LEARNING SUPERPIXEL RELATIONS FOR SUPERVISED IMAGE SEGMENTATION

Marco Manfredi *Costantino Grana* *Rita Cucchiara*

Università degli Studi di Modena and Reggio Emilia
Dipartimento di Ingegneria "Enzo Ferrari"
Via Vignolese 905/b, 41125 Modena, Italy

ABSTRACT

In this paper we propose to extend the well known graph cut segmentation framework by learning superpixel relations and use them to weight superpixel-to-superpixel edges in a superpixel graph. Adjacent superpixel-pairs are analyzed to build an object boundary model, able to discriminate between superpixel-pairs belonging to the same object or placed on the edge between the foreground object and the background. Several superpixel-pair features are investigated and exploited to build a non-linear SVM to learn object boundary appearance. The adoption of this modified graph cut enhances the performance of a previously proposed segmentation method on two publicly available datasets, reaching state-of-the-art results.

Index Terms— Image segmentation,Supervised learning

1. INTRODUCTION

Binary segmentation of images, that is, separating foreground objects from the background, is a main topic in computer vision. It can help object recognition techniques, by outlining the contours of the object of interest, or human pose estimation through human silhouette segmentation [1, 2]. Interaction with the user has been used to obtain several successful solutions [3], and it is a main research topic since the very popular GrabCut [4]. Several recent approaches for automatic image segmentation [5, 6] are based on graph cut [7], and range from medical imaging [8, 9] to 3D images [10]. A prior information on the object shape has been used to constrain the segmentation, such as in [11] and [12].

Recently, superpixel segmentation has gathered a lot of attention in the community, due to the high-level reasoning that can be carried out on superpixels [13, 14]. Superpixels have been used to extract high level information such as SIFT statistics [15] or directly as graph nodes [16, 17]. Different algorithm for superpixel computation are employed in the same solution by [18]. The majority of recent approaches are focused on finding effective ways to locate the object in the scene at pixel (or superpixel) level, constructing a *foreground likelihood* [19]. We think that the importance of the relations between neighboring nodes (pixels or superpixels) in the graph cut framework has been underestimated. Modeling the connections between adjacent superpixels using training data, would allow the construction of a graph where binary edges are learned to highlight object boundaries. In such a graph, object segmentation would be much easier, because of the introduction of a new source of information, represented by the learned binary connections.

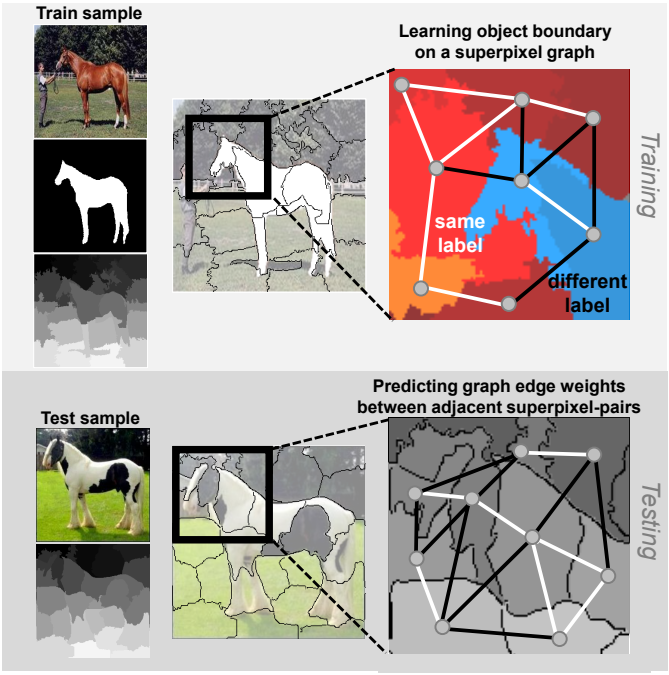

Fig. 1. Superpixel-pair extraction. The training set is used to discriminate between adjacent superpixel-pairs of the same label (either foreground or background) and superpixel-pairs of different labels. At testing time, graph edge weights between neighboring nodes are predicted using a classifier.

The main contribution of the paper is the formulation of a learning scheme able to model the relations between superpixels, applied to supervised binary object segmentation. The concept of *superpixel-pairs* is introduced, representing neighboring superpixel couples in a graph.

The paper is organized as follows: in Section 2 the graph cut framework is introduced and our proposal is presented,

Section 3 describes the baseline segmentation algorithm that we extend with our superpixel graph formulation. Section 4 reports some implementation details useful to reproduce the results obtained in Section 5. Section 6 summarizes the contributions of the paper.

2. GRAPH CUT AND METHOD OVERVIEW

The graph cut optimization approach is a widely used technique for binary image segmentation [7], due to the wide range of energy functions that can be minimized using efficient max-flow algorithms [20, 21]. Given a set of pixel $\mathbf{P}$, the task is to assign to each pixel p a label $l \in \{0, 1\}$. Let f be the set of all label assignments and f_p the label assigned to p. Two types of constraints are used. The unary constraint $D_p(l)$ expresses the likelihood of label l for pixel p. The binary constraint (smoothness term) B_{pq} express the likelihood of adjacent pixels p and q to share the same label. The energy function is:

$$E(f) = \sum_{p \in P} D_p(f_p) + \lambda \sum_{\{p,q\} \in \mathcal{N}} \delta(f_p \neq f_q) B_{pq}, \qquad (1)$$

where $\mathcal{N}$ is the set of neighboring pixels (8-connected), $\delta(\cdot)$ is the indicator function defined as:

$$\delta(x) = \begin{cases} 1 & \text{if} \quad x \text{ is true} \\ 0 & \text{if} \quad x \text{ is false} \end{cases}. \qquad (2)$$

The parameter λ balances the importance of the smoothness term w.r.t. the unary term, the higher the λ, the smoother will be the final segmentation. B_{pq} express the gradient between pixel p and pixel q and usually, for color images, takes the form proposed by Rother *et al.* [4]:

$$B(pq) = \frac{1}{dist(p,q)} \exp\left(-\frac{\|x_p - x_q\|^2}{2\sigma^2}\right), \qquad (3)$$

where $dist(p, q)$ is the distance between pixels and σ is the expectation of the euclidean distance in color space $\|x_p - x_q\|^2$.

2.1. Superpixel Graph

Our proposal, summarized in Figure 1, is to learn the smoothness term of the graph cut, instead of computing it using color gradients like all previous approaches. Instead of learning pixel-to-pixel connections, we propose to learn adjacent superpixel relations. The use of a superpixel graph allows the computation of high level features, such as SIFTs, resulting in an informative representation of superpixel appearance. Inspired by [16] the unary term of superpixel S_i in the graph is computed as the sum of the unary terms of the pixels belonging to S_i.

To describe a superpixel-pair we use several superpixel features:

- isB: a boolean value indicating if the superpixel touches the border of the image;
- isC: a boolean value indicating if the superpixel includes the center of the image;
- pos: a 36-D descriptor indicating the superpixel location, obtained by linearly downsampling the image-size binary mask of the superpixel to 6×6 size;
- bow: the Bag of Words descriptor computed on SIFTs;
- $bowLab$: the Bag of Words descriptor computed on CIEL*a*b* color values;
- bE: an histogram of the gradient magnitude computed on the border between two superpixels, it only considers the gradient component orthogonal to the border direction;

The feature vector of a superpixel S_i is the concatenation of all superpixel features:

$$F(S_i) = [isB, isC, pos, bow, bowLab], \qquad (4)$$

the feature vector of a superpixel-pair Sp_{ij} is:

$$F(Sp_{ij}) = [F(S_i), F(S_j), bE_{ij}]. \qquad (5)$$

To learn the object boundary appearance, a binary non-linear Support Vector Machine (SVM) is used. Since the superpixel-pair feature vector is not symmetric, each sample is taken into account twice, inverting the superpixel order. Superpixel-pairs belonging to the same label are used as positive samples, while superpixel-pairs belonging to different labels are used as negative samples. Positive and negative classes are balanced to obtain an equal number of samples. At testing time, the edges between superpixels in the superpixel graph are weighted using the scores given by the SVM. Graph edges must be positive values, for this reason the SVM is set to produce probabilistic output, giving a score ranging from 0 to 1.

3. KERNEL BASED SEGMENTATION

Our proposal focuses on the smoothness term of the graph cut, thus a unary potential must be defined. To compute the unary potential we chose the segmentation algorithm proposed in [22]. The method tackles the segmentation problem as a structured prediction problem and employs a one-class SVM working with joint kernels between image-mask pairs.

The main idea is to exploit one-class SVMs in a kernel space to learn a set of support vectors and their relative weights, also deleting outliers from the training set. Given a training set of sample-label pairs $(x_1, y_1), ..., (x_n, y_n)$, the probability density function $p(x, y)$ is modeled and

$f(x) = \arg\max p(x, y)$ is used for prediction. This maximization can be done using graph cuts as demonstrated in [6]. Assuming that $p(x, y)$ is high only if y is a correct label for x, only the support of $p(x, y)$ is needed [23]. This can be effectively obtained by a one-class support vector machine (OC-SVM). We can express $p(x, y)$ as:

$$p(x, y) = \exp(\mathbf{w}^\top \phi(x, y)). \tag{6}$$

It is difficult to find an explicit formulation of $\phi(x, y)$, while it is easier to find a suitable joint kernel K that matches two sample-label pairs. The output of the OC-SVM learning process becomes a linear combination of kernel evaluations with training samples, thus the prediction function can be formulated as:

$$f(x) = \arg\max_{y \in Y} \sum_{i=1}^{n} \alpha_i K\left((x, y), (x_i, y_i)\right). \tag{7}$$

The selected support vectors are the training samples that have non zero α.

3.1. Joint Kernels

Joint kernels are computed on two image-mask pairs, and express the degree of similarity between two segmentations. The selected kernel is the product of an image kernel and a mask kernel:

$$K((x_i, y_i), (x_j, y_j)) = \theta(x_i, x_j) \cdot \Omega(x_i, x_j, y_i, y_j), \tag{8}$$

The image similarity kernel θ computes the similarity between two images using HOG descriptors [24], although many other descriptors could be used without changing the model [25, 26]. The mask similarity kernel Ω takes into consideration both images and masks to extract knowledge about how comparable two segmentations are. The mask kernel is composed of a linear combination of three parts, leveraging color information on the RGB color space and shape information of the binary masks y_i and y_j. For more details refer to [22].

4. IMPLEMENTATION DETAILS

Our proposal does not depend on a specific superpixel extraction algorithm, therefore we chose a well-known state-of-the-art method: the SLIC superpixels [13]. The number of superpixel extracted from each image is set to 200 by experimental evaluation. Too many superpixels would lead to a finer segmentation, but at the cost of a poorer superpixel representation, due to the limited amount of pixels involved. The number of bins of the bow,$bowLab$ and bE superpixel features was set to 32 by maximizing superpixel classification accuracy on a validation set. An higher number of bins produces features that are too sparse and leads to poor results. The SVM for superpixel-pair learning is trained using an RBF kernel.

The parameters of the segmentation algorithm of Sec. 3 have been optimized on a validation set with grid search.

5. EXPERIMENTAL RESULTS

We compared the proposed method with several approaches on two publicly available datasets, containing images of flowers and horses. The first is the Weizmann horses dataset [27], that contains 328 images of horses with strong differences in background, contrast and pose. The second is the Oxford flower dataset [28], composed of 753 images of flowers belonging to different species.

We compare our performance with the following methods:

1. Segmentation without superpixels [22]: the starting method on which our proposal has been built (see Section 3). It uses a standard graph cut smoothness term, based on color gradients between neighboring pixels;

2. KSSVM [6]: Kernelized Structured Support Vector Machines, it uses structural learning in a discriminative solution;

3. Grabcut [4]: initialized using the average of the masks of the k nearest images found with the image similarity kernel (see Section 3);

4. Flower shape model [28]: strictly related to the domain of flowers, it exploits a flower shape model made of center and petals.

All the images are resized to the same dimension to allow the kernel computation, the chosen size is 256×256 for both the datasets. We split each dataset in three parts and performed training on the first part, parameters optimization on the second part and testing on the third part; eventually we exchanged the parts and averaged the results (three tests are conducted for each experiment). We used two metrics to evaluate the segmentation performance: the overall pixel accuracy P_{acc}, that measures the percentage of correctly labeled pixels, and the intersection-over-union metric IoU, defined as the intersection of the output mask and the ground truth mask divided by the union of the two masks.

$$\begin{aligned} P_{acc} &= \frac{1}{P} \sum_{p=1}^{P} \delta(M_p = M_{GTp}) \\ IoU &= \frac{\sum_{p=1}^{P} M_p = \text{“}obj\text{”} \wedge M_{GTp} = \text{“}obj\text{”}}{\sum_{p=1}^{P} M_p = \text{“}obj\text{”} \vee M_{GTp} = \text{“}obj\text{”}} \end{aligned} \tag{9}$$

where M is the output mask of the system, M_{GT} is the ground truth mask and $\delta(\cdot)$ is the indicator function defined in (2).

Table 1. Segmentation performance on one run of the Weizmann Horses dataset. Segmentation accuracy IoU is reported along with the superpixel-pair classification accuracy SPP_{acc} for different superpixel-pair feature combinations.

SPP Features	$SPP_{acc}(\%)$	$IoU(\%)$
w/o superpixel	-	74.7
isB, isC, pos	63.1	71.0
isB, isC, pos, bE	79.0	74.3
isB, isC, pos, bE, bow	85.8	77.1
$isB, isC, pos, bE, bow, bowLab$	88.3	77.4
Ideal Classification	100.0	86.5

Table 2. Performance comparison on the Weizmann horses dataset.

Horses Dataset	$P_{acc}(\%)$	$IoU(\%)$
Our Proposal	93.72	78.31
KSSVM + Hog feature [6]	93.9	77.9
Manfredi et al. [22]	93.04	76.32
GrabCut init. with 1-NN mask	85.66	62.34
GrabCut init. with 5-NN masks	86.93	63.83
GrabCut init. with 10-NN masks	86.46	63.20

When using superpixel-pair learning in the graph cut framework, the final segmentation accuracy strongly depends on the accuracy of the superpixel-pair classification. Having a 100% accurate classification would lead to a graph where adjacent superpixels belonging to the same label (foreground or background) would be connected tighter then superpixels of different labels. To highlight the contribution of each superpixel-pair feature, we report in Table 1 the superpixel-pair classification accuracies SPP_{acc} for several feature combinations. In the same table the segmentation performance on one run of the Weizmann horses dataset w.r.t. each feature combination is also reported. Two special cases are also reported: segmentation without superpixels and segmentation using a manually provided 100% correct superpixel-pair classification, to be used as an upper bound. The results clearly show that a poor superpixel-pair representation, made of location information only (isB,isC,pos), does not provide an effective discrimination between same-label and different-label superpixel-pairs. This leads to a IoU measure that is even lower than the w/o superpixel case (71.0 vs 74.7). The most informative features seem to be the Bag of Words on SIFT descriptors (bow) and the border edge magnitude histogram (bE). The complete solution, able to correctly classify 88.3% of superpixel-pairs, outperforms the original segmentation method (77.4 vs 74.7).

Our proposal is compared to other solutions on Weizmann horses dataset in Table 2. The improvement in performance between the original proposal and our method is significant, and highlights the importance of learning the smoothness term in the graph cut framework. The complete solution is able to outperform all the Grabcut baselines and to obtain slightly better results than KSSVMs.

Table 3. Performance comparison on the Oxford flowers dataset.

Flower Dataset	$P_{acc}(\%)$	$IoU(\%)$
Our Proposal	97.51	92.85
KSSVM + Hog feature [6]	97.66	92.33
Manfredi et al. [22]	97.17	92.14
Flower shape model [28]	-	94

On the Oxford flowers dataset, the performance improvement is less significant. This behavior can be explained measuring the upper bound on the segmentation accuracy reachable by the superpixel segmentation, that is clearly lower than the pixel-wise segmentation case, due to misalignment between the superpixel edges and the object boundaries. Using 200 superpixels, as in our experiments, the IoU upperbound is 95.56% (compared to 100% of the pixel graph case). All the methods are able to reach IoU metrics of about 90% and this strongly limits the performance gain that our approach can give. The main insight is that our proposal is effective where the segmentation performance is not saturated, leaving space for improvements.

6. CONCLUSION

An extension of the graph cut segmentation framework is proposed in this paper. Using a graph composed of superpixels instead of pixels, allows us to modify how neighboring nodes in the graph are connected. We propose to learn superpixel-pair relations and use the probabilistic output of an SVM to weight graph edges. We showed that, when superpixels are able to correctly separate foreground and background, it is possible to leverage their pairwise properties to chose if they should be kept together or not. This information is indeed effective in providing a segmentation boost within the graph cut framework.

7. REFERENCES

[1] M. Manfredi, C. Grana, S. Calderara, and R. Cucchiara, "A Complete System for Garment Segmentation and Color Classification," *Mach. Vision Appl.*, vol. 25, no. 4, pp. 955–969, May 2014.

[2] C. Grana, D. Borghesani, and R. Cucchiara, "Automatic Segmentation of Digitalized Historical Manuscripts," *Multimed. Tools Appl.*, vol. 55, no. 3, pp. 483–506, Dec. 2011.

[3] V. Lempitsky, P. Kohli, C. Rother, and T. Sharp, "Image Segmentation with a Bounding Box Prior," in *IEEE Int. Conf. Comput. Vision*, 2009, pp. 277–284.

[4] C. Rother, V. Kolmogorov, and A. Blake, ""GrabCut": interactive foreground extraction using iterated graph cuts," in *ACM SIGGRAPH Papers*, 2004, SIGGRAPH '04, pp. 309–314.

[5] V. Lempitsky, A. Blake, and C. Rother, "Branch-and-Mincut: Global Optimization for Image Segmentation with High-Level Priors," *J. Math. Imaging Vis.*, vol. 44, pp. 315–329, 2012.

[6] L. Bertelli, T. Yu, D. Vu, and B. Gokturk, "Kernelized structural SVM learning for supervised object segmentation," in *IEEE Int. Conf. Comput. Vis. Pattern Recognit.*, June 2011, pp. 2153–2160.

[7] Y.Y. Boykov and M.-P. Jolly, "Interactive Graph Cuts for Optimal Boundary & Region Segmentation of Objects in N-D Images," in *IEEE Int. Conf. Comput. Vision*, 2001, vol. 1, pp. 105–112.

[8] S. Andrews, C. McIntosh, and G. Hamarneh, "Convex Multi-Region Probabilistic Segmentation with Shape Prior in the Isometric Log-Ratio Transformation Space," in *IEEE Int. Conf. Comput. Vision*, 2011, pp. 2096–2103.

[9] A. Lucchi, K. Smith, R. Achanta, G. Knott, and P. Fua, "Supervoxel-Based Segmentation of Mitochondria in EM Image Stacks With Learned Shape," *IEEE Trans. Med. Imaging*, vol. 31, no. 2, pp. 474–486, 2012.

[10] X. Chen, J.K. Udupa, U. Bagci, Y. Zhuge, and J. Yao, "Medical Image Segmentation by Combining Graph Cuts and Oriented Active Appearance Models," *IEEE Trans. Image Process.*, vol. 21, no. 4, pp. 2035–2046, Apr. 2012.

[11] O. Veksler, "Star Shape Prior for Graph-Cut Image Segmentation," in *Eur. Conf. Comput. Vision*, 2008, vol. 5304 of *LNCS*, pp. 454–467.

[12] N. Vu and B.S. Manjunath, "Shape Prior Segmentation of Multiple Objects with Graph Cuts," in *IEEE Int. Conf. Comput. Vis. Pattern Recognit.*, June 2008, pp. 1–8.

[13] R. Achanta, A. Shaji, K. Smith, A. Lucchi, P. Fua, and S. Susstrunk, "SLIC Superpixels Compared to State-of-the-Art Superpixel Methods," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 34, no. 11, pp. 2274–2282, 2012.

[14] A.P. Moore, S. Prince, J. Warrell, U. Mohammed, and G. Jones, "Superpixel Lattices," in *IEEE Int. Conf. Comput. Vis. Pattern Recognit.*, 2008, pp. 1–8.

[15] Y. Chai, V. Lempitsky, and A. Zisserman, "BiCoS: A Bi-level Co-Segmentation Method for Image Classification," in *IEEE Int. Conf. Comput. Vision*, 2011, pp. 2579–2586.

[16] V.S. Lempitsky, A. Vedaldi, and A. Zisserman, "A Pylon Model for Semantic Segmentation," in *Neural Inf. Process. Syst.*, 2011, pp. 1485–1493.

[17] B. Fulkerson, A. Vedaldi, and S. Soatto, "Class Segmentation and Object Localization with Superpixel Neighborhoods," in *IEEE Int. Conf. Comput. Vision*, 2009, pp. 670–677.

[18] Zhenguo Li, Xiao-Ming Wu, and Shih-Fu Chang, "Segmentation Using Superpixels: A Bipartite Graph Partitioning Approach," in *IEEE Int. Conf. Comput. Vis. Pattern Recognit.*, 2012, pp. 789–796.

[19] M. Marszatek and C. Schmid, "Accurate Object Localization with Shape Masks," in *IEEE Int. Conf. Comput. Vis. Pattern Recognit.*, 2007, pp. 1–8.

[20] V. Kolmogorov and R. Zabin, "What Energy Functions Can Be Minimized via Graph Cuts?," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 26, no. 2, pp. 147–159, Feb. 2004.

[21] Y. Boykov and V. Kolmogorov, "An Experimental Comparison of Min-Cut/Max-Flow Algorithms for Energy Minimization in Vision," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 26, no. 9, pp. 1124–1137, Sept. 2004.

[22] M. Manfredi, C. Grana, and R. Cucchiara, "Learning Graph Cut Energy Functions for Image Segmentation," in *Int. Conf. Pattern Recognit.*, 2014, in press.

[23] C.H. Lampert and M.B. Blaschko, "Structured prediction by joint kernel support estimation," *Mach. Learn.*, vol. 77, pp. 249–269, 2009.

[24] N. Dalal and B. Triggs, "Histograms of Oriented Gradients for Human Detection," in *IEEE Int. Conf. Comput. Vis. Pattern Recognit.*, June 2005, vol. 1, pp. 886–893.

[25] M. Bertini, A. Del Bimbo, G. Serra, C. Torniai, R. Cucchiara, C. Grana, and R. Vezzani, "Dynamic Pictorially Enriched Ontologies for Video Digital Libraries," *IEEE MultiMedia*, vol. 16, no. 2, pp. 41–51, Apr. 2009.

[26] G. Serra, C. Grana, M. Manfredi, and R. Cucchiara, "Modeling Local Descriptors with Multivariate Gaussians for Object and Scene Recognition," in *ACM Int. Conf. Multimedia*, Barcelona, Spain, Oct. 2013, pp. 709–712.

[27] E. Borenstein, E. Sharon, and S. Ullman, "Combining Top-Down and Bottom-Up Segmentation," in *IEEE Int. Conf. Comput. Vis. Pattern Recognit. Workshops*, 2004, vol. 4, p. 46.

[28] M.E. Nilsback and A. Zisserman, "Delving deeper into the whorl of flower segmentation," *Image Vision Comput.*, vol. 28, no. 6, pp. 1049–1062, June 2010.