Classification of Segmented Objects through a Multi-net Approach

Alessandro Zamberletti, Ignazio Gallo, Simone Albertini, Marco Vanetti, and Angelo Nodari

University of Insubria
Dipartimento di Scienze Teoriche ed Applicate
via Mazzini 5, Varese, Italy

Abstract. The proposed model aims to extend the MNOD algorithm adding a new type of node specialized in object classification. For each potential object identified by the MNOD, a set of segments are generated using a *min-cut* based algorithm with different seeds configurations. These segments are classified by a suitable neural model and then the one with higher value is chosen, in agreement with a proper energy function. The proposed method allows to segment and classify each object simultaneously. The results showed in the experiment section highlight the potential and the cost of having unified segmentation and classification in a single model.

Keywords: object segmentation, object classification, neural networks, minimum cut.

1 Introduction

Many computer vision problems can be reconducted to the identification and recognition of an object as belonging to a class of objects. In some cases, however, it is fundamental to segment the object before classifying it, in order to be able to extract the relevant characteristics from the object. Given an image that can contain objects belonging to different classes, the generic problem of class segmentation consists in the prediction of the membership class of each pixel, or to map a pixel to the "background" if the pixel does not belong to one of the given classes. The most common approach applied for the object segmentation is based on sliding windows [1,2]; other recent works try to solve the problem by classifying a set of segments through a bottom-up approach, towards the classification of the entire object [3,4]. In this work we want to join the two techniques aforementioned in a single model, with the aim to use both methods in a synergistic way.

The Multi-Net for Object Detection (MNOD) [2] is an algorithm which relies on multiple neural networks to deal with the object detection problem, through an image segmentation approach. This model is organized as a directed acyclic graph where each internal node is a neural network: the source nodes extract the features from the input image, and the output of a node becomes the input of

N. Mana, F. Schwenker, and E. Trentin (Eds.): ANNPR 2012, LNAI 7477, pp. 151–162, 2012.

another neural network in a feedforward manner. The output produced by the last node is the final result of the whole algorithm.

The concept that underlies this model is that the prediction performed by a node, given to the following node, should lead to an improvement as if it were a refinement process, and also leads to an improvement in the generalization ability of the whole system, similarly to [5].

The goal of this work is the realization of an algorithm able to segment and classify all the objects of interest present in an image. The segmentation maps produced by the MNOD are used in order to detect several regions of interest (ROI's) inside the processed image. They are detected following a pyramidal approach, in order to maximize the probability that each object of interest is completely surrounded by a ROI boundary. The object in each ROI is segmented using the MinCut algorithm [6] in combination with an energy function inspired by the Boykov function [7]. A set of seeds for the alleged object and the background must be chosen in order to perform the segmentation with the MinCut algorithm; in this work we compared different strategies, some of these exploit the information from the MNOD segmentation masks to better identify the seeds. It is possible to obtain different segmentations for each ROI just by running the MinCut algorithm with different seed extraction strategies. The segmentations obtained from each ROI are processed by a neural model that predicts, for each of them, the degree of membership to each class. Consequently, after a specific ranking algorithm, the class for the object surrounded by each ROI, and the segment that best represents the separation between the object and its background is chosen. The algorithm we devised exploits the information from the segmentation masks, produced by the MNOD model, to limit the number of segments and ROIs that are generated, and then performs a classification of each object.

Section 2 briefly presents the MNOD model on which this work is based, while section 3 presents the multiclass extension proposed in this paper.

2 The Existing Method: Multi-Net Object Detection

The MNOD is a Multi-Net System [8] which consists of an ensemble of supervised neural networks able to detect an object in a cognitive manner, locating the object through the use of a segmentation process. The MNOD is tolerant to many of the problems that usually afflict the images in a real scenario and also has a high generalization ability and good robustness [2].

The MNOD can be represented by a directed acyclic graph, composed by multiple source nodes (which have no incoming edges) used as feature extractors, internal nodes that aggregate the output of other nodes and a single terminal node (which has no outgoing edges) that provides the final segmentation map. Each node n is properly configured with its own parameters $\mathbf{P}$ and acts like an independent module $C^n_{\mathbf{P}}$ providing a segmentation map as result. The process starts from the source nodes, which apply operators and filters on the input images in order to generate feature-images which enhance the peculiarities of the input data. Internal nodes process the maps provided by other input nodes firstly

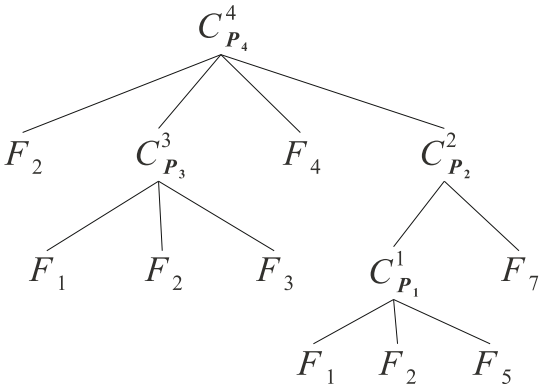

Fig. 1. Generic structure of the MNOD algorithm. The nodes $C^m_{\mathbf{P}}$ represent the supervised neural models which receive their input directly from the source nodes F_i and/or other internal nodes $C^m_{\mathbf{P}}$.

resizing them according to the parameter I_S. After the resize, internal nodes generate the pattern vectors for the neural network using pixel values that fall within a sliding window of size W_S and gives as output a map image where each pixel has an intensity value proportional to the probability that it belongs to the object. In the present work, each internal node consists of a feed-forward Multi-Layer Perceptron (MLP) trained using the Resilient Backpropagation learning algorithm proposed by Riedmiller and Braun [9].

The particular aspect of this model lies in the connections among the nodes, since the links among the nodes in the structure define the flow of the image segmentation process that passes through the whole structure from the source nodes to the terminal node containing the final segmentation. Source nodes extract different information from the input images; as an example we may use color channels and brightness (Br) for the luminance of a visual target. As suggested in [10], information on edges were extracted using 1-D $[-1, 0, 1]$ masks at $\sigma = 0$, obtaining the two source nodes: Horizontal Edges (HE_s) and Vertical Edges (VE_s), where s is the scale of the input image used as a resize factor to compute the feature.

The MNOD configuration is chosen considering which nodes in which layer gave better results using a trial and error procedure. In particular, we use the following strategy to optimize the results: nodes belonging to the first layer are used to find areas of interest, and nodes of subsequent layers are used to eliminate the false positives and to confirm the true positives.

3 The Proposed Method: Multi-Net Object Detection with Object Classification

One characteristic of the MNOD model presented in the previous section is the possibility to change the node type for each $C^m_{\mathbf{P}}$. The only requirement is that

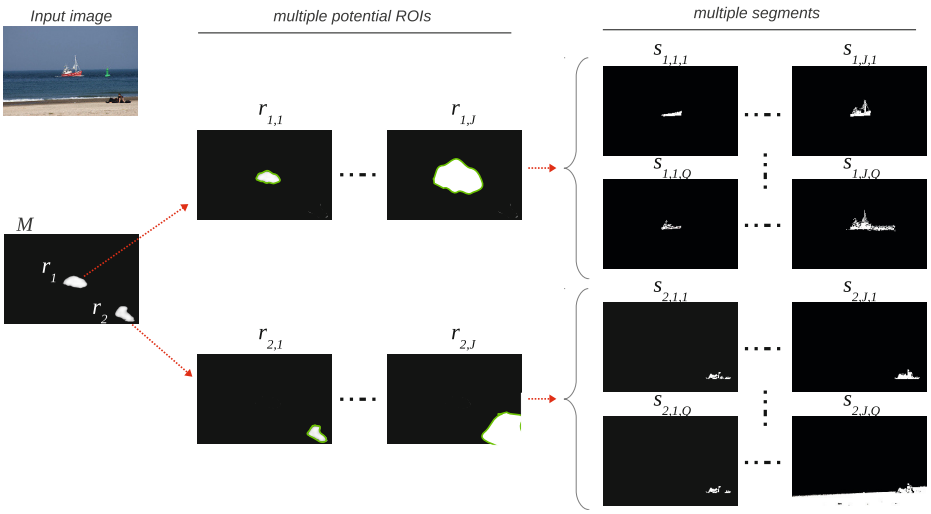

Fig. 2. An example of the segments $s_{k,j,q}$ generated from an input image and an output MNOD map M

every node must implement a common interface. In this work we exploited this feature by integrating a new node $\hat{C}^n_{\mathbf{P}}$ able to classify the potential input ROI into different classes of objects.

More formally, let $\mathbb{C} = \{c_1, \ldots, c_N\}$ be the set of object classes for the classification problem; a node $\hat{C}^n_{\mathbf{P}}$ generates as output a set $\mathbb{I} = \{i_1, \ldots, i_{N+1}\}$ of images, such that i_k is the segmentation mask of all the objects belonging to the class c_k or the background. In order to be able to perform this task, a node $\hat{C}^n_{\mathbf{P}}$ must first generate a set of segmentations of potential objects of interest and then choose which one of these represents the best segmentation. After that, it must select an approriate class for each potential object of interest.

These two steps are described in detail in the following sections 3.1 and 3.2.

3.1 Multiple Segments Creation

A constraint of the proposed strategy imposes only one node of type $\hat{C}^n_{\mathbf{P}}$ as a terminal node that receives as input a single node of type $C^n_{\mathbf{P}}$. This constraint is adopted in order to simplify the algorithm, but it is not mandatory.

Starting from a segmentation map M, representing the input of the node $\hat{C}^n_{\mathbf{P}}$, the algorithm first identifies all the ROI's $R = \{r_1, \ldots, r_K\}$ for a given input image I. Each r_k corresponds to a connected region identified as in the example showed in figure 2. If the child node worked properly, every r_k contains exactly one of all the objects of interest contained in I. However, the identified r_k may only contain partially the object; this situation can lead to a non-optimal segmentation of the object itself. To maximize the probability that a r_k entirely encloses the object of interest, a set of new ROI's $r_{k,j}$ were generated dilating

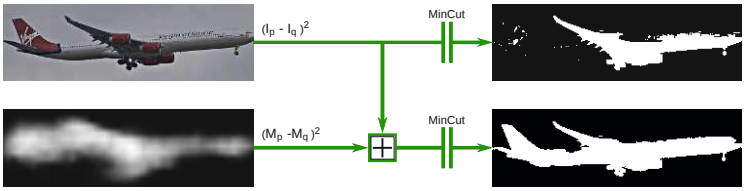

Fig. 3. Comparison between the segmentation results obtained using the energy function proposed by Boykov and our modified version, obtained by introducing an additional energy term that exploits the information contained into the MNOD map M

r_k J times with a mathematical morphology dilation operator, obtaining a set $R_{MP} = \{r_{1,1} \ldots, r_{K,J}\}$ of multiple potential ROI's. For each $r_{k,j}$ we finally generated a set of segments $S = \{s_{k,j,1}, \ldots, s_{k,j,Q}\}$ using a modified version of the Boykov and Komolgorov *min-cut* minimization algorithm [11]. In details, Q corresponds to the number of different strategies, used to initialize the seeds of the *min-cut* algorithm. The same strategy can be applied more than one time, using different parameters.

We modified the energy function proposed by Boykov by adding a new term $T_M = (M_p - M_q)^2$, derived from the segmentation map M, as defined in (1). The use of this additional term helps to ensure good quality segmentations even if the object to be segmented has a color similar to that of the background which surrounds it, as in the example shown in figure 3.

$$V_{p,q} \propto exp\left(-\frac{(I_p - I_q)^2 + (M_p - M_q)^2}{2\sigma^2}\right)\frac{1}{dist(p,q)} \tag{1}$$

To segment each $r_{k,j}$ using the *min-cut* algorithm, a node $\hat{C}^n_{\mathbf{P}}$ defines a set of *seed* pixels P_{obj} belonging to the potential object, and a set of seed pixels P_{bg} belonging to the background, based on information derived from the bounding box of the same ROI $r_{k,j}$. In this paper, we used $Q = 5$ different strategies for the creation of all the seeds (see example in figure 4). In this way we can create a set of possible segments $\{s_{k,j,1}, \ldots, s_{k,j,Q}\}$ for each $r_{k,j}$, as summarized in the example shown in figure 2. Below are briefly presented all the algorithms used to create the seeds. Many of these algorithms exploit the soft segmentation map M, identifying the minimum intensity value of p_{min}, and the maximum p_{max}, in order to extract the seeds.

Random. This strategy extracts a random set of seed P_{obj} from a neighborhood of p_{max} and the set of seed P_{bg} from a neighborhood of p_{min}. The cardinality of the two sets of seeds is a parameter and is such that $|P_{obj}| = |P_{bg}|$.

MaxMin. Similar to the random strategy, but it selects as seeds P_{obj} all the pixels that have value exactly equal to p_{max}, while selects as seeds P_{bg} all the pixels that have value exactly equal to p_{min}.

Distance. The selection of seeds P_{obj} is exactly equal to the random strategy, while to select the seeds P_{bg} this strategy also maximizes the Euclidean distance from the centroid of the P_{obj} set.

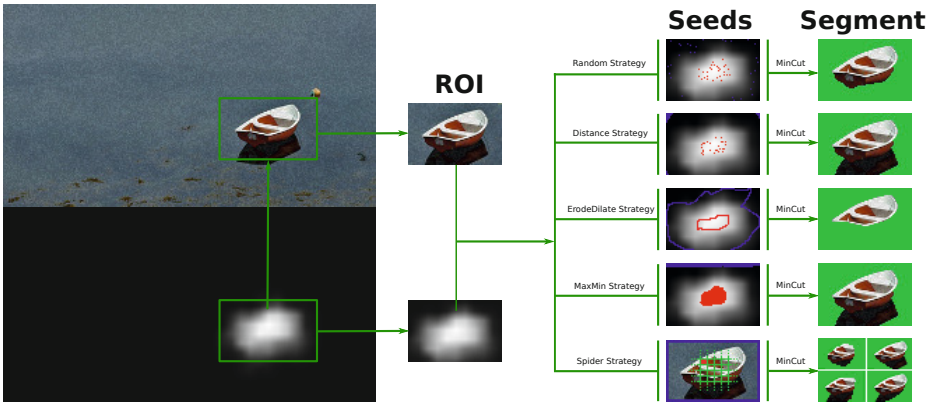

Fig. 4. Example of the segments produced by the *min-cut* algorithm, varying the algorithm that generates the set of seeds

ErodeDilate. This strategy applies the morphological operator erosion to the ROI at $r_{k,j}$ and takes all the edge pixels as belonging to the set of seeds P_{obj}, while to obtain the seeds P_{bg} applies the morphological operator dilation still to the ROI $r_{k,j}$.

Spider. This strategy operates in a completely different way and is inspired by the seed selection technique used in [12]. The pixels that are located near the edges of the ROI $r_{k,j}$ are marked as belonging to P_{bg}; the internal portion of $r_{k,j}$ is divided into N rectangles $\{f_1, \ldots, f_N\}$ of equal size, for each of which a set of seeds P_{obj} is extracted. In this way we obtain a segmentation for each rectangle f_n.

3.2 Classification of the Winner Segment

The final step of a node $\hat{C}^n_{\mathbf{P}}$ is to decide what is the segment that best replaces a ROI r_k, and what is the class of this segment. To do this, a node $\hat{C}^n_{\mathbf{P}}$ uses a trained MLP neural network, which receives as input a set of features extracted from each segment $s_{k,j,q}$, and generates as output N values $\langle o_{1,j,q}, \ldots, o_{N,j,q} \rangle$. Each value $o_{n,j,q}$ is the degree of membership of the segment $s_{k,j,q}$ to the class $c_n \in \mathbb{C}$. An input pattern for this neural model is constructed by combining the following descriptors for each segment $s_{k,j,q}$ processed: PHOG [13], Gray Level Histogram, Area and Perimeter of the segment. These features were chosen after a careful analysis, documented in the section 4.

Starting from the set of predictions made by the neural model on the set of segments $\{s_{k,j,1}, \ldots, s_{k,j,Q}\}$, we determine the class of the object under consideration (eq. 2) as the index w of the first component with maximum value, adding up all the Q predictions:

$$w = \arg\max_{n=1,\ldots,N} \sum_{j=1}^{J} \sum_{q=1}^{Q} o_{n,j,q} \tag{2}$$

After identifying a proper class for the ROI r_k, we choose the best segment $s_{k,j,q} \quad \forall j, q$. In order to do that, all the candidate segments are submitted as input to the trained neural model. The segment $s_{k,j,q}$ is the winner if it generates an output component $o_{w,j,q}$ having a maximum value when compared with all other $J \times Q$ values.

4 Experiments

In this section we evaluate two aspects of our algorithm: the best combination of features for object classification in a real context, and the analysis of the proposed $\hat{C}^n_{\mathbf{P}}$ node when integrated in the existing MNOD model. To assess the performance of the classification and the segmentation phases, we employed the following evaluation metric:

$$\mathrm{OA} = \frac{\mathrm{TP}}{\mathrm{TP} + \mathrm{FP} + \mathrm{FN}} \tag{3}$$

where the true positives (TP), false positives (FP) and false negative (FN) are computed considering the whole image (obtaining the overall classification accuracy OA_c), or considering the individual pixels (obtaining the overall segmentation accuracy OA_s). For all the experiments we used a computer having the following configuration: a single C# thread, on an Intel®Core ™i5 CPU at 2.67GHz.

In the first experiment the most successful features for image classification were analyzed independently: PHOG [13], PHOW [14], Geometric Blur [15] and Visual Self-Similarity [16]. We employed the Drezzy-46[1] dataset, composed by images representing clothing in the context of online shopping, firstly presented in [17]. The number of images per class varies from a minimum of 52, to a maximum of 227. The dataset contains 4841 images, divided into 46 classes, having a resolution equals to 200x200. Our goal was to determine the most effective feature for the classification of objects belonging to the Drezzy-46 dataset. Varying the features parameters we obtained the highest OA_c using the PHOG feature computed on the segments $s_{k,j,q}$.

The next step attempts to combine the PHOG feature with other simpler features, in order to try to increase the classification accuracy. In particular, we tried to combine the segment area (A), segment perimeter (P) and gray level histogram (GLH) features. Table 1 shows that the use of the feature PHOG in combination with other features allows to achieve a slight increase in terms of accuracy of classification without increasing the computational time. This result allowed us to determine the final configuration: PHOG with 3 layers and 6 bins for each histogram, P, A and GL.

In the second experiment conducted, we analyzed the segmentation accuracy OA_s, using two datasets of manually segmented real images. As the first dataset we employed the DrezzyDataset[2], proposed in [18], considering only three classes:

[1] The dataset is available online at: `http://artelab.dicom.uninsubria.it/download/`
[2] The dataset is available online as: `http://artelab.dicom.uninsubria.it/download/`

Table 1. OA_c computed on the dataset Drezzy-46, using different combinations of features for the patterns creation

Feature	Test OA_c	Time (ms)
PHOG 2L/06B	0,79	5,0
PHOG 3L/06B	0,82	5,4
PHOG 3L/15B	0,82	6,8
PHOG 2L/06B ∪ A ∪ P ∪ GLH	0,79	4,5
PHOG 3L/06B ∪ A ∪ P ∪ GLH	**0,83**	5,7
PHOG 3L/15B ∪ A ∪ P ∪ GLH	0,81	7,3
PHOG 2L/06B ∪ P ∪ GLH	0,79	5,1
PHOG 3L/06B ∪ P ∪ GLH	0,82	5,6
PHOG 3L/15B ∪ P ∪ GLH	0,80	7,3
PHOG 2L/06B ∪ GLH	0,76	5,0
PHOG 3L/06B ∪ GLH	0,81	5,5
PHOG 3L/15B ∪ GLH	0,82	7,2
A ∪ P ∪ GLH	0,64	0,1

Table 2. OA_s computed on the DrezzyDataset. The same model was tested, using the same configuration, with/without R_{MP} regions and with/without the modified energy term T_M.

	without R_{MP}	with R_{MP}	
Category		without T_M	with T_M
Hat	58,30	60,76	60,91
Shoe	76,66	80,00	80,24
Tie	83,59	87,45	88,21

Hat, Shoe and Tie. To make possible a comparison with other methods, we also employed the VOC2011 [19] dataset. Using the set of R_{MP}, and comparing the results obtained on the DrezzyDataset with the version that does not use this set, we obtained the results presented in table 2. It is possible to observe how the use of such set allows to obtain an increase in terms of OA_s. Obviously the R_{MP} extraction causes an increase in terms of time required to perform the segmentation and classification of a single object; this increase depends on the parameter J. The results presented in table 2 were obtained using only the MaxMin strategy, setting the number of R_{MP} regions to $J = 5$. In the same table we reported the results obtained using the modified Boykov energy function (eq. 1). The use of the term T_M leads to a slight increase in terms of OA_s, while the time required to perform the classification and segmentation of a single object remains unchanged.

Table 3 shows the time required by the final model to perform a classification and segmentation of a single image, varying the image resolution W_s, and using three different configurations of seeds extraction algorithms. Using

Table 3. Time needed to train the model showed in figure 5, and to both classify and segment a single object, varying the size of the image processed by the node $\hat{C}^n_{\mathbf{P}}$, and the seed extraction strategy. The presented values were obtained by averaging on 10 independent executions of the proposed model.

	MaxMin		3x Random(20)		5x Random(20)	
W_s	Train (sec)	Test per Obj. (sec)	Train (sec)	Test per Obj. (sec)	Train (sec)	Test per Obj. (sec)
100x100	14,41	0,48	33,16	0,64	58,46	0,92
150x150	26,16	0,88	68,35	1,13	112,56	1,78
200x200	38,93	1,55	118,43	2,19	207,70	2,96
250x250	61,10	2,18	163,30	4,43	302,10	5,78
300x300	86,99	2,82	227,09	7,08	523,04	8,67

Table 4. OA_s obtained on the dataset DrezzyDataset, using the overall strategy

Category	Test OA_s
Background	90,56
Hat	64,81
Shoe	68,13
Tie	74,09
Overall	74,02

Table 5. Comparison between the OA_s obtained on a subset of classes of the VOC2011 dataset, using the existing MNOD algorithm and the proposed model

	Multi Class	Single Class	
Category	using $\hat{C}^n_{\mathbf{P}}$ node	without $\hat{C}^n_{\mathbf{P}}$ node	using $\hat{C}^n_{\mathbf{P}}$ node
Background	83,10	//	//
Aeroplane	23,13	43,12	36,91
Bicycle	3,56	9,58	12,14
Boat	15,66	24,19	21,67
Bus	25,01	58,51	48,72
Car	20,55	33,87	33,24
Motorbike	23,80	50,51	36,02
Train	21,39	48,48	35,92
Overall	27,02	38,32	32,08

$W_s = 200 \times 200$ (typical size of the images belonging to the DrezzyDataset), the proposed model needs approximately 2.9 *sec* to perform the classification and segmentation of an image, using 5 times the Random seed extraction strategy.

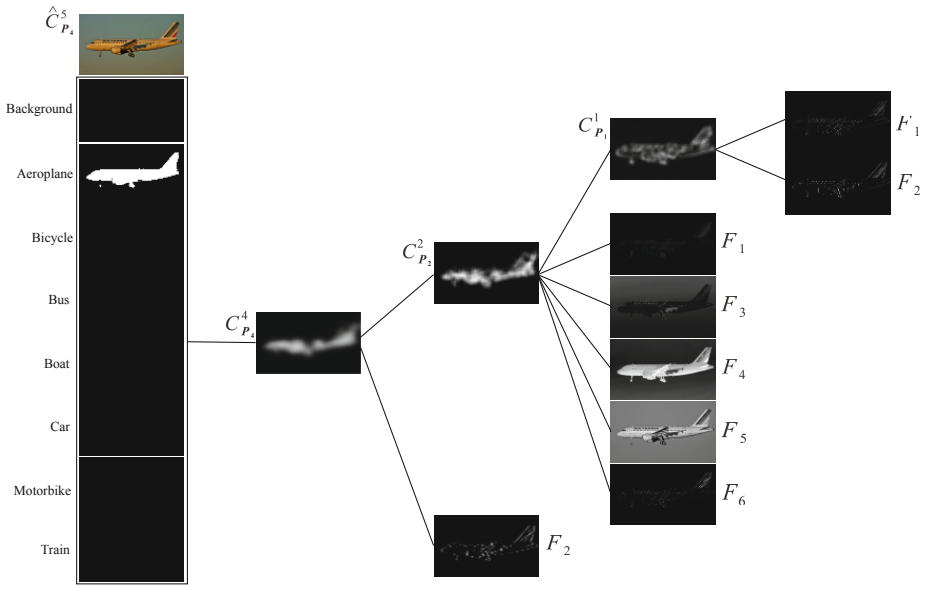

Fig. 5. Topology of a MNOD model using a node $\hat{C}^m_{\mathbf{P}}$; this configuration is used to classify and to segment the objects belonging to a subset of classes of the VOC2011 dataset

The model is therefore very efficient when compared with other algorithms presented in the literature that perform at the same time the classification and segmentation.

Finally, we evaluated the overall segmentation and classification strategy on the DrezzyDataset. These results are very encouraging and are presented in table 4; however, our results are inferior to those obtained using the existing model in [18], mainly because we need to perform both segmentation and classification of each object of interest. In table 5 we present the results obtained in the segmentation and classification of the images belonging to some classes of the Pascal VOC2011 dataset. The classification of segments with the proposed model leads to a considerable decrease of accuracy, caused mainly by the features used in the classification stage; infact, they reveal to be not significant for the objects belonging to the classes of the dataset VOC2011.

5 Conclusions

The proposed model $\hat{C}^m_{\mathbf{P}}$ can be integrated into the existing MNOD algorithm in order to realize a multiclass segmentation. The choice of using multiple potential ROIs, and the introduction of the MNOD energy term, leaded to an increase in overall accuracy in both the classification and segmentation phases if compared to the base model. A MNOD network that makes use of the node $\hat{C}^m_{\mathbf{P}}$ can be

trained in a reasonable time; setting up an optimal configuration, the evaluation of a single image takes less than three seconds. The fact that the developed features strongly depend on the application context is a major issue to take into consideration.

The use of the *min-cut* algorithm with our modified Boykov energy function permits to exploit the information from the soft segmentation masks produced by the sliding window nodes. Thus, the choice of setting the position of the seeds starting from the available detection mask led to a good solution as it allows to considerably reduce the number of segments to be generated. However, we could not define a stable and optimal seed extraction strategy, which always produces the same set of seeds starting from the same detection mask.

The proposed model inspires interesting future developments such as: development of new features, more general than the proposed ones; development of a stable and optimal seed extraction strategy; employment of more than one classification node in the MNOD network, both in cascade or in parallel.

References

1. Felzenszwalb, P.F., Mcallester, D.A., Ramanan, D.: A discriminatively trained, multiscale, deformable part model. In: Proceedings of International Conference of Computer Vision and Pattern Recognition (2008)
2. Gallo, I., Nodari, A.: Learning object detection using multiple neural netwoks. In: Proceedings of International Joint Conference on Computer Vision, Imaging and Computer Graphics Theory and Applications. INSTICC Press (2011)
3. Li, F., Carreira, J., Sminchisescu, C.: Object recognition as ranking holistic figure-ground hypotheses. In: Proceedings of International Conference on Computer Vision and Pattern Recognition, pp. 1712–1719. IEEE (2010)
4. Carreira, J., Sminchisescu, C.: Constrained parametric min-cuts for automatic object segmentation. In: Proceedings of International Conference on Computer Vision and Pattern Recognition (2010)
5. Serre, T., Wolf, L., Bileschi, S., Riesenhuber, M., Poggio, T.: Robust object recognition with cortex-like mechanisms. IEEE Transactions on Pattern Analysis and Machine Intelligence 29, 411–426 (2007)
6. Greig, D.M., Porteous, B.T., Seheult, A.H.: Exact maximum a posteriori estimation for binary images. Journal of the Royal Statistical Society, 271–279 (1989)
7. Boykov, Y.Y., Jolly, M.P.: Interactive graph cuts for optimal boundary & region segmentation of objects in n-d images. In: Proceedings of IEEE International Conference on Computer Vision, vol. 1 (2001)
8. Sharkey, A.J.: Multi-Net Systems. In: Combining Artificial Neural Nets: Ensemble and Modular Multi-Net Systems. Springer (1999)
9. Riedmiller, M., Braun, H.: A direct adaptive method for faster backpropagation learning: the RPROP algorithm. In: Proceedings of IEEE International Conference on Neural Networks, pp. 586–591 (1993)
10. Dalal, N., Triggs, B.: Histograms of oriented gradients for human detection. In: Proceedings of International Conference on Computer Vision and Pattern Recognition, pp. 886–893 (2005)
11. Boykov, Y., Kolmogorov, V.: An experimental comparison of min-cut/max-flow algorithms for energy minimization in vision. IEEE Transactions on Pattern Analysis and Machine Intelligence 26, 1124–1137 (2004)

12. Carreira, J., Sminchisescu, C.: CPMC: Automatic object segmentation using constrained parametric min-cuts. IEEE Transactions on Pattern Analysis and Machine Intelligence (2012)
13. Bosch, A., Zisserman, A., Munoz, X.: Representing shape with a spatial pyramid kernel. In: Proceedings of the 6th ACM International Conference on Image and Video Retrieval, CIVR 2007, pp. 401–408. ACM (2007)
14. Lowe, D.G.: Object recognition from local scale-invariant features. In: Proceedings of the International Conference on Computer Vision. IEEE Computer Society (1999)
15. Berg, A.C., Malik, J.: Geometric blur for template matching. In: Proceedings of International Conference on Computer Vision and Pattern Recognition, vol. 1(C), pp. 607–614 (2001)
16. Shechtman, E., Irani, M.: Matching local self-similarities across images and videos. In: Proceedings of International Conference on Computer Vision and Pattern Recognition (2007)
17. Nodari, A., Ghiringhelli, M., Zamberletti, A., Albertini, S., Vanetti, M., Gallo, I.: A mobile visual search application for content based image retrieval in the fashion domain. In: Workshop on Content-Based Multimedia Indexing (2012)
18. Albertini, S., Gallo, I., Vanetti, M., Nodari, A.: Learning object segmentation using a multi network segment classification approach. In: Proceedings of International Conference on Computer Vision Theory and Applications (2012)
19. Everingham, M., Van Gool, L., Williams, C.K.I., Winn, J., Zisserman, A.: The pascal visual object classes challenge (VOC) 2007-2011 results