher, and F.-F. Li, "Towards total scene understanding: Classification, annotation and segmentation in an automatic framework." in *CVPR*, 2009, pp. 2036–2043.

[34] R. Mottaghi, X. Chen, X. Liu, N.-G. Cho, S.-W. Lee, S. Fidler, R. Urtasun, and A. Yuille, "The role of context for object detection and semantic segmentation in the wild," in *CVPR*, 2014, pp. 891–898.

[35] A. Krizhevsky, I. Sutskever, and G. E. Hinton, "ImageNet classification with deep convolutional neural networks," in *Advances in Neural Information Processing Systems 25*, 2012, pp. 1097–1105.

[36] C. Farabet, C. Couprie, L. Najman, and Y. LeCun, "Learning hierarchical features for scene labeling," *IEEE Trans. Pattern Analysis and Machine Intelligence*, vol. 35, no. 8, pp. 1915–1929, 2013.

[37] B. Hariharan, P. Arbelaez, R.Girshick, and J. Malik, "Simultaneous detection and segmentation," in *ECCV*, 2014, pp. 297–312.

[38] L.-C. Chen, G. Papandreou, I. Kokkinos, K. Murphy, and A. Yuille, "Semantic image segmentation with deep convolutional nets and fully connected CRFs," in *ICLR*, 2015.

[39] D. Pathak, E. Shelhamer, J. Long, and T. Darrell, "Fully convolutional multi-class multiple instance learning," in *ICLR*, 2015.

[40] A. Joulin, F. Bach, and J. Ponce, "Multi-class cosegmentation," in *CVPR*, 2012, pp. 542–549.

[41] Z. Yuan, T. Lu, and P. Shivakumara, "A novel topic-level random walk framework for scene image co-segmentation," in *ECCV*, 2014, pp. 695–709.

[42] K. Tang, A. Joulin, L.-J. Li, and L. Fei-Fei, "Co-localization in real-world images," in *CVPR*, 2014, pp. 1464–1471.

[43] J. Lin, L.-Y. Duan, J. Yuan, Q. Li, and S. Luo, "Learning sparse tag patterns for social image classification," in *ICIP*, 2012, pp. 2881–2884.

[44] B. Frénay and M. Verleysen, "Classification in the presence of label noise: A survey," *IEEE Trans. Neural Networks and Learning Systems*, vol. 25, no. 5, pp. 845–869, 2014.

[45] X. Zhu, Z. Ghahramani, and J. Lafferty, "Semi-supervised learning using Gaussian fields and harmonic functions," in *ICML*, 2003, pp. 912–919.

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TPAMI.2016.2552172, IEEE Transactions on Pattern Analysis and Machine Intelligence

[46] D. Zhou, O. Bousquet, T. Lal, J. Weston, and B. Schölkopf, "Learning with local and global consistency," in *Advances in Neural Information Processing Systems 16*, 2004, pp. 321–328.

[47] Y. Fu, T. Hospedales, T. Xiang, J. Xiong, S. Gong, Y. Wang, and Y. Yao, "Robust subjective visual property prediction from crowdsourced pairwise labels," *IEEE Trans. Pattern Analysis and Machine Intelligence*, 2015.

[48] J. Yang and Y. Zhang, "Alternating direction algorithms for ℓ_1-problems in compressive sensing," *SIAM journal on scientific computing*, vol. 33, no. 1, pp. 250–278, 2011.

[49] G. Liu, Z. Lin, and Y. Yu, "Robust subspace segmentation by low-rank representation," in *ICML*, 2010, pp. 663–670.

[50] R. Fergus, Y. Weiss, and A. Torralba, "Semi-supervised learning in gigantic image collections," in *Advances in Neural Information Processing Systems 22*, 2010, pp. 522–530.

[51] P. Arbeláez, J. Pont-Tuset, J. Barron, F. Marques, and J. Malik, "Multiscale combinatorial grouping," in *CVPR*, 2014, pp. 328–335.

[52] P. Rantalankila, J. Kannala, and E. Rahtu, "Generating object segmentation proposals using global and local search," in *CVPR*, 2014, pp. 2417–2424.

[53] R. Girshick, J. Donahue, T. Darrell, and J. Malik, "Rich feature hierarchies for accurate object detection and semantic segmentation," in *CVPR*, 2014, pp. 580–587.

[54] J. Deng, W. Dong, R. Socher, L.-J. Li, K. Li, and L. Fei-Fei, "ImageNet: A large-scale hierarchical image database," in *CVPR*, 2009, pp. 248–255.

[55] P. Sermanet, D. Eigen, X. Zhang, M. Mathieu, R. Fergus, and Y. LeCun, "OverFeat: Integrated recognition, localization and detection using convolutional networks," in *ICLR*, 2014.

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

[57] J. Shotton, J. Winn, C. Rother, and A. Criminisi, "Textonboost: Joint appearance, shape and context modeling for multi-class object recognition and segmentation," in *ECCV*, 2006, pp. 1–15.

[58] C. Liu, J. Yuen, and A. Torralba, "Nonparametric scene parsing via label transfer," *IEEE Trans. Pattern Analysis and Machine Intelligence*, vol. 33, no. 12, pp. 2368–2382, 2011.

[59] W. Xie, Y. Peng, and J. Xiao, "Semantic graph construction for weakly-supervised image parsing," in *AAAI*, 2014, pp. 2853–2859.

[60] X. Liu, B. Cheng, S. Yan, J. Tang, T. S. Chua, and H. Jin, "Label to region by bi-layer sparsity priors," in *ACM Multimedia*, 2009, pp. 115–124.

[61] D. Larlus, J. Verbeek, and F. Jurie, "Category level object segmentation by combining bag-of-words models with Dirichlet processes and random fields," *International Journal of Computer Vision*, vol. 88, no. 2, pp. 238–253, 2010.

[62] X. Boix, J. M. Gonfaus, J. V. de Weijer, A. D. Bagdanov, J. Serrat, and J. Gonzalez, "Harmony potentials: Fusing global and local scale for semantic image segmentation," *International Journal of Computer Vision*, vol. 96, pp. 83–102, 2012.

[63] A. Lucchi, Y. Li, K. Smith, and P. Fua, "Structured image segmentation using kernelized features," in *ECCV*, 2012, pp. 400–413.

[64] J. Yao, S. Fidler, and R. Urtasun, "Describing the scene as a whole: Joint object detection, scene classification and semantic segmentation," in *CVPR*, 2012, pp. 702–709.

[65] H. Caesar, J. Uijlings, and V. Ferrari, "Joint calibration for semantic segmentation," in *BMVC*, 2015.

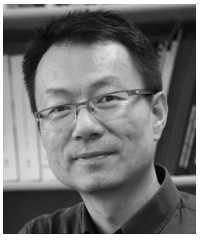
[66] M. Rubinstein, C. Liu, and W. T. Freeman, "Annotation propagation in large image databases via dense image correspondence," in *ECCV*, 2012, pp. 85–99.

[67] J. Tighe and S. Lazebnik, "Superparsing: Scalable nonparametric image parsing with superpixels," in *ECCV*, 2010, pp. 352–365.

[68] ——, "Finding things: Image parsing with regions and per-exemplar detectors," in *CVPR*, 2013, pp. 3001–3008.

Zhiwu Lu received the M.S. degree in applied mathematics from Peking University in 2005, and the Ph.D. degree in computer science from City University of Hong Kong in 2011. He is currently an associate professor of School of Information, Renmin University of China. He won the Best Paper Award at CGI 2014 and IBM SUR Award 2015. His research interests lie in machine learning, pattern recognition, and computer vision.

Zhenyong Fu received the Ph.D. degree in Computer Science from Shanghai Jiao Tong University in 2012. He is currently a Postdoctoral Researcher in Computer Vision Group in the School of Electronic Engineering and Computer Science, Queen Mary University of London. His research interests includes computer vision and machine learning.



Tao Xiang received the Ph.D. degree in electrical and computer engineering from the National University of Singapore in 2002. He is currently a reader (associate professor) in the School of Electronic Engineering and Computer Science, Queen Mary University of London. His research interests include computer vision, machine learning, and data mining. He has published over 120 papers in international journals and conferences.

Peng Han received the B.S. degree in information and computing science from University of Science and Technology Beijing in 2014. He is currently working toward the M.Eng degree in machine learning and pattern recognition at Renmin University of China. His main research interests include machine learning and parallel computing.

Liwei Wang received the Ph.D. degree from School of Mathematical Sciences, Peking University in 2005; the B.S. and M.S. degrees from Department of Electronic Engineering, Tsinghua University in 1999 and 2002, respectively. He is currently a full professor of School of Electronics Engineering and Computer Sciences, Peking University. He was named among "AI's 10 to Watch" in 2010. His research interest is machine learning, with application to computer vision.

Xin Gao received the bachelor degree in 2004 from Department of Computer Science and Technology at Tsinghua University, China, and the Ph.D. degree in 2009 from School of Computer Science at University of Waterloo, Canada. He is currently an assistant professor of computer science at King Abdullah University of Science and Technology (KAUST), Saudi Arabia. His research interests lie in bioinformatics, machine learning, and optimization.