# Classification of Segmented Objects through a Multi-Net Approach

A. Zamberletti, I. Gallo, S. Albertini, M. Vanetti, and A. Nodari

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

**Abstract.** We propose an algorithm for the classification of segmented objects based on a set of neural networks. The proposed model aims to extend the MNOD algorithm adding a new type of node specialized in the objects 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. The segments are classified by a suitable neural model and then the segment with higher value is chosen, in according to a proper energy function. The advantage leading by the proposed method is twofold because it allows to segment and classify the objects 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], whilst other recent works try to solve the problem 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, based on a set of neural networks, with the aim to use both the 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 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, when it is given to the following node, should lead to an improvement as if it were a refinement process and also lead to an improvement in the ability to generalize of the system.

The goal of this work is the realization of an algorithm to segment and classify all the objects of interest present in an image. The segmentation maps that are produced by the MNOD are used in order to detect several ROIs inside the processed image. They are detected following a pyramidal approach in order to maximize the probability that each object of interest was completely surrounded by a ROI boundary. The object in each ROI is segmented by the MinCut algorithm [5] with an energy function inspired by the Boykov function [6]. A set of seeds for the alleged object and background must be chosen in order to perform the segmentation with the MinCut: in this work we compared different strategies, some of that exploit the information from the MNOD segmentation masks. It is possible to obtain different segmentations for each ROI just running the MinCut algorithm with different seed extraction strategies. The segmentations obtained from each ROI are processed by a neural model which 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 the classification for each single 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

The MNOD is a Multi-Net System [7] 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. This model 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 an segmentation map as result. The process starts from the source nodes, which apply operators and filters on the input

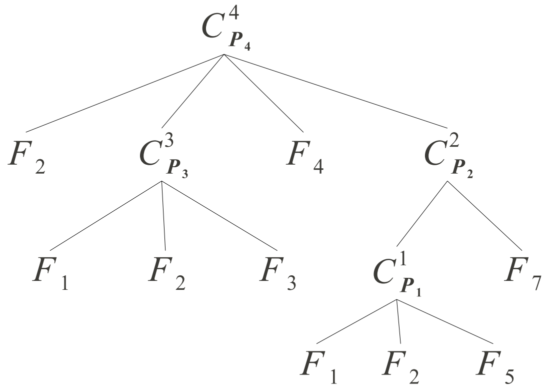

Fig. 1: Generic structure of the proposed MNOD. The nodes $C^n_{\mathbf{P}}$ represent the supervised neural models which receive their input directly from the source nodes $F_i$ and/or other internal nodes $C^m_{\mathbf{P}}$.

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 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 in 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 in a feed-forward Multi-Layer Perceptron (MLP) and is trained using the Resilient Backpropagation learning algorithm proposed by Riedmiller and Braun [8].

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 [9], 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

One characteristic of the MNOD model presented in the previous section is the possibility to change the node type for each $C^n_{\mathbf{P}}$. It is possible to implement a new node and make it works just respecting the interface's contract. In this work we used this feature just trying to integrate 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; then 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 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 is the best segmentation and its class. 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 in input a single node of type $C^n_{\mathbf{P}}$. This constraint is used to simplify the algorithm, but the same algorithm can be generalized by removing this constraint. Starting from the segmentation map $M$ that the node $\hat{C}^n_{\mathbf{P}}$ receives as input, the algorithm first identifies all the ROIs $R = \{r_1, \ldots, r_K\}$ for a given input image $I$. Each $r_k$ corresponds to a connected region identified as

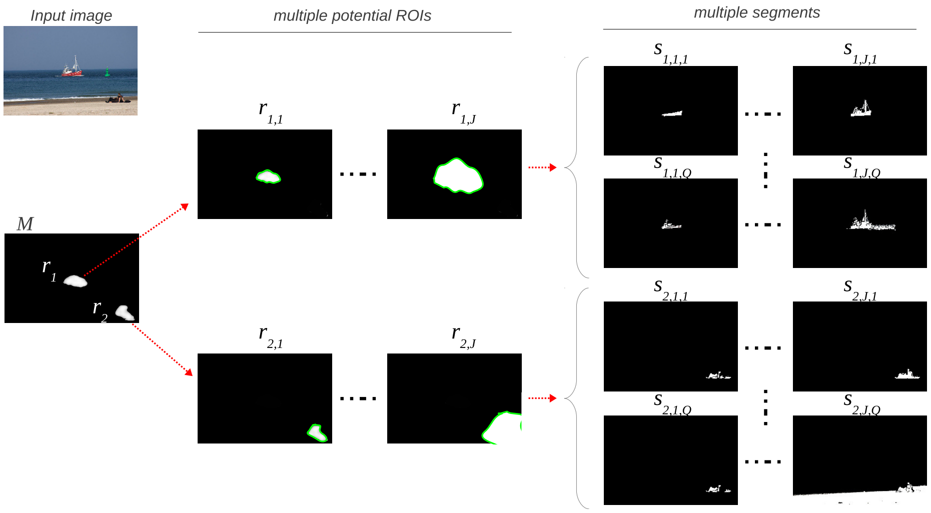

Fig. 2: An example of segments $s_{k,j,q}$ created from an input image and an output MNOD map $M$

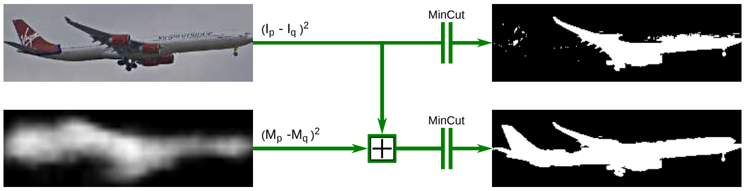

Fig. 3: Example that compares the original energy function of the *min-cut* algorithm here used, with that modified by us

in the example showed in Figure 2. If the child node has worked properly, every $r_k$ contains exactly one of all the objects of interest contained in $I$. However, the identified $r_k$ may contain only partially the object, which can lead to a not optimal segmentation of the object. To maximize the probability that a $r_k$ entirely encloses the object of interest, a set of new ROI $r_{k,j}$ were generated dilating $r_k$ $J$ times with a mathematical morphology dilation operator, obtaining a set $R_{MP} = \{r_{1,1} \ldots, r_{K,J}\}$ of Multiple Potential ROIs.

For each $r_{k,j}$ were 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 [10]. $Q$ corresponds to the number of different algorithms used to initialize the seeds of the *min-cut* algorithm, or can be considered as the number of applications of the same algorithm configured using different parameters.

We modified the energy function proposed by Boykov, 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.

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

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 boundind box of the same ROI $r_{k,j}$. In this paper we used $Q = 5$ different algorithms for the creation of all the seed (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 $|P_{obj}| = |P_{bg}|$.

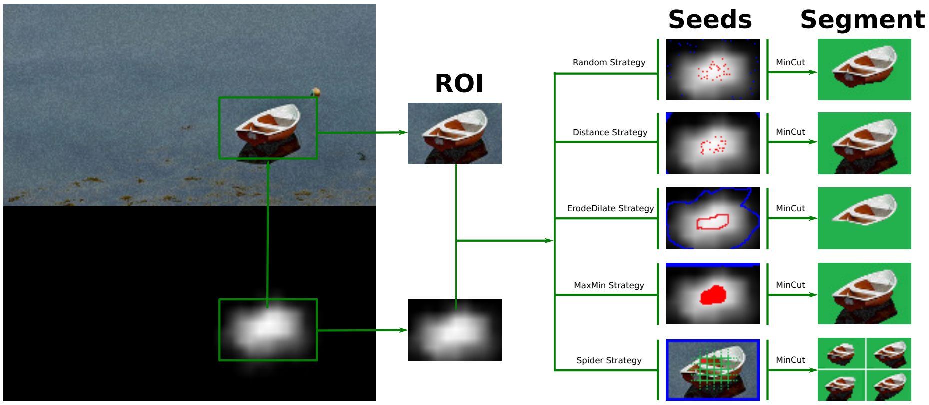

Fig. 4: Example of segments produced by the proposed min-cut algorithm, varying the algorithm that creates the set of seeds

**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.

**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 [11]. 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 best 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 [12], 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 determining the class for the ROI $r_k$, we choose the best segment $s_{k,j,q}$ $\forall j, q$. 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 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 objects 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 $\mathrm{OA}_c$, or considering the individual pixels, obtaining the Overall Segmentation Accuracy $\mathrm{OA}_s$,

All the results were obtained by running the proposed algorithm in computer having the following configuration: a single C# thread, on an Intel®Core ™i5 CPU at 2.67GHz.

In the first experiment the following 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 and 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 having a resolution equals to 200x200, and divided into 46 classes.

The objective is to determine the feature most effective for the classification of objects belonging to the *Drezzy-46* dataset. Varying the features parameters we obtained that the highest $\mathrm{OA}_c$ is computed using the PHOG feature computed on the segments $s_{k,j,q}$ and not on the entire image.

The next step attempts try to associate with the PHOW feature other simple features, in order to increase the classification accuracy. In particular, we tried to

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

Table 1: $OA_c$ computed on the dataset *Drezzy-46* by combining the input features used to create the patterns

| 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 |

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 fix the parameters of the PHOG to 3 layers and 6 bins for each histogram and to complement the PHOG feature with P, A and GL as the final configuration.

For 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 the three classes *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 obtain 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 $J$ parameter. The results obtained and showed in Table 2 were obtained using only the *MaxMin* strategy as seed extraction, fixing the number of $R_{MP}$ regions to $J = 5$. In the same table we reported the results obtained with the modified Boykov energy function (Eq. 1). The use of the term $T_M$ leads to a slight increase in terms of $OA_s$. The time required to perform the classification and segmentation of a single object remains unchanged.

In Table 3 the time required by the final model to classify and segment a single image, varying the image resolution $W_s$, was evaluated on three different

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

Table 2: OA$_s$ accuracies computed on the *DrezzyDataset*. The same configuration of the proposed model was tested with/without the use of regions $R_{MP}$ regions and with/without the 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 |

Table 3: Time needed to complete the training phase of the model showed in Figure 5 and for classification and segmentation of a single object. We varied the size of the image processed by the proposed node and the strategy for the *seed* extraction. The values were obtained by averaging on 10 independent executions of the same algorithm

| | 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 |

configurations of algorithms for the seeds extraction. Using $W_s = 200 \times 200$ (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. 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 from the proposed model in [18] mainly because classifying and segmenting simultaneously is a more difficult problem than only segment. In Table 5 we present the results obtained with 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, that are not very significant for the objects belonging to the classes of the dataset VOC2011.

Table 4: $OA_s$ obtained using the overall strategy on the dataset *DrezzyDataset*.

| Category | Test OA$_s$ |
|---|---|
| Background | 90,56 |
| Hat | 64,81 |
| Shoe | 68,13 |
| Tie | 74,09 |
| Overall | 74,02 |

Table 5: Overall segmentation accuracy obtained using a single class model compared to the accuracy obtained with the multiclass model on the *VOC2011* dataset.

|  | 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 |

## 5 Conclusions

The proposed node $\hat{C}^n_{\mathbf{P}}$ can be integrated into the existing algorithm MNOD to make possible a multiclass segmentation. The choice of using multiple potential ROIs and the introduction of the MNOD energy term increased the accuracy obtained both in the classification and segmentation processes of the ROIs campared to the base model. A MNOD network which uses the new node 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 choice of using the *min-cut* algorithm with the modified Boykov energy function is a good choice because, using the proposed seed extraction strategies, it is possible to full 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.

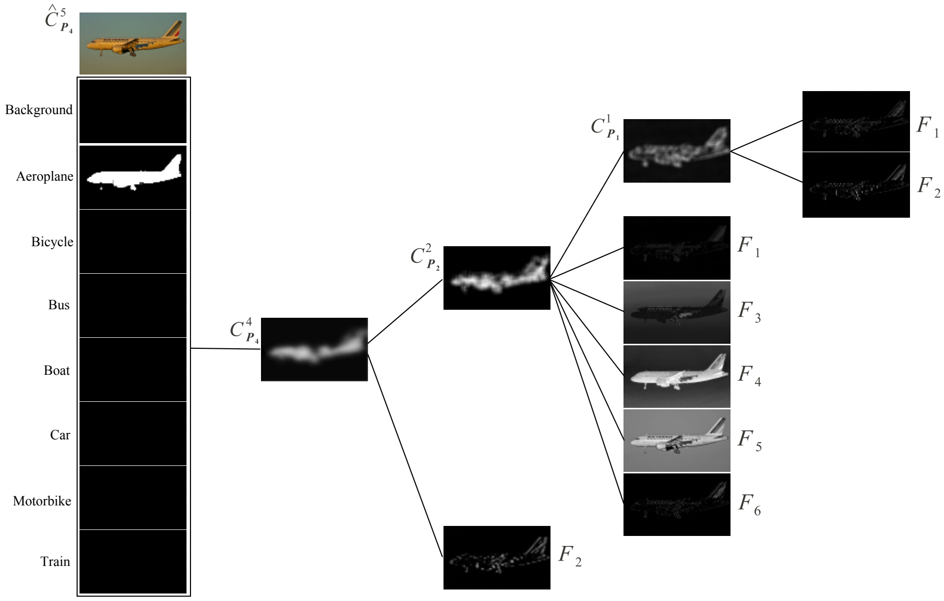

Fig. 5: Topology of the MNOD with the proposed node $\hat{C}^n_{\mathbf{P}}$. This configuration is used to classify and to segment the objects belonging to a subset of classes of the VOC2011 dataset

However, we could not define a stable and optimal seed extraction strategy, that is a 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, both in cascade or in parallel in the MNOD network.

## References

1. Felzenszwalb, P.F., Mcallester, D.A., Ramanan, D.: A Discriminatively Trained, Multiscale, Deformable Part Model. In: Computer Vision and Pattern Recognition. (2008) 1–8
2. Gallo, I., Nodari, A.: Learning object detection using multiple neural netwoks. In: VISAPP 2011, INSTICC Press (2011)
3. Li, F., Carreira, J., Sminchisescu, C.: Object recognition as ranking holistic figure-ground hypotheses. In: CVPR, IEEE (2010) 1712–1719
4. Carreira, J., Sminchisescu, C.: Constrained Parametric Min-Cuts for Automatic Object Segmentation. In: IEEE International Conference on Computer Vision and Pattern Recognition. (June 2010) description of our winning PASCAL VOC 2009 and 2010 segmentation entry.

5. Greig, D.M., Porteous, B.T., Seheult, A.H.: Exact maximum a posteriori estimation for binary images. Journal of the Royal Statistical Society, (1989) 271–279
6. Boykov, Y.Y., Jolly, M.P.: Interactive graph cuts for optimal boundary & region segmentation of objects in n-d images. In: Computer Vision, 2001. ICCV 2001. Proceedings. Eighth IEEE International Conference on. Volume 1. (2001) 105–112
7. Sharkey, A.J.: Multi-Net Systems. In: Combining Artificial Neural Nets: Ensemble and Modular Multi-Net Systems. Springer (1999)
8. Riedmiller, M., Braun, H.: A direct adaptive method for faster backpropagation learning: The rprop algorithm. In: IEEE International Conference On Neural Networks. (1993) 586–591
9. Dalal, N., Triggs, B.: Histograms of oriented gradients for human detection. In: Proc. CVPR. (2005) 886–893
10. Boykov, Y., Kolmogorov, V.: An experimental comparison of min-cut/max-flow algorithms for energy minimization in vision. IEEE Trans. Pattern Anal. Mach. Intell. **26** (September 2004) 1124–1137
11. Carreira, J., Sminchisescu, C.: Cpmc: Automatic object segmentation using constrained parametric min-cuts. IEEE Transactions on Pattern Analysis and Machine Intelligence (2012)
12. 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 '07, ACM (2007) 401–408
13. Bosch, A., Zisserman, A., Muñoz, X.: Representing shape with a spatial pyramid kernel. In: CIVR. (2007) 401–408
14. Lowe, D.G.: Object recognition from local scale-invariant features. In: Proceedings of the International Conference on Computer Vision-Volume 2 - Volume 2. ICCV '99, Washington, DC, USA, IEEE Computer Society (1999) 1150–
15. Berg, A.C., Malik, J.: Geometric blur for template matching. Proceedings of the 2001 IEEE Computer Society Conference on Computer Vision and Pattern Recognition CVPR 2001 **1**(C) (2001) I–607–I–614
16. Shechtman, E., Irani, M.: Matching local self-similarities across images and videos. In: IEEE Conference on Computer Vision and Pattern Recognition 2007 (CVPR'07). (June 2007)
17. Nodari, A., Ghiringhelli, M., 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 (CBMI2012). (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 (VISAPP). (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. http://www.pascal-network.org/challenges/VOC