Weakly Supervised Semantic Segmentation Using Constrained Dominant Sets

Sinem Aslan[1,2,3](✉) and Marcello Pelillo[1,2]

[1] DAIS, Ca' Foscari University of Venice, Venice, Italy
{sinem.aslan,pelillo}@unive.it
[2] ECLT, Ca' Foscari University of Venice, Venice, Italy
[3] International Computer Institute, Ege University, Izmir, Turkey

Abstract. The availability of large-scale data sets is an essential prerequisite for deep learning based semantic segmentation schemes. Since obtaining pixel-level labels is extremely expensive, supervising deep semantic segmentation networks using low-cost weak annotations has been an attractive research problem in recent years. In this work, we explore the potential of *Constrained Dominant Sets* (*CDS*) for generating multi-labeled full mask predictions to train a fully convolutional network (FCN) for semantic segmentation. Our experimental results show that using CDS's yields higher-quality mask predictions compared to methods that have been adopted in the literature for the same purpose.

Keywords: Semantic image segmentation · Weak training set annotations · Dominant sets · Constrained Dominant Sets · Weakly supervised semantic segmentation

1 Introduction

Semantic segmentation is one of the most well-studied research problems in computer vision. The goal is to achieve pixel-level classification, i.e., to label each pixel in a given input image with the class of the object or region that covers it. Predicting the class of each pixel yields to complete scene understanding which is the main problem of a wide range of computer vision applications, e.g. autonomous driving [7], human-computer interaction [15], earth observation [3], biomedical applications [27], dietary assessment systems [2], etc. Stunning performances of DCNNs (Deep Convolutional Neural Networks) at image classification tasks have encouraged researchers to employ them for pixel-level classification as well. Outstanding methods in well-known benchmarks, e.g. PASCAL VOC 2012, train some fully convolutional networks (FCN) with supervision

This work is supported by The Scientific and Technological Research Council of Turkey under TUBITAK BIDEB-2219 grant no 1059B191701102.

E. Ricci et al. (Eds.): ICIAP 2019, LNCS 11752, pp. 425–436, 2019.
https://doi.org/10.1007/978-3-030-30645-8_39

of fully-annotated ground-truth masks. However, obtaining such precise fully-annotated masks is extremely expensive and this limits the availability of large-scale annotated training sets for deep learning architectures. In order to address the aforementioned issue, recent works explored supervision of DCNN architectures for semantic segmentation using low-cost annotations like image-level labels [11], point tags [4], bounding box [8,12,16] and scribbles [13,21,23,26], that are weaker than the pixel-level labels.

Creating weak annotations is much easier than creating full annotations which helps to obtain large training sets for semantic segmentation. However, these annotations are not as precise as full annotations and their quality depends on the decisions made by the users, which degrades their reliability. Hence, literature works proposed different strategies for weakly-supervised semantic segmentation to deal with these issues. While a number of works [21,23] proposed to employ a genuine cost function to get into account only the initially given true weak annotations at the training stage, another and the most common approach [8,12,13,16,26] has been supervising DCNN architectures by *predicted full mask annotations* which are obtained by post-processing the weak-annotations.

Among these two strategies, we follow the second one and propose to generate full mask annotations from scribbles by an interactive segmentation technique which has proven to be extremely effective in a variety of computer vision problems including image and video segmentation [18,28]. For the same purpose, literature works have used a number of shallow interactive segmentation methods, e.g. variants of GrabCut [20] are used in [12,16] for propagating bounding box annotations to supervise a convolutional network. In order to propagate bounding box annotations, [8] proposed to perform iterative optimization between generating full mask approximations and training the network. Using a similar iterative scheme, [13] propagated scribble annotations by superpixels via optimizing a multi-label graph cuts model of [5]. [26] proposed a random-walk based label propagation mechanism to propagate scribble annotations.

In this paper, we aim to explore the potential of *Constrained Dominant Sets* (*CDS*) [28,29] for generating predicted full annotations to be used in supervision of a convolutional neural network for semantic segmentation. Representing images in an edge-weighted graph structure, main idea in constrained segmentation approach in [28] is finding the collection of dominant set clusters on the graph that are constrained to contain the components of a given annotation. CDS approach is applied for co-segmentation and interactive segmentation using modalities of bounding box or scribble and superiority of it over the state of the art segmentation techniques like Graph Cut, Lazy Snapping, Geodesic Segmentation, Random Walker, Transduction, Geodesic Graph Cut, Constrained Random Walker is proved in [28]. Motivated by the reported performance achievements for single cluster extraction (i.e. foreground extraction) in [28], we used CDS for multiple cluster extraction involving multi-label scribbles for the PASCAL VOC 2012 dataset. Since our goal is mainly exploring the performance of CDS in full mask prediction for weakly-supervised semantic segmentation, we trained a basic segmentation network, namely Fully Convolutional Network (FCN-8s) of

[14] based on VGG16 architecture, and compared our performance with other full mask prediction schemes in the literature that supervise the same type of deep learning architecture. Our experimental results on the standard dataset PASCAL VOC 2012 show the effectiveness of our approach compared to existing algorithms.

2 Constrained Dominant Sets

Dominant Set Framework. In the dominant-set clustering framework [17,18], an input image is represented as an undirected edge-weighted graph with no self-loops $G = (V, E, w)$, where $V = \{1, ..., n\}$ is the set of vertices that correspond to image points (pixels or superpixels), $E \subseteq V \times V$ is the set of edges that represent the neighborhood relations between vertices, and $w = E \rightarrow R^*_+$ is the (positive) weight function that represent the similarity between linked node pairs. A symmetric affinity (or similarity) matrix is constructed to represent the graph G that is denoted by $A = (a_{ij})_{n \times n}$ where $a_{ij} = w(i, j)$, if $(i, j) \in E$ and $a_{ij} = 0$ otherwise.

Next, a weight $w_S(i)$, which is (recursively) defined as Eq. 1, is assigned to each vertex $i \in S$,

$$w_S(i) = \begin{cases} 1 & \text{if } |S| = 1, \\ \sum_{j \in S \setminus \{i\}} \phi_{S \setminus \{i\}}(j, i) w_{S \setminus \{i\}}(j), & \text{otherwise.} \end{cases} \tag{1}$$

where $\phi_S(i, j)$ denotes the (relative) similarity between nodes j ($j \notin S$) and i, with respect to the average similarity between node i and its neighbours in S (defined by $\phi_S(i, j) = a_{ij} - \frac{1}{|S|} \sum_{k \in S} a_{ik}$).

A positive $w_S(i)$ indicates that adding i into its neighbours in S will increase the internal coherence of the set, while when it is negative overall coherence gets decreased. Based on aforementioned definitions, a non-empty subset of vertices $S \subseteq V$ such that $\sum_{i \in T} w_T(i) > 0$ for any non-empty $T \subseteq S$, is said to be *dominant set* if it is a maximally coherent data set, i.e. satisfying two basic properties of a cluster that are *internal coherence* ($w_S(i) > 0$, for all $i \in S$) and *external incoherence* ($w_{S \cap \{i\}} < 0$, for all $i \notin S$).

Consider the following linearly-constrained quadratic optimization problem,

$$\begin{aligned} &\text{maximize } f(x) = x' A x \\ &\text{subject to} \quad x \in \Delta \end{aligned} \tag{2}$$

where x' is the transposition of the vector x and Δ is the standard simplex of R^n, defined as $\Delta = \{x \in R^n : \sum_{i=1}^n x_i = 1, \text{ and } x_i \geq 0 \text{ for all } i = 1...n\}$. With the assumption of affinity matrix A is symmetric, it is shown by [17] that if S is an dominant set, then its *weighted characteristic vector* $x^S \in \Delta$ defined as in Eq. 3 is the strict local solution of the Standard Quadratic Program in Eq. 2.

$$x_i^S = \begin{cases} \frac{w_S(i)}{\sum_{j \in S} w_S(j)}, & i \in S \\ 0, & \text{otherwise} \end{cases} \tag{3}$$

Conversely, if x^* is a strict local solution to Eq. 2, then its *support* $\sigma(x^*) = \{i \in V : x_i > 0\}$ is a dominant set of A. Thus, a dominant set can be found by localizing a solution of Eq. 2 by a continuous optimization technique and gathering the support set of the found solution. Notice that the value of a component in the found $x^S \in \Delta$ provides a measure of how strong that component contributes to the cohesiveness of the cluster.

Constrained Dominant Set Framework. In [28,29] the notion of a constrained dominant set is introduced, which aims at finding a dominant set constrained to contain vertices from a given seed set $S \subseteq V$. Based on the edge-weighted graph definition with affinity matrix A, a parameterized family of quadratic programs is defined as in Eq. 4 [28] for the set S and a parameter $\alpha > 0$,

$$\begin{aligned} &\text{maximize } f_S^\alpha(x) = x'(A - \alpha \hat{I}_S)x \\ &\text{subject to } \quad x \in \Delta \end{aligned} \tag{4}$$

where $\hat{I}_S$ is the $n \times n$ diagonal matrix whose elements are set to 1 if the corresponding vertices are in $V \setminus S$ and to 0 otherwise. It is theoretically proven, and empirically illustrated for interactive image segmentation [28], that if S is the set of vertices selected by the user, by setting $\alpha > \lambda_{\max}(A_{V \setminus S})$ it is guaranteed that all local solutions of (4) will have a support that necessarily contains at least one element of S. Here, $\lambda_{\max}$ is the largest eigenvalue of the principal submatrix of A indexed by elements of $V \setminus S$.

In order to find constrained dominant sets by solving the aforementioned quadratic optimization problem (4), [28] used Replicator Dynamics that is developed and studied in evolutionary game theory [17]. In this work we use Infection and Immunization Dynamics (InImDyn) [19] which proved to be a faster and as accurate alternative to it.

3 Proposed Approach

We propose to generate full mask predictions (to be used for supervising a semantic segmentation network) by post-processing weak annotations, i.e. scribble annotations, using CDS. Moreover, we propose to use CDS for multiclass clustering of pixels, i.e. semantic segmentation, while previously CDS has been used only for interactive foreground segmentation [28,29].

3.1 Preprocessing Step for CDS

Superpixel Generation. A common approach followed by image segmentation works has been using superpixels as input entities instead of image pixels. A superpixel is a group of pixels with similar colors and using superpixels not only provides reduced computational complexity, but also yields computing features on meaningful regions. Among a variety of techniques, i.e. SLIC, Oriented Watershed Transform (OWT), we have preferred to use the method developed by Felzenszwalb and Huttenlocher [9] similar to [24] which is a fast and publicly available algorithm. Method of Felzenszwalb and Huttenlocher [9] has also been used in another weakly-supervised semantic segmentation framework [13] experimenting on the same dataset with us. Proposed method in [9] is a graph-based segmentation scheme where a graph is constructed for an image such that each element to be segmented represents a vertex of the graph and dissimilarity, i.e. color differences, between two vertices constitutes a weighted edge. The vertices (or subgraphs) are started to be merged regarding to a merging criteria given in Eq. 5, where e_{ij} is the edge between two subgraphs C_i and C_j, $w(e)$ is the weight on edge e and $\mathrm{MST}(C_x)$ be the minimum spanning tree of C_x.

$$w(e_{ij}) \leq \min_{x \in \{i,j\}} \left(\max_{e \in \mathrm{MST}(C_x)} w(e) + \frac{k}{|C_x|} \right) \tag{5}$$

Here, $\frac{k}{|C_x|}$ is a threshold function in which k is decided by the user, i.e. high values of k yield to lower number of (large) segments, and vice-versa. Another parameter given by the user is the smoothing factor (we denote by σ_{FH}) of the Gaussian kernel that is used to smooth the image at the preprocessing step.

Feature Extraction. Once the superpixels are generated on the image, a feature vector is computed for each superpixel. In the application of CDS model for interactive image segmentation in [28], median of the color of all pixels in RGB, HSV, and L*a*b* color spaces and Leung-Malik (LM) Filter Bank are concatenated in the feature extraction process. Differently from [28], we compute the same feature types with ScribbleSup [13], which has experimented on the same dataset with us, that are color and texture histograms denoted by $h_c(.)$ and $h_t(.)$ in Eq. 6. More specifically, $h_c(x_i)$ is a histogram computed on the color space using 25 bins and $h_t(x_i)$ is a histogram of gradients at the horizontal and vertical orientations where 10 bins are used for each orientation for the superpixel x_i.

3.2 Application of CDS for Full Mask Predictions

In order to generate full mask predictions using the CDS model, an input image is represented as a graph G where vertices depict the superpixels of the image and edge-weights between vertices reflect the similarity between corresponding superpixels. We use scribbles as the given weak annotations in this work which serve as constraints in the CDS implementation. Previously, CDS has been applied for interactive foreground segmentation [28] where dominant set clusters covering a set of given nodes S for a single object class were explored. In this work our problem demand for multiclass clustering of pixels. Hence, here S_c represents the manually selected pixels of the class c where $c \in \{1, ..., C\}$ and C is the number of classes in the dataset, e.g. $C = 21$ for PASCAL VOC 2012.

Accordingly, for each class of scribbles that exist in a given image, by ignoring the existence of the remaining classes in the image we perform foreground segmentation, i.e. 2-class clustering of image pixels, as in [28] by computing its CDS's. Thus, for the class c the union of the extracted dominant sets, i.e.

$UDS_c = D_1 \cup D_2 \cup ...D_L$ if L dominant sets are extracted which contain the set S_c, represents the segmented regions of object in class c. We then repeat this process for every class that exist in the image using the corresponding S_c information. If a node, i.e. superpixel, is found in more than one class of UDS_c, we assign it to the one having the highest value in its weighted characteristic vector $x^{S_c} \in \varDelta$ which is found by solving the quadratic program in Eq. 4 by InImDyn (see Sect. 2).

Computation of the Affinity Matrix. Before computing the CDS clusters, the affinity (or similarity) between superpixels should be computed to construct the matrix A in Eq. 4. In [28], dissimilarity measurements are transformed to affinity space by using the Gaussian kernel $A^{\sigma}_{ij} = \nVdash_{i \neq j} \exp\left(\frac{||f_i - f_j||^2}{2\sigma^2}\right)$, where f_i is the feature vector of the superpixel i, σ is the scale parameter for the Gaussian kernel and $\nVdash_P = 1$ if P is true, 0 otherwise. Differently from [28], we use the Gaussian kernel in Eq. 6 where different σ values are used for different feature types. The kernel in Eq. 6 is also adopted in [13] which experiments on the same dataset and uses the same feature types with us.

$$A^{\sigma_c,\sigma_t}_{ij} = \nVdash_{i \neq j} \exp\left(-\frac{||h_c(x_i) - h_c(x_j)||^2_2}{\sigma^2_c} - \frac{||h_t(x_i) - h_t(x_j)||^2_2}{\sigma^2_t}\right) \qquad (6)$$

Using Different Color Spaces. Quality of generated superpixels effects the performance of the segmentation algorithm directly and a number of segmentation works (examples include but not limited to [1,24]) have emphasized that higher segmentation performances can be obtained by using different color transformations of the input image to deal with different scene and lighting conditions. Motivated by the related literature studies [1,24], we compute superpixels in a variety of color spaces with a range of invariance properties. Specifically, we use five color spaces, that were also used in [24] for determining high quality object locations by employing segmentation as a selective search strategy, that are *Intensity* (grey-scale image), *Lab*, *rgI* which denotes *rg* channels of normalized *RGB* plus intensity, *HSV*, *H* that denotes the Hue channel of *HSV*. We generate superpixels and compute mask predictions using CDS model for each color space of the input image, then we decide the final label for a pixel based on most frequently occurred class label, i.e. by using the scheme of majority voting. In addition to using different color spaces we also vary the threshold parameter k (in Eq. 5) to get benefit from a large set of diversification as recommended in [24].

4 Experiments

Dataset and Evaluation. We trained the models on the 10582 augmented PASCAL VOC training set [10] and evaluated them on the 1449 validation set. We used the scribble annotations published in [13]. In what follows accuracy is evaluated using *pixel accuracy* ($\sum_i n_{ii} / \sum_i t_i$), *mean accuracy* ($(1/n_{cl}) \sum_i n_{ii} / \sum_i t_i$)

and *mean Intersection over Union* ($(1/n_{cl}) \sum_i n_{ii}/(t_i + \sum_j n_{ji} - n_{ii})$ as in [14], where n_{ij} is the number of pixels of class i predicted to belong to class j, n_{cl} is the number of different classes, and $t_i = \sum_j n_{ij}$ be the total number of pixels of class i.

Implementation Details. We used the VGG16-based FCN-8s network [14] of the MatConvNet-FCN toolbox [25] which we initialized by ImageNet pretrained model, i.e. VGG-VD-16 in [25]. We trained by SGD with momentum and, similar to [14], we used momentum 0.9, weight decay of 5^{-4}, mini batch size of 20 images and learning rate of 10^{-3}. With these selected hyperparameters we observed that the pixel accuracy is being converged on the validation set.

Performance of CDS is sensitive to the selection of the σ parameter of the Gaussian kernel (see Sect. 3.2) and in [28] three different results are reported for different selections of σ: (1) *CDSBestSigma*, where best σ is selected separately for every image; (2) *CDSSingleSigma*, by searching in a fixed range, i.e. 0.05 and 0.2; (3) *CDSSelfTuning*, where σ^2 is replaced by $\sigma_i \times \sigma_j$, where $\sigma_i = mean(KNN(f_i))$, i.e. the mean of the K-NearestNeighbor of the sample f_i, K is fixed to 7. To decide values of the σ_c and σ_t parameters (in Eq. 6) we followed *CDSBestSigma* strategy in [28]. Additionally, in the graph structure we cut the edges between vertices correspond to non-adjacent superpixels vertices by setting the corresponding items to zero in the affinity matrix A like has been done in [13], which has provided better segmentation maps. We then min-max normalized the matrix A to be scaled in the range of $[0, 1]$ and symmetrized it.

Performance Evaluation. We first explored the performance using different color spaces on the predicted full annotations of 10582 images (denoted by *PredSet* to mention "Predicted Set" in Table 1), before training the network with them. Then, by training the network with the Predicted Sets we report performance on the Test Set, i.e. PASCAL VOC 2012 Val set. In the implementation of the superpixel generation of [9] we used smoothing factor of $\sigma_{FH} = 0.8$ (*FH* stands for Felzenszwalb and Huttenloche [9]) in the experiments of Table 1. For each color space we performed majority voting (denoted by *MV* both in Tables 1 and 2) over obtained maps with $k = \{225, 250, 300, 400\}$ (in Eq. 5).

We see at Table 1 that using different color spaces affects the quality of the predicted full annotations (*PredSet*) and highest quality mask predictions in terms of mIoU are obtained when we use the Intensity (66.51%). Performing majority voting over maps obtained in all color spaces provided highest quality mask predictions for both CDS (73.28%) and GraphCut (63.51%). We then trained the network with the predicted sets of *CDS-Intensity*, *CDS-MV*, *GraphCut-MV* and published full mask annotations and present their performance on the test set in Table 1. We see that by using CDS-MV in training we outperform GraphCut (which was employed in [13]) significantly and we are quiet approaching to the performance of fully-annotated mask training (59.2% vs. 61.6%).

Comparison with Other Full-Mask Prediction Methods. There is a large variety of interactive segmentation algorithms that can be used for full mask prediction

Table 1. Quality of obtained mask predictions (*PredSet*) and using them in network training performance on the PASCAL VOC 2012 Val set (*TestSet*) (MV: Majority Voting, $^{(*)}$ implementation of GraphCut in our framework.)

Color space	mean IoU	Pixel Acc.	mean Acc.
PredSet-CDS-$Intensity$	66.51	89.05	75.95
PredSet-CDS-Lab	65.47	88.36	76.15
PredSet-CDS-rgI	64.70	88.13	75.29
PredSet-CDS-HSV	66.49	89.27	74.60
PredSet-CDS-H	57.16	85.12	68.21
PredSet-CDS-MV	73.28	91.47	82.05
PredSet-GraphCut$^{(*)}$-MV	63.51	86.48	81.83
TestSet-CDS-Intensity	57.41	89.01	70.56
TestSet-CDS-MV	59.20	89.59	73.05
TestSet-GraphCut$^{(*)}$-MV	52.25	85.80	72.43
TestSet-With Full Masks	61.60	90.27	78.95

to train a semantic segmentation network. To be as fair as possible we make comparison with the reported performances of the methods that are carried on in similar conditions with us, e.g. the ones which employ scribbles as weak annotations, achieve network training using cross entropy loss computed over all pixel predictions but not only on given weak annotations, and do not iterate between the shallow segmentation method and network training with the obtained mask predictions as in ScribbleSup [13]. On the other hand, we performed the Graph Cut algorithm employed in ScribbleSup [13] in our framework by using the published code[1] referred in [13] and present its performance. In fact, our approach can be considered as the first iteration step of such an iterative scheme, and it can be extended to be used in further iterations by updating initial scribble annotations by considering network scores obtained with high confidence.

Considering the above issues we compare with the methods whose accuracy on the test set is reported when their mask predictions are used to train a segmentation network. Specifically, we refer to the performance results of the popular methods GrabCut [20], NormalizedCut [21], and KernelCut [22] reported in [21]. It is mentioned in [21,22] that for each image pixel, RGB (color) and XY (location) features are concatenated to be used in these algorithms. Then, segmentation proposals generated by them are used to train a VGG16-based DeepLab-Msc-largeFOV network [6]. It is reported in [6] that DeepLab-Msc-largeFOV, which employs atrous convolution and multiscale prediction, outperforms FCN-8s by around 9% (71.6% vs. 62.2%) at PASCAL VOC 2012 validation set when trained by full mask annotations, which provides an advantage at comparative works. On the other hand, we also present the performance gap between weak and full mask training to provide a more fair comparison in Table 2. In Table 2,

[1] mouse.cs.uwaterloo.ca/code/gco-v3.0.zip.

the performance results of full mask training (64.1 %), GrabCut [20], NormalizedCut [21], and KernelCut [22] are acquired from [21].

Table 2. Performance comparison on PASCAL VOC 2012 val set.

Method	mIoU	Gap between full and weak supervision
With Full Masks [21]	64.1	
GrabCut [20]	55.5	8.6
NormalizedCut [21]	58.7	5.4
KernelCut [22]	59.8	4.3
With Full Masks	61.6	
GraphCut$^{(*)}$-MV$_{(\sigma_{\text{FH}}=0.8)}$	52.25	9.35
CDS-MV$_{(\sigma_{\text{FH}}=0.8)}$	59.20	2.40
CDS-MV$_{(\sigma_{\text{FHBest}})}$	60.22	1.38

For CDS, we train with mask predictions generated by two different selections of σ_{FH}: *(i)* $\sigma_{\text{FH}} = 0.8$ (corresponding to *PredSet*-CDS-*MV* in Table 1); and *(ii)* σ_{FHBest}, where we selected the best among $\sigma_{\text{FH}} = 0.7$ and $\sigma_{\text{FH}} = 0.8$ for each image. It can be seen at the segmentation performances on the *val* set given in Table 2 that we outperform the literature works at σ_{FHBest} (60.22%), and we are superior at both parameter selections in terms of performance gap between full and weak supervision, i.e. we approach to the performance of our full mask training (61.6%) by 2.4% and 1.38% at selection of $\sigma_{\text{FH}} = 0.8$ and σ_{FHBest}, respectively. Two example images from the generated set, i.e. PredSet, of σ_{FHBest} are presented in Fig. 1. Figure 2 shows examples from testing on the *val* set when it is trained by *PredSet-CDS-MV*$_{\sigma_{\text{FHBest}}}$. It can be seen in Figs. 1 and 2 that our results are the ones most closest to the ground truth of input images.

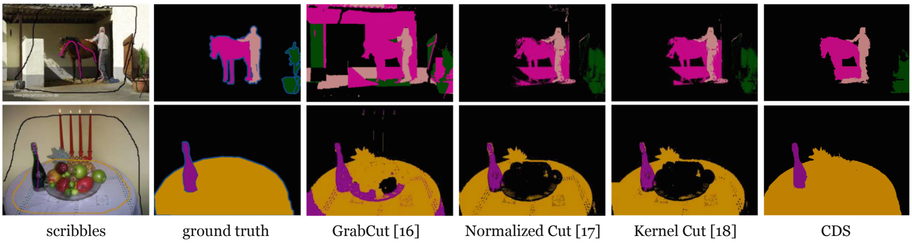

Fig. 1. Generated mask predictions (Images for GrabCut [20], Normalized Cut [21], and KernelCut [22] are acquired from [21])

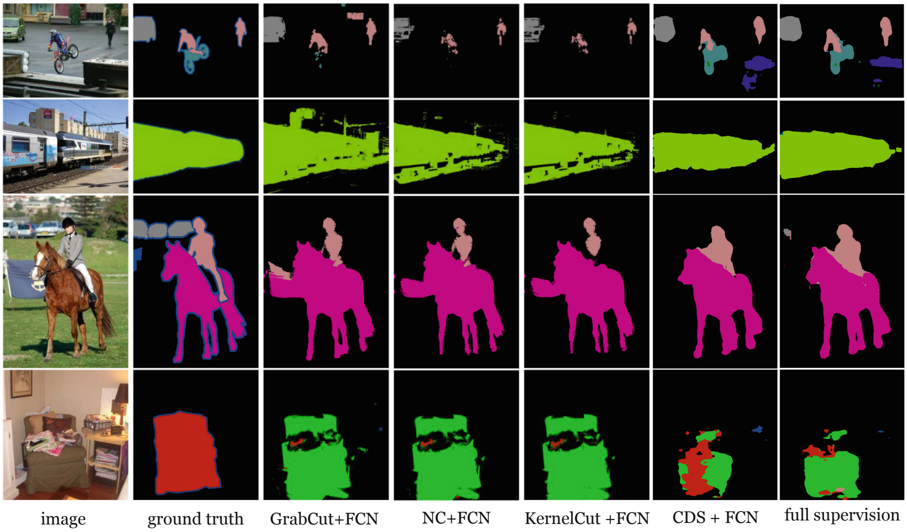

Fig. 2. Testing on PASCAL VOC 2012 *val* set. (Images for GrabCut [20], Normalized Cut [21], and KernelCut [22] are acquired from [21])

5 Conclusions

In this paper we have proposed to apply Constrained Dominant Set (CDS) model, which is proved to be an effective method compared to state-of-the-art interactive segmentation algorithms, for propagating weak scribble annotations of a given set of images to obtain the multi-labeled full mask predictions of them. Achieved mask predictions are then used to train a Fully Convolutional Network for semantic segmentation. While CDS has been applied for pixelwise binary classification problem, it has not been explored for semantic segmentation before and this paper presents our work in this direction. Experimental results showed that proposed approach generates higher quality full mask predictions than the existing methods that have been adopted for weakly-supervised semantic segmentation in literature works.

References

1. Aslan, S., Ciocca, G., Schettini, R.: On comparing color spaces for food segmentation. In: Battiato, S., Farinella, G.M., Leo, M., Gallo, G. (eds.) ICIAP 2017. LNCS, vol. 10590, pp. 435–443. Springer, Cham (2017). https://doi.org/10.1007/978-3-319-70742-6_42
2. Aslan, S., Ciocca, G., Schettini, R.: Semantic food segmentation for automatic dietary monitoring. In: IEEE 8th International Conference on Consumer Electronics - Berlin (ICCE-Berlin) (2018)

3. Audebert, N., Le Saux, B., Lefèvre, S.: Semantic segmentation of earth observation data using multimodal and multi-scale deep networks. In: Lai, S.-H., Lepetit, V., Nishino, K., Sato, Y. (eds.) ACCV 2016. LNCS, vol. 10111, pp. 180–196. Springer, Cham (2017). https://doi.org/10.1007/978-3-319-54181-5_12
4. Bearman, A., Russakovsky, O., Ferrari, V., Fei-Fei, L.: What's the point: semantic segmentation with point supervision. In: Leibe, B., Matas, J., Sebe, N., Welling, M. (eds.) ECCV 2016. LNCS, vol. 9911, pp. 549–565. Springer, Cham (2016). https://doi.org/10.1007/978-3-319-46478-7_34
5. Boykov, Y., Kolmogorov, V.: An experimental comparison of min-cut/max-flow algorithms for energy minimization in vision. IEEE Trans. Pattern Anal. Mach. Intell. **9**, 1124–1137 (2004)
6. Chen, L.C., Papandreou, G., Kokkinos, I., Murphy, K., Yuille, A.: Semantic image segmentation with deep convolutional nets and fully connected CRFs. In: International Conference on Learning Representations (2015)
7. Cordts, M., et al.: The cityscapes dataset for semantic urban scene understanding. In: IEEE Conference on Computer Vision and Pattern Recognition, pp. 3213–3223 (2016)
8. Dai, J., He, K., Sun, J.: BoxSup: exploiting bounding boxes to supervise convolutional networks for semantic segmentation. In: IEEE International Conference on Computer Vision, pp. 1635–1643 (2015)
9. Felzenszwalb, P.F., Huttenlocher, D.P.: Efficient graph-based image segmentation. Int. J. Comput. Vis. **59**(2), 167–181 (2004)
10. Hariharan, B., Arbeláez, P., Bourdev, L., Maji, S., Malik, J.: Semantic contours from inverse detectors. In: IEEE International Conference on Computer Vision (ICCV), pp. 991–998 (2011)
11. Huang, Z., Wang, X., Wang, J., Liu, W., Wang, J.: Weakly-supervised semantic segmentation network with deep seeded region growing. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 7014–7023 (2018)
12. Khoreva, A., Benenson, R., Hosang, J., Hein, M., Schiele, B.: Simple does it: weakly supervised instance and semantic segmentation. In: IEEE Conference on Computer Vision and Pattern Recognition, pp. 876–885 (2017)
13. Lin, D., Dai, J., Jia, J., He, K., Sun, J.: ScribbleSup: scribble-supervised convolutional networks for semantic segmentation. In: IEEE Conference on Computer Vision and Pattern Recognition, pp. 3159–3167 (2016)
14. Long, J., Shelhamer, E., Darrell, T.: Fully convolutional networks for semantic segmentation. In: IEEE Conference on Computer Vision and Pattern Recognition, pp. 3431–3440 (2015)
15. Oberweger, M., Wohlhart, P., Lepetit, V.: Hands deep in deep learning for hand pose estimation. arXiv preprint arXiv:1502.06807 (2015)
16. Papandreou, G., Chen, L.C., Murphy, K.P., Yuille, A.L.: Weakly- and semi-supervised learning of a deep convolutional network for semantic image segmentation. In: The IEEE International Conference on Computer Vision (ICCV) (2015)
17. Pavan, M., Pelillo, M.: Dominant sets and pairwise clustering. IEEE Trans. Pattern Anal. Mach. Intell. **29**(1), 167–172 (2007)
18. Rota Bulò, S., Pelillo, M.: Dominant-set clustering: a review. Eur. J. Oper. Res. **262**(1), 1–13 (2017)
19. Rota Bulò, S., Pelillo, M., Bomze, I.M.: Graph-based quadratic optimization: a fast evolutionary approach. Comput. Vis. Image Underst. **115**(7), 984–995 (2011)

20. Rother, C., Kolmogorov, V., Blake, A.: GrabCut: interactive foreground extraction using iterated graph cuts. In: ACM Transactions on Graphics (TOG), vol. 23, pp. 309–314 (2004)
21. Tang, M., Djelouah, A., Perazzi, F., Boykov, Y., Schroers, C.: Normalized cut loss for weakly-supervised CNN segmentation. In: IEEE conference on Computer Vision and Pattern Recognition (CVPR), Salt Lake City (2018)
22. Tang, M., Marin, D., Ayed, I.B., Boykov, Y.: Normalized cut meets MRF. In: Leibe, B., Matas, J., Sebe, N., Welling, M. (eds.) ECCV 2016. LNCS, vol. 9906, pp. 748–765. Springer, Cham (2016). https://doi.org/10.1007/978-3-319-46475-6_46
23. Tang, M., Perazzi, F., Djelouah, A., Ayed, I.B., Schroers, C., Boykov, Y.: On regularized losses for weakly-supervised CNN segmentation. In: Ferrari, V., Hebert, M., Sminchisescu, C., Weiss, Y. (eds.) ECCV 2018. LNCS, vol. 11220, pp. 524–540. Springer, Cham (2018). https://doi.org/10.1007/978-3-030-01270-0_31
24. Uijlings, J.R., Van De Sande, K.E., Gevers, T., Smeulders, A.W.: Selective search for object recognition. Int. J. Comput. Vis. **104**(2), 154–171 (2013)
25. Vedaldi, A., Lenc, K.: MatConvNet - convolutional neural networks for MATLAB. In: Proceeding of the ACM International Conference on Multimedia (2015)
26. Vernaza, P., Chandraker, M.: Learning random-walk label propagation for weakly-supervised semantic segmentation. In: IEEE Conference on Computer Vision and Pattern Recognition, pp. 7158–7166 (2017)
27. Wang, J., MacKenzie, J.D., Ramachandran, R., Chen, D.Z.: A deep learning approach for semantic segmentation in histology tissue images. In: Ourselin, S., Joskowicz, L., Sabuncu, M.R., Unal, G., Wells, W. (eds.) MICCAI 2016. LNCS, vol. 9901, pp. 176–184. Springer, Cham (2016). https://doi.org/10.1007/978-3-319-46723-8_21
28. Zemene, E., Alemu, L.T., Pelillo, M.: Dominant sets for "constrained" image segmentation. IEEE Trans. Pattern Anal. Mach. Intell. (2018)
29. Zemene, E., Pelillo, M.: Interactive image segmentation using constrained dominant sets. In: Leibe, B., Matas, J., Sebe, N., Welling, M. (eds.) ECCV 2016. LNCS, vol. 9912, pp. 278–294. Springer, Cham (2016). https://doi.org/10.1007/978-3-319-46484-8_17