Pattern Recognition 108 (2020) 107557

Contents lists available at ScienceDirect

Pattern Recognition

journal homepage: www.elsevier.com/locate/patcog

SEMEDA: Enhancing segmentation precision with semantic edge aware loss

Yifu Chen*, Arnaud Dapogny, Matthieu Cord

Sorbonne Université, UMR 7606, LIP6, 4 place Jussieu, Paris, France

ARTICLE INFO

Article history:
Received 14 January 2020
Revised 21 May 2020
Accepted 20 July 2020
Available online 31 July 2020

Keywords:
Semantic segmentation
Loss function
Computer vision

ABSTRACT

Per-Pixel Cross entropy (PPCE) is a commonly used loss on semantic segmentation tasks. However, it suffers from a number of drawbacks. Firstly, PPCE only depends on the probability of the ground truth class since the latter is usually one-hot encoded. Secondly, PPCE treats all pixels independently and does not take the local structure into account. While perceptual losses (e.g. matching prediction and ground truth in the embedding space of a pre-trained VGG network) would theoretically address these concerns, it does not constitute a practical solution as segmentation masks follow a distribution that differs largely from natural images. In this paper, we introduce a SEMantic EDge-Aware strategy (SEMEDA) to solve these issues. Inspired by perceptual losses, we propose to match the 'probability texture' of predicted segmentation mask and ground truth through a proxy network trained for semantic edge detection on the ground truth masks. Through thorough experimental validation on several datasets, we show that SEMEDA steadily improves the segmentation accuracy with negligible computational overhead and can be added with any popular segmentation networks in an end-to-end training framework.

1. Introduction

Semantic segmentation is one of the fundamental domains of computer vision, which aims at assigning a semantic label to each pixel of an image. Current state-of-the-art methods mainly rely on fully convolutional neural network architectures [1], that are trained by optimizing a per-pixel loss between predictions and ground truth labels. Fully convolutional neural networks cleverly inherit the idea of classifying each pixel by using a patch centered on it. In this manner, popular classification networks can be adapted into a fully convolutional fashion and their learned representations can be transferred to segmentation tasks by training them with per-pixel cross-entropy (PPCE) loss. This loss is a natural choice as a spatial extension of cross-entropy loss, which is the gold standard for classification. However, in this paper, we argue that PPCE loss has some drawbacks in image semantic segmentation context.

Firstly, PPCE loss is just an average over each pixel's accuracy and can not capture structural differences between output and ground-truth segmentation masks (**problem a**). Each pixel is treated equally and independently in PPCE loss. For example, a good segmentation mask should preserve the semantic boundary of each object. However, PPCE loss is not capable to measure this similarity. As a result, models trained with PPCE loss usually struggle to output geometrically correct predictions. A similar issue has been found in image transformation problems, where an input image is transformed into an output image. A per-pixel loss such loss L1 and loss L2 is usually used to in these problems [2–4]. As criticized by [5], per-pixel losses can not measure perceptual differences between predicted and ground-truth images. An interesting solution provided by [5] is called "perceptual loss", where generated and ground truth images are matched in the embedding spaces of the layers of a pre-trained classification network (usually VGG network). Traditionally, more emphasis is put on the weights corresponding to the first layers: to a certain extent, perceptual losses lead to match higher-order moments (e.g. gradients) extracted by these layers, thus taking into account the neighborhood of each pixel. However, the perceptual loss cannot directly be used for semantic segmentation. Firstly, the perceptual loss is applied to RGB images while segmentation mask has C channels where C is to the total number of classes. Secondly, the distribution of segmentation masks is wildly different from natural images.

Secondly, PPCE only depends on the predicted value of the ground truth class but not the entire distribution over all classes. (**problem b**). For instance, consider a case where there are three classes (cat, dog and horse) and a pixel that belongs to the region of a cat. Prediction (0.5, 0.25, 0.25) and (0.5, 0.45, 0.05) will have the same loss value since the cross entropy loss only depends on

* Corresponding author.
E-mail addresses: yifu.chen@lip6.fr (Y. Chen), matthieu.cord@lip6.fr (M. Cord).

https://doi.org/10.1016/j.patcog.2020.107557
0031-3203/

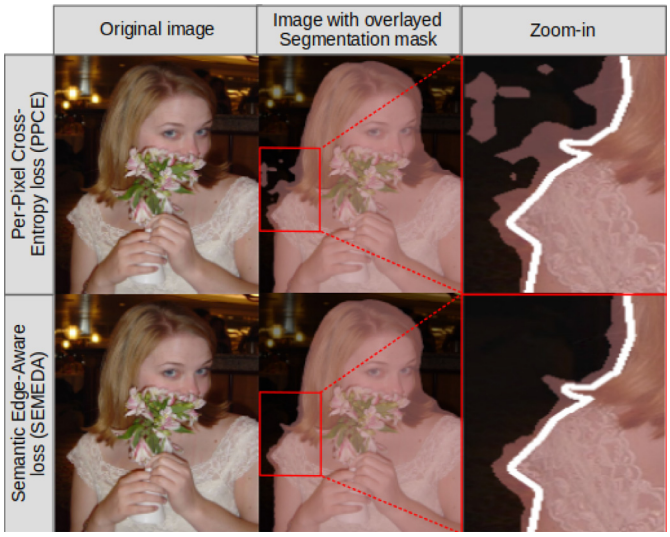

Fig. 1. Segmentation results: contrarily to traditional losses for semantic segmentation, the proposed SEMEDA approach enforces structure on the output masks. Ground truth edges are represented by white lines in the last column. The proposed SEMEDA framework allows a significantly better fit to the object boundaries and structure. This visualization of our method has been done with a Deeplab-v2 backbone.

the value 0.5 (class cat). However, these two predictions are not equivalent. If the pixel is located in the center of the cat region, we may prefer the first prediction since it is more robust. If the pixel is located at the border where a cat and a dog overlap, the second prediction may be more reasonable.

Problem a-b of the traditional PPCE loss are illustrated on Fig. 2. To address these problems, we propose in this paper a novel SEMantic EDge-Aware (SEMEDA) loss to obtain better shaped segmentation results (Fig. 1). Classically, a semantic segmentation network (displayed in red in Fig. 3) is trained to output a segmentation mask from an image. In SEMEDA, we use the embedding space of another network that we refer to as the SEMEDA network (blue in Fig. 3) to match the predicted segmentation mask with the corresponding ground truth. The SEMEDA network is pre-trained to detect semantic edges from the segmentation masks. The contributions of this paper are three-fold:

- We analyze the drawbacks of the standard PPCE loss in semantic segmentation problem and propose a novel SEMEDA loss that fixes these issues.
- We design a Semantic Edge Detection Network to capture local spatial structure in the segmentation mask space.
- We perform thorough experiments on VOC 2012, Cityscapes, and HELEN datasets and show that SEMEDA leads to substantial improvements in all tested configurations.

The rest of this paper is organized as follows. First, section 2 presents the state of the art methods in semantic segmentation and the relationship with our method. Then, section 3 defines the methodology of the proposed method. Thorough experiments including quantitative and qualitative analysis are reported in section 4, and discussion and conclusion of these results are addressed in section 5.

2. Related work

A traditional supervised machine learning algorithm contains three parts: a hypothesis space, an objective function, and an optimization method. In this point of view, the standard pipeline of nowadays methods for semantic segmentation is to train a well-designed convolution neural network with per-pixel cross entropy loss by using an optimization algorithm such as stochastic gradient descent through back-propagation. Fully Convolutional Network (FCN) [1] is one of the first works who proposed this pipeline and it becomes a benchmark for later research. Methods based on FCN have made great progress in image semantic segmentation. These methods are generally composed of three parts: a backbone network, segmentation blocks and a classifier. The backbone network is transferred from a pre-trained classification network and produces semantic features from an input image. Segmentation blocks are designed to leverage geometry information to refine the feature obtained by the backbone network. Finally, prediction is produced by a simple linear classifier (1*1 convolution). FCN [1] produces coarse segmentation masks and achieves quite good results on segmentation datasets. Recent works are almost all based on FCN and are working on refining segmentation masks. Unlike traditional segmentation methods which require specific optimization methods [6,7], learning methods based on stochastic gradient descent work well for most types of neural networks, including segmentation networks. Thus, the research of segmentation methods based on deep learning mainly focuses on models and loss function design. The objective is to build models and loss functions that are more suitable for segmentation tasks.

In order to build segmentation-specific networks, there are two key components: spatial information and context information. Since segmentation tasks require pixel-level classification, models must use spatial information (no semantic) to determine the class of each pixel, and then combine with contextual information (semantics) to enhance pixel classification. One of the most efficient way to exploit spatial information is "Atrous" convolution introduced by Chen et al. [8], which enlarges the receptive field without increasing the number of model parameters. It can

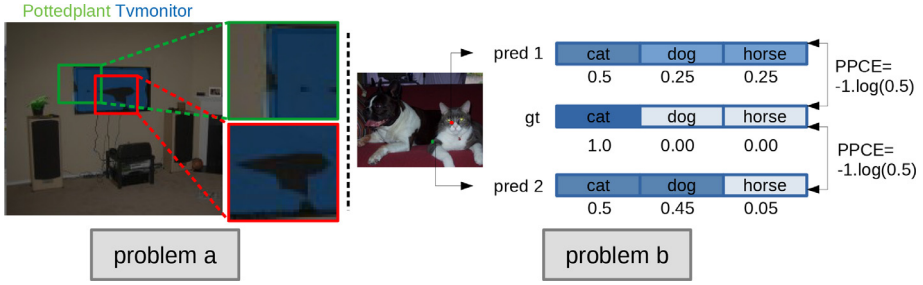

Fig. 2. Illustration of **problem a-b** of traditional PPCE loss for semantic segmentation. **problem a:** the segmentation maps predicted by a network trained with PPCE exhibit a lack of structure (upper box: loose semantic boundaries, lower box: holes in the predicted semantic regions). **problem b:** Two different predictions (*pred1* and *pred2*) that are equivalent for cross entropy loss. However, we may prefer *pred2* for boundary pixels (lower point) and *pred1* for center pixels (upper point).

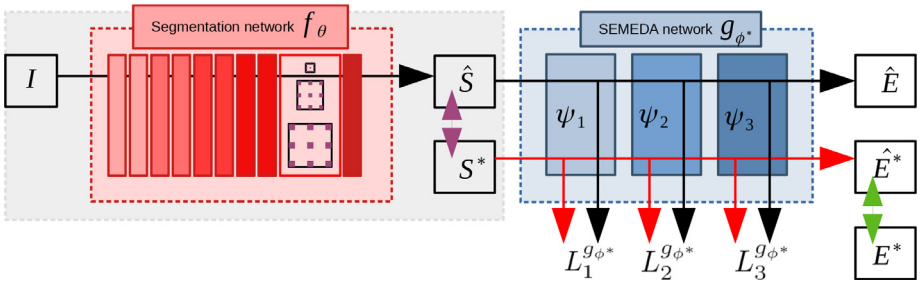

Fig. 3. Overview of our SEMEDA framework for semantic segmentation. A segmentation network f_θ outputs a predicted mask $\hat{S}$. f_θ can be any backbone network, e.g. in our case, a Deeplab v2 or Deeplab v3+ architecture. Both this predicted mask, as well as the corresponding ground truth mask are provided independently to a SEMEDA network g_ϕ that is trained for semantic edge detection (arrow between $\hat{E}^*$ and E^*, Eq. 7). We complete the traditional PPCE loss (arrow between $\hat{S}$ and S^*) by adding a novel SEMEDA loss term (Eq. 4), which consists in matching the predicted segmentation mask with its corresponding ground truth within the embeddings of the SEMEDA network, each layer l gives rise to a contribution $L_l^{g_{\phi^*}}$ (Eq. 5).

be used in the backbone to maintain more spatial information and also in segmentation blocks to capture information of different ranges [9]. The encoder-decoder architecture is another popular strategy employed by many modern methods. The idea is that spatial information is encoded in hidden layers of the encoder, and can be recovered by progressively reusing these features in the segmentation blocks [1,10,11]. use deconvolution [12] to learn the upsampling while [13] reuses the pooling indices from the encoder and learns extra convolutional layers. U-Net [14] adds skip connections from the encoder features to the corresponding decoder activations and SeENet [15] concatenates enhanced the low level features with deep layers features. A particularly useful type of spatial information is edge information, which is believed to be encoded in low level features. ET-Net [16] further proposes to use these features for edge detection, and the resulting edge maps are then incorporated into the segmentation prediction. Contextual information has also proven to be effective for segmentation. The prediction of one pixel could be corrected or enhanced by the semantic meaning of context pixels. A simple and effective method is to aggregate contextual information at a fixed scale, such as ASPP module [9] and pyramid average pooling [17]. Recently, attention mechanisms [18–21] have been introduced to estimate pixel-aware context before aggregating. Although state-of-the-art methods incorporate both spatial and contextual information, the proposed architectures are generally trained with PPCE.

Another important direction of research is to define more relevant loss functions for semantic segmentation. There are three main criticisms of the commonly used PPCE loss. First, PPCE treats each pixel equally and, as such, may suffer from data imbalanced problem. To address this problem, Long *et al.* [1] proposes an individual weighting of the classes. Focal loss [22] tackles this problem by putting more emphasis on pixels with difficult configurations. The second issue is that the evaluation metric used in tests is not cross entropy loss but the Jaccard index (usually called mean intersection-over-union (mIoU) score). Training models with a loss function that differ from the evaluation metric could lead to undesired results. However, the limitation for using IoU directly as a loss function in semantic segmentation lies from the fact that it is non-differentiable. Therefore, some approaches employed surrogate Jaccard index losses [23,24] or generalized dice loss [25] to slightly improve the mean IoU test score. Nevertheless, neither PPCE nor Jaccard index can accurately describe the structure quality of the segmentation mask as argued in [26,27]. To better measure structural similarities between predicted and ground truth segmentation masks, some metrics such as E-measure [28], S-measure [29] and weighted F-measure [30] have been proposed to replace mIoU for evaluations. However, none of them are used as loss functions [31]. proposes to add an adversarial loss to capture implicit structure similarities. However, the shape of the semantic segments is too vague. A semantic segment does not separate two objects of the same class if they are overlapping. Moreover, considering the limited dataset size for semantic segmentation with respect to the intrinsic data variability (due to e.g. partial occlusion, object scaling or camera viewpoint change), the possible shape of a semantic segment could be extremely complicated and difficult to learn with a discriminator network. Another idea is to leverage edge information in loss functions. The most direct way is to use a weighted per-pixel loss where edge pixels are emphasized more than other pixels [32]. proposed to group pixels according to the distance from the pixel to the closest semantic boundary and each group of pixels has a different weight. Essentially, these methods assume that the closer to the boundary, the more important. However, this assumption is not always true and may provide discontinuities in the center of large objects since these pixels will have little impact on total loss under this assumption.

3. Method overview

Fig. 3 illustrates the proposed SEMEDA method to train a deep network for semantic segmentation problems. First, we introduce how traditional approaches for semantic segmentation can be formulated, and what are common pitfalls of such approaches. *A contrario*, our approach consists of matching a predicted segmentation mask with its corresponding ground truth in the embedding spaces of a pre-trained SEMEDA network. This network is trained to predict semantic edges from the ground truth segmentation masks. It is worth noting that our network is not detecting semantic edges from RGB images as it was done in [33]. SEMEDA network is trained to predict semantic edges from the ground truth segmentation masks. The features of SEMEDA network thus encode information of segmentation masks, and then are used to define SEMEDA loss function.

3.1. The pitfalls of naive segmentation approaches

A standard approach for training semantic segmentation networks is illustrated in Fig. 3 (gray box). Let's consider a semantic segmentation network f_θ parameterized by weights θ, mapping a RGB image $I \in \mathbb{R}^{H\times W\times 3}$ from a training dataset $\{I, S^*\}$ into a segmentation mask $\hat{S}$:

$$\hat{S} = f_\theta(I) \in [0,1]^{H\times W\times C} \tag{1}$$

where H (resp. W) is the height (resp. width) of the input image and C is the total number of semantic classes (including background). Such network is usually trained upon optimization of a Per-Pixel Cross Entropy (PPCE) loss $\mathcal{L}^{PPCE}$ (purple arrow in Fig. 3).

Fig. 4. Left to right: original image, ground truth segmentation and two boundary/non-boundary trimaps: one with 1 pixel width and the other with 10 pixels width. The red areas represent the boundary regions while black areas represent non-boundary regions. White areas depict 'void' pixels. (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

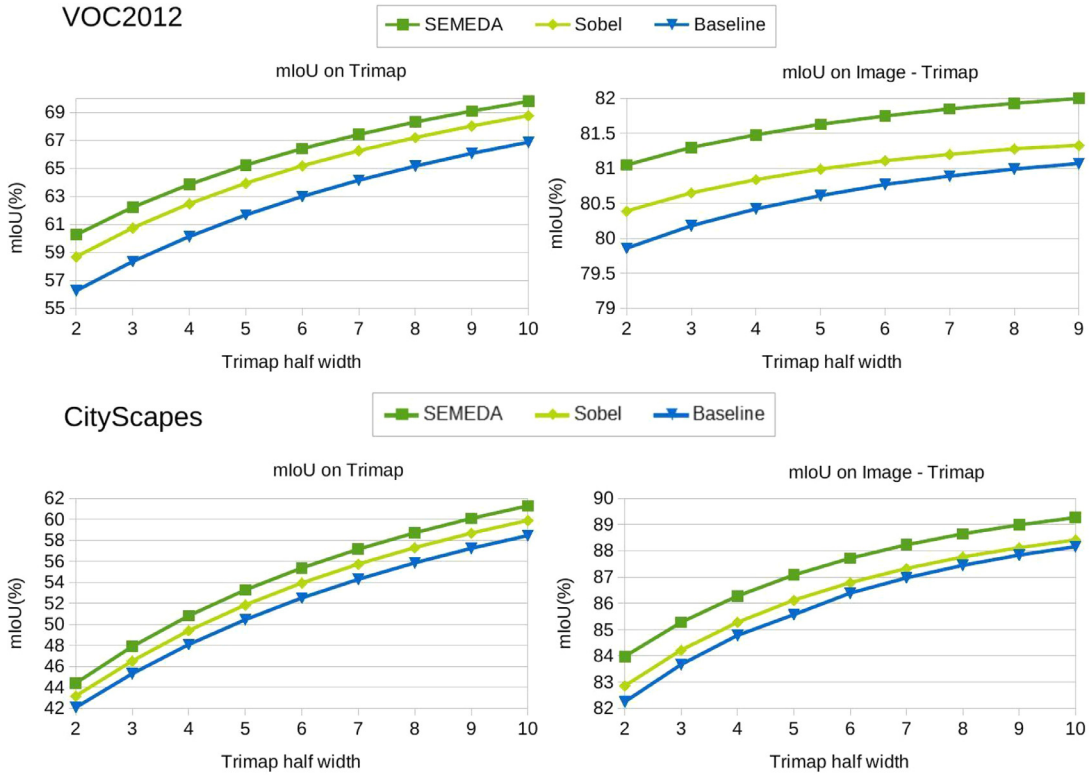

Fig. 5. mIoU for different models on boundary and non-boundary regions on VOC 2012 and Cityscapes datasets. SEMEDA consistently enhances the performance on both regions for all bandwidth.

PPCE measures the difference between the predicted label mask $\hat{S}$ and the ground truth mask S^*:

$$\mathcal{L}^{PPCE}(\hat{S}, S^*) = -\sum_{i=1}^{H}\sum_{j=1}^{W}\sum_{c=1}^{C} S_{i,j}^{c*}\log(\hat{S}_{i,j}^{c}) \tag{2}$$

We argue that this loss function exhibits several drawbacks for semantic segmentation tasks. Firstly, PPCE treats all pixels independently and does not take the local structure into account. As a result, the predicted segmentation masks usually contain holes in the structure of the segmented objects or inconsistencies at their boundaries (**problem a**). Secondly, for each pixel, since $\{S_{i,j}^{c*}\}_{c\in C}$ are usually one-hot encoded, PPCE only depends on the probability of the ground truth class. Therefore, several predictions are given the same penalty while the entropy of the predicted mask $\{\hat{S}_{i,j}^{c}\}_{c\in C}$ may vary a lot for that pixel (**problem b**).

3.2. Structure learning through edge embeddings

In the frame of style transfer and image synthesis, the authors of [5] obtained impressive results by defining high-level perceptual losses that involve a fixed pre-trained network, such as ImageNet pre-trained VGG-19 [34] network. The idea of this method is to measure the semantic difference between two images as the difference of their feature representations as computed by the fixed network. However, this method cannot be straightforwardly translated to semantic segmentation, as natural images and segmentation masks have different numbers of channels and belong to wildly different distributions. However, let's suppose that we have access to another pre-trained network g_{ϕ^*} (that we refer to as the SEMEDA network) that maps any segmentation mask $S \in [0, 1]^{H\times W\times C}$ into a binary semantic edge map $\hat{E}$:

$$\hat{E} = g_{\phi^*}(S) \in [0,1]^{H\times W\times 2} \tag{3}$$

In what follows, we respectively note $\hat{\psi}_1, \hat{\psi}_2, \ldots, \hat{\psi}_L$ the embeddings of the L layers of $g_{\phi^*}(\hat{S})$, and $\psi_1^*, \psi_2^*, \ldots, \psi_L^*$ the embeddings of the L layers of $g_{\phi^*}(S^*)$. Similarly to [5], we can thus match $\hat{S}$ and S^* within the embeddings of g_{ϕ^*} (black and red arrows in Fig. 3). We thus define our SEMantic EDge-Aware (SEMEDA) loss as follows:

$$\mathcal{L}_{\text{SEMEDA}}^{g_{\phi^*}}(\hat{S}, S^*) = \sum_{l=1}^{L}\lambda_l \mathcal{L}_l^{g_{\phi^*}}(\hat{S}, S^*) \tag{4}$$

where for all layers l, the contribution of this layer of the SEMEDA network to the total loss is:

$$\mathcal{L}_l^{g_{\phi^*}}(\hat{S}, S^*) = ||\hat{\psi}_l - \psi_l^*||_1 \tag{5}$$

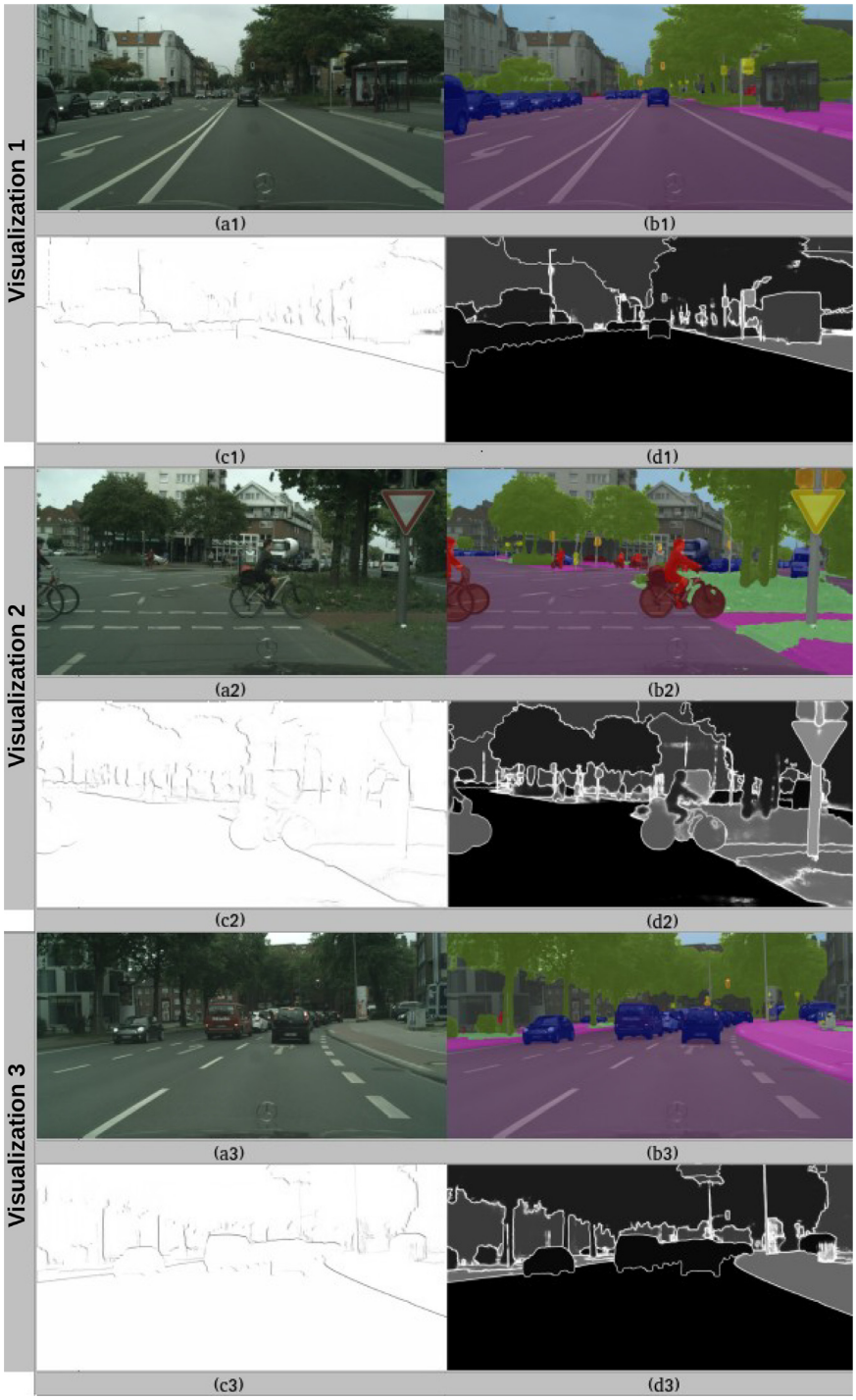

Fig. 6. Example of embeddings of the SEMEDA network: (a1,a2,a3) original images, (b1,b2,b3) predicted segmentation masks, (c1,c2,c3) and (d1,d2,d3) two feature maps of the SEMEDA network.

Where λ_l is a hyperparameter representing the importance of this layer in the total loss. The final loss function for the segmentation network is a combination of the PPCE loss on the segmentation masks and the proposed SEMEDA loss:

$$\mathcal{L}_{tot}^{g_{\phi^*}}(\hat{S}, S^*) = \mathcal{L}^{PPCE}(\hat{S}, S^*) + \mathcal{L}_{\text{SEMEDA}}^{g_{\phi^*}}(\hat{S}, S^*) \tag{6}$$

Since g_{ϕ^*} is trained to detect inter-class boundaries, minimizing $\mathcal{L}_{\text{SEMEDA}}$ naturally penalizes high-entropy configurations where the contributions of several classes are important (**problem b**). Furthermore, it also heavily penalizes discontinuities in the structure of objects (**problem a**). Thus, semantic edge detection from segmentation masks is a particularly interesting candidate objective for g_{ϕ^*}. In what follows, we describe how it can be formalized.

3.3. Learning to detect semantic edges

Given an image I and its corresponding ground truth segmentation mask S^*, we generate a binary ground truth semantic edge map E^* by setting all pixels that do not have 8 identically-labeled neighbor pixels as 1, and other pixels as 0. These ground truth semantic edge maps are calculated beforehand and no further computation is needed afterward.

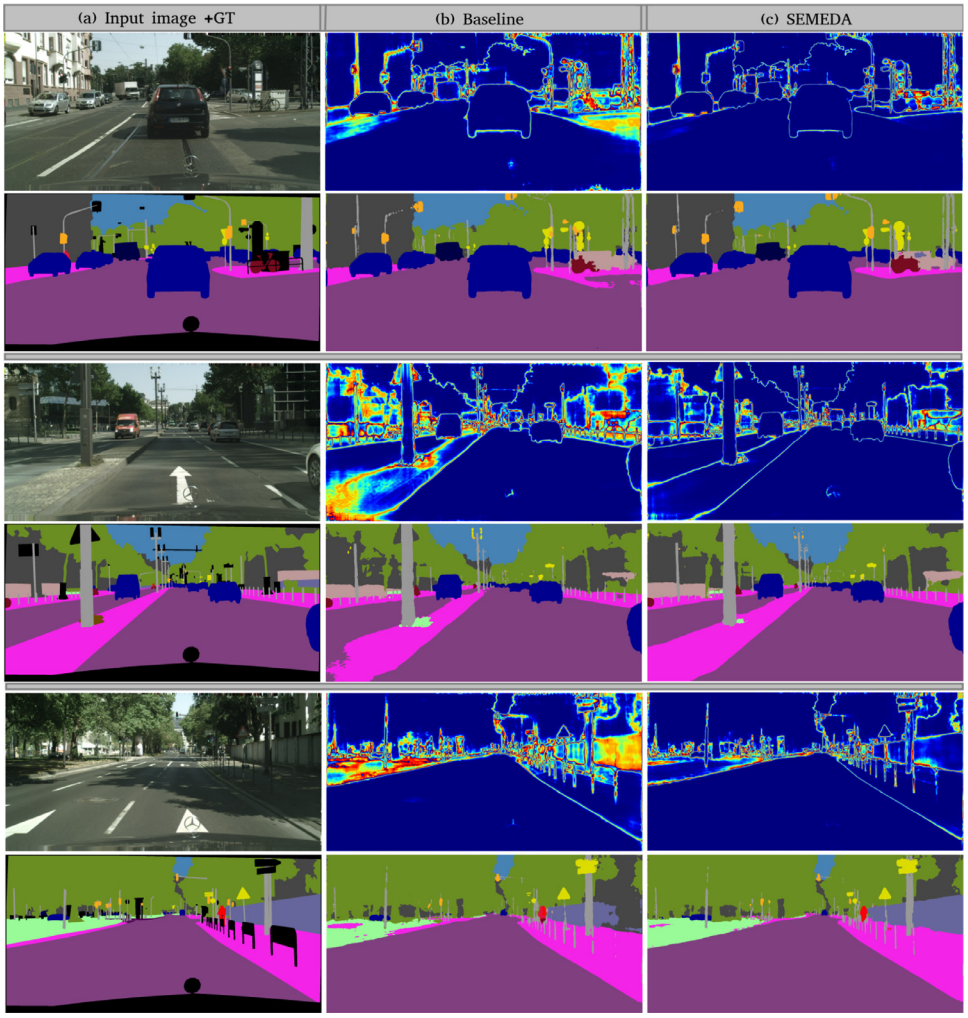

Fig. 7. Column (a) shows an input image and the corresponding semantic segmentation ground-truth. Column (b) and (c) show segmentation results (bottom) along with prediction entropy maps produced by different approaches (top). Baseline model produces noisy segmentation predictions as well as high entropy activations. Our models, on the other hand, manage to produce correct predictions at high level of confidence. (Red signifies a high entropy value.).

We train the SEMEDA network g_ϕ to minimize PPCE loss between semantic edge maps $\hat{E}^* = g_\phi(S^*)$ predicted upon the ground truth segmentation masks S^*, and ground truth generated edge maps E^*, as indicated by the green arrow in Fig. 3:

$$\mathcal{L}^{PPCE}(\hat{E}^*, E^*) = -\sum_{i=1}^{H}\sum_{j=1}^{W}\sum_{c=1}^{C}\hat{E}^{c*}_{i,j}\log(\hat{E}^{c}_{i,j}) \tag{7}$$

This network is depicted in Fig. 3 (in blue). Once the SEMEDA network is trained with parameters ϕ^*, we train the segmentation network f_θ by minimizing loss $\mathcal{L}^{g_{\phi^*}}_{\text{SEMEDA}}$. The steps for training a segmentation network with SEMEDA are summarized in Algorithm 1.

3.4. Implementation details

Segmentation network:

SEMEDA is agnostic to the architecture of the segmentation network f_θ, which can be any popular architecture: in what follows, we experiment with Deeplab-v2 [9] as well as the recent Deeplab V3+ [35]. Both architectures are composed of a backbone feature extraction network and differ by the refinement portion of the network. For that matter, we also experiment with either ImageNet pre-trained ResNet-101 and Xception-71 backbone networks.

SEMEDA network:

Because the task of detecting semantic edges from the segmentation mask is rather straightforward, we use a simple CNN composed of $L = 3$ layers with $16 \rightarrow 32 \rightarrow 2$ channels and ReLU activation (except for the last layer, which has a softmax activation) as g_ϕ.

In order to keep the runtime and memory footprint reasonable, we train our models by feeding the SEMEDA networks with 321×321 random crops for both datasets without multi-scale inputs. We augment the data with random scaling and random mirroring. Since we use mini-batches of 6 images, we fix parameters in batch norm layers and we set the initial learning rate to $5 \cdot 10^{-4}$. As is classically done in the literature, we report the mean intersection over union (mIoU) metric over all the classes as our evaluation metric.

4. Experiments

In this section, we present our experiments to validate SEMEDA. First, we perform an ablation study and discuss hyperparameter settings. Then, we validate our approach on two semantic segmentation datasets and a face parsing dataset, both quantitatively and qualitatively.

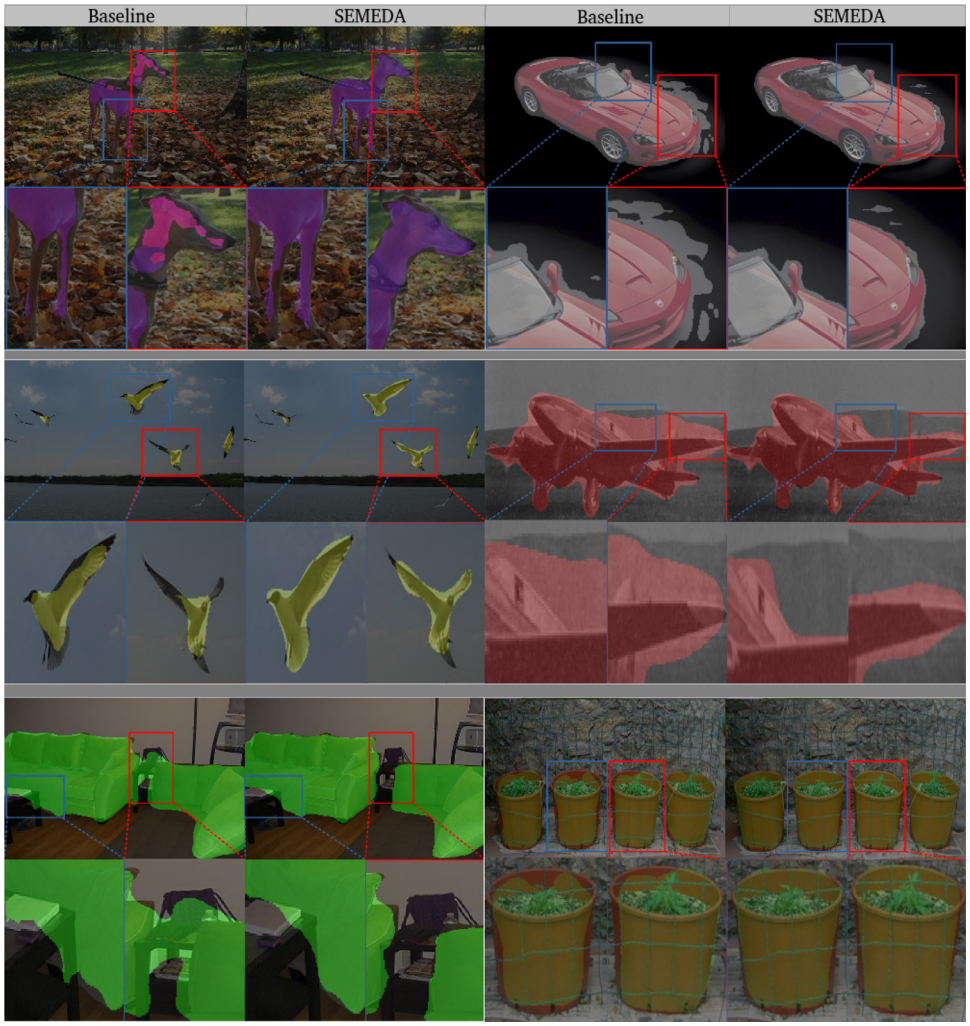

Fig. 8. Examples of predicted segmentation masks on VOC 2012 dataset, and comparison between a baseline Deeplab v3+ model trained with PPCE and SEMEDA. SEMEDA produces better shaped predictions compared with the baseline.

4.1. Experimental setup

We conduct our experiments on the **Pascal VOC 2012** dataset [36], which contains 20 foreground object classes as well as one background class. We train our models on the augmented version of the dataset [37] containing 10,582 training images and report the mean Intersection over Union (mIoU) score on the validation set which contains 1449 images.

The **Cityscapes** dataset contains 2975 images annotated with pixel-wise annotations of 18 object classes and one background class for training. We report mIoU scores on the validation set, which contains 500 images. For memory reasons, we downsample images to half resolution, i.e. 1024×512 for Deeplab v2 based experiments. We keep them in original size for experiments with Deeplab v3+ in order to achieve better results.

HELEN is a face parsing dataset containing 2000 images for train and 330 images for test set. Each image is annotated with 11 labels, such as background, hair, skin, brows, eyes, nose, "upper/lower lip", and mouth sub-regions. We report the F_1-score for each face part which is commonly used in existing face parsing literature.

We validate SEMEDA by showing (a) that introducing a structural term *via* semantic edge detection allows to better capture the underlying structure of the objects, and (b) that the SEMEDA network encodes richer embeddings compared to a naive approach (e.g. using Sobel kernels).Most edge-based segmentation methods consist in performing edge detection as an auxiliary task and then incorporating the edge features into the segmentation prediction. Our method involves using an additional loss function which can better capture structural information, including edge information, from the segmentation masks. Thus, our method is orthogonal to the edge-based methods mentioned above, and can be used in conjunction with them. The most naive approach to come close to our idea is to use Sobel kernels to define a loss function. To this end, we compare our model with two baselines:

- **PPCE:** a segmentation network trained with a traditional per-pixel cross-entropy loss.
- **Sobel:** a setup similar to SEMEDA, except the SEMEDA network is replaced by Sobel kernels that independently process each class in the segmentation mask.

In order to better analyze the improvements obtained by SEMEDA, we do not use tricks such as adding multi-scale and flipped images during inference. In what follows, we show that SEMEDA significantly enhances the segmentation accuracy regardless of the architecture of the segmentation network, the underlying backbone, the pre-training strategy and on multiple datasets.

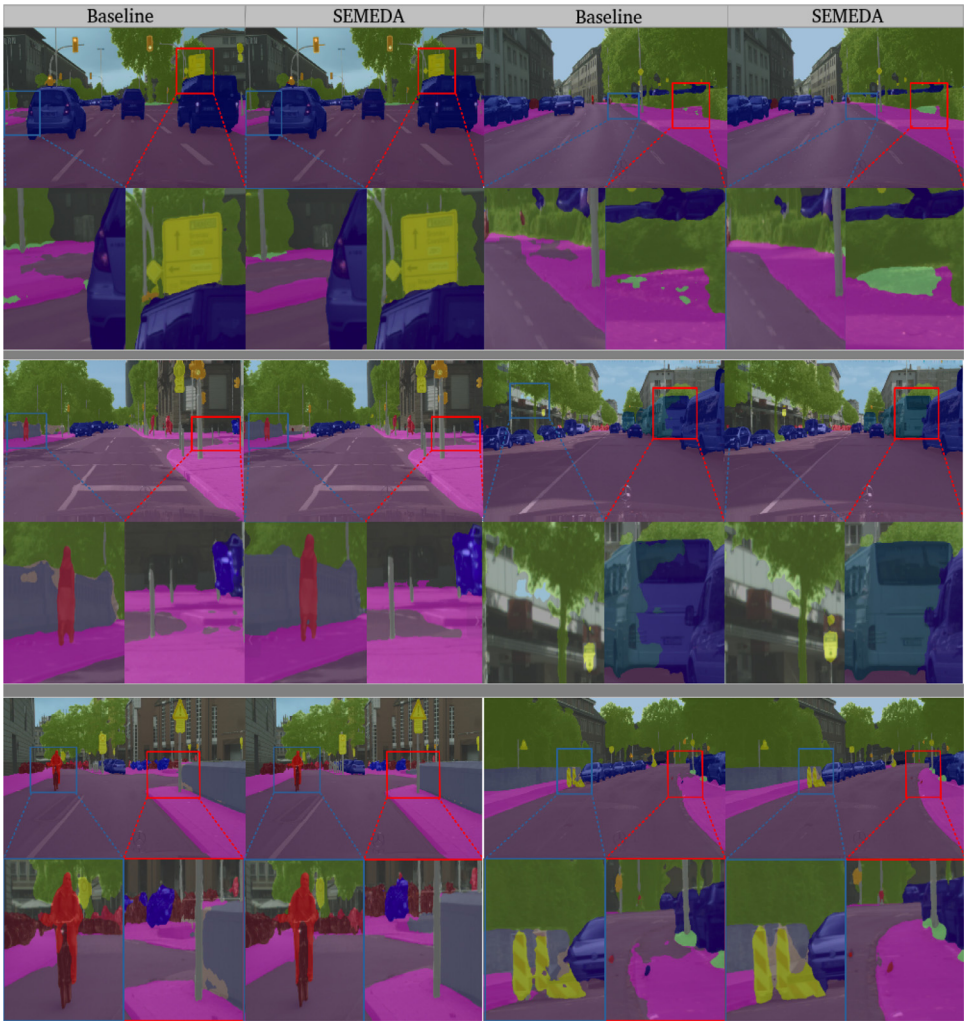

Fig. 9. Examples of predicted segmentation masks on Cityscapes dataset, and comparison between a baseline Deeplab v3+ model trained with PPCE and SEMEDA. SEMEDA produces better shaped predictions compared with the baseline.

4.2. Ablation study

In this section, we provide insight into the behavior of SEMEDA depending on the only additional hyperparameters $\{\lambda_l\}_{l=1...L}$. For Sobel, there is only one hyperparameter, the coefficient of the edge term λ_1. We perform ablation study on Cityscapes and VOC 2012 datasets with an ImageNet pre-trained Deeplab v3+/Xception 71 model. Results under the standard evaluation metric mIoU are shown in Table 1. While the Sobel baseline model does improve the accuracy slightly, particularly with $\lambda_1 > 1$, SEMEDA allows a far more significant accuracy boost in all tested configurations. Likely, this is due to the fact that through its convolutional layers, SEMEDA mixes the class-wise segmentation channels in a one-vs-one manner, while the Sobel baseline model does not, separating classes in a one-vs-all manner. Thus, SEMEDA encodes richer embeddings that more efficiently capture structure in the segmentation masks. It is worth noting that matching the output of the last layer corresponding to semantic edges (λ_3) contributes less to the performance. The reason is that the output of SEMEDA is a binary mask, where no distinction is made between the semantic edges belonging to different classes. In other words, in such a case, the presence of an edge at a specific location simply implies that this pixel marks a boundary between different objects whose categories are unknown. Therefore, this last layer of the SEMEDA network contains much less information than the first ones. To draw a parallel, this echoes results obtained in [5], where it is better to put more emphasis on the first layers of the SEMEDA network (in our case, λ_1 and $\lambda_2 > 1$).

Table 1
Comparison of results (% mIoU) on Cityscapes and VOC 2012 validation sets with different hyperparameter (λ_l) values and an ImageNet pre-trained Deeplab v3+/Xception 71 model. SEMEDA provides better results in all tested configurations.

Method	λ_1	λ_2	λ_3	Cityscapes	VOC 2012
PPCE	-	-	-	75.5	77.1
Sobel	0.5	-	-	75.4	77.3
	1	-	-	76.1	77.5
	2	-	-	76.2	77.4
	4	-	-	76.3	77.6
SEMEDA	0.5	0.5	0.5	76.9	77.6
	1	0.5	0	76.2	77.9
	0.5	1	0	76.3	77.8
	1	0.5	0.25	76.3	78.1
	1	1	0	76.1	78.5
	2	4	0	76.9	**78.8**
	4	2	0	**77.3**	78.6
	4	4	0	77.1	78.4

As many works suggest, the *mIoU* metric has many drawbacks and does not necessarily fully reflect the quality of the segmentation results. To this end, we also evaluate our methods against another commonly used metric, *F1-score*, as well as the recently proposed *E-measure* (enhanced-alignment measure [28]) which si-

Table 2
Different evaluation metrics (mIoU, F1-score and E-measure) on VOC 2012 val set. Our method is better than the baseline for all of these evaluation measures.

Metric	Method	aero	bike	bird	boat	bottle	bus	car	cat	chair	cow	table	dog	horse	mbike	person	plant	sheep	sofa	train	tv	mean
IoU	Baseline	86.3	40.1	**89.5**	71.6	81.6	94.6	86.4	93.0	39.3	87.9	54.8	89.1	85.4	82.2	85.2	58.8	88.7	50.1	85.1	76.1	77.1
	Semeda	**88.3**	**41.0**	88.8	**73.1**	**82.8**	**95.1**	**87.1**	**93.1**	**42.9**	**90.3**	**56.7**	**89.2**	**88.7**	**85.4**	**86.2**	**59.6**	**90.1**	**51.5**	**89.6**	**80.9**	**78.8**
F1-score	Baseline	92.6	57.2	**94.5**	83.5	89.8	97.2	92.7	96.4	56.4	93.5	70.8	94.2	92.1	90.3	92.0	74.0	94.0	66.8	92.0	86.4	85.9
	Semeda	**93.8**	**58.1**	94.1	**84.5**	**90.6**	**97.5**	**93.1**	96.4	**60.0**	**94.9**	**72.4**	**94.3**	**94.0**	**92.1**	**92.6**	**74.7**	**94.8**	**68.0**	**94.5**	**89.4**	**87.0**
E-measure	Baseline	92.9	74.3	**94.3**	87.9	69.6	90.4	**81.5**	94.4	68.7	**92.4**	67.9	**95.7**	93.2	92.9	88.0	65.9	91.9	**73.4**	93.3	83.6	85.1
	Semeda	**94.8**	**75.9**	93.8	87.9	**73.6**	**91.4**	81.2	**94.9**	**76.1**	92.2	**69.5**	95.1	**94.6**	**94.0**	88.0	**70.6**	**95.5**	73.0	**95.1**	**86.1**	**86.6**

Table 3
Different evaluation metrics (mIoU, F1-score and E-measure) on Cityscapes val set. Our method is better than the baseline for all of these evaluation measures.

Metric	Method	road	sidewalk	building	wall	fence	pole	light	sight	veg	terrain	sky	person	rider	car	truch	bus	train	mbike	bike	mean
IoU	Baseline	98.1	84.2	92.0	57.7	**60.4**	59.6	60.0	72.3	91.9	63.7	94.4	77.8	55.6	94.1	83.1	83.8	73.5	59.5	72.4	75.5
	Semeda	**98.2**	**85.3**	**92.6**	**58.3**	59.9	**62.4**	**63.6**	**75.1**	**92.4**	**63.9**	**94.9**	**79.9**	**59.2**	**95.0**	**85.2**	**89.2**	**76.9**	**62.0**	**74.2**	**77.3**
F1-score	Baseline	99.0	91.4	95.8	73.2	**75.3**	74.7	75.0	83.9	95.8	77.8	97.1	87.5	71.4	96.9	90.8	91.2	84.7	74.6	84.0	85.3
	Semeda	**99.1**	**92.01**	**96.2**	**73.7**	74.9	**76.9**	**77.8**	**85.8**	**96.1**	**77.9**	**97.4**	**88.8**	**74.4**	**97.5**	**92.0**	**94.3**	**86.9**	**76.5**	**85.2**	**86.5**
E-measure	Baseline	99.2	96.6	97.8	77.4	**87.8**	**94.6**	85.7	94.2	**98.5**	80.4	97.3	92.2	79.0	98.2	75.9	87.7	86.7	73.7	88.3	89.0
	Semeda	99.2	**96.7**	**97.9**	**79.0**	87.4	94.5	**87.8**	**94.4**	98.4	**81.1**	**97.4**	**92.7**	**80.0**	**98.5**	**83.6**	**87.5**	**87.7**	**77.3**	**88.7**	**90.0**

Algorithm 1 Train a segmentation network with SEMEDA.

Input:
I RGB Images
S^* Ground truth segmentation masks
Output:
θ^* Parameters of the segmentation net

// pre-training the SEMEDA network g_ϕ
for all batches $\mathcal{B}$ **do**
  for all masks $S_k^*, k = 1, \ldots, K$ in $\mathcal{B}$ **do**
    Generate ground truth edge map E_k^* from S_k^* by examining neighbouring pixels labels
    $\hat{E}_k^* = g_\phi(S_k^*)$
    $\phi \leftarrow \phi - \frac{1}{K}\frac{\partial}{\partial \phi}\mathcal{L}^{PPCE}(\hat{E}_k^*, E_k^*)$
  end for
end for
$\phi^* \leftarrow \phi$
// training the segmentation network f_θ
for all batches $\mathcal{B}$ **do**
  for all labelled images I_k in $\mathcal{B}$ **do**
    $\hat{S}_k = f_\theta(I_k)$
    $\theta \leftarrow \theta - \frac{1}{K}\frac{\partial}{\partial \theta}\mathcal{L}_{tot}^{g_{\phi^*}}(\hat{S}_k, S_k^*)$
  end for
end for
$\theta^* \leftarrow \theta$

Table 4
Results (%mIoU) on Cityscapes and VOC 2012 validation sets with Deeplab v2 and different pre-training strategies. For both pre-trained strategies, SEMEDA substantially enhances the performance.

Pre-train	Method	VOC 2012	Cityscapes
ImageNet	PPCE	72.9	64.1
	SEMEDA	**74.2**	**66.6**
ImageNet + COCO	PPCE	75.3	65.8
	SEMEDA	**76.5**	**67.8**

multaneously captures image level statistics and local pixel matching information. The comparison between the baseline model and the same model trained with SEMEDA on VOC 2012 val set (resp. Cityscapes val set) are shown in Table 2 (resp. Table 3). Our method is better than the baseline method for all these evaluation metrics on both datasets. Tables 2 and 3 also feature per-class comparison for each metric, showing that SEMEDA consistently improves the baseline accuracy for nearly every class and metric. This shows that SEMEDA allows to better capture the object structure by empathising object boundaries as well as enforcing continuity inside the objects.

4.3. Quantitative validation

Table 4 shows results obtained on VOC 2012 and Cityscapes datasets, with Deeplab v2 architecture and either Imagenet/MSCoco pre-training. Table 5 shows results obtained with and without SEMEDA with a stronger Deeplab v3+ baseline and two different backbone networks. In all tested configurations, regardless of the dataset, pre-training method, or backbone, SEMEDA substantially enhances the overall accuracy. Finally, Table 6 shows results obtained with Deeplab v3+ trained upon either PPCE or SEMEDA loss for face parsing on HELEN dataset. Similarly to benchmarks on VOC 2012 and Cityscapes, SEMEDA significantly enhances the segmentation accuracy for all classes by enforcing structure in the segmentation masks.

In order to more precisely assess this phenomenon, we evaluate SEMEDA on boundary/non-boundary trimaps (a narrow band around boundary illustrated in Fig. 4), as it was done in [26,38]: at test time, we divide the pixels into two subsets, whether they belong to a boundary (trimap) or non-boundary region, as indicated by the semantic edge maps generated from the ground truth segmentation masks. To do so, we vary the width of a band centered on the boundary and count as positive all the pixels in the region defined by this band, negative otherwise: thus, the more the width increases, the less precise the boundary definition is. Results are illustrated in Fig. 5 with Deeplab v3+ architecture. SEMEDA significantly enhances the mIoU on the boundary regions on both datasets, meanwhile, the Sobel baseline lies closer to the baseline performance. Particularly for strict boundaries (trimap width 2,3), the mIoU improvement is 4 pts on VOC 2012 and 2.4 pts on Cityscapes, which is considerable. On non-boundary regions, the improvement is also very significant on both datasets. This is due to the fact that SEMEDA strongly penalizes the presence of holes or discontinuities in the internal structure of the predicted objects (which are tagged as non-boundary on the ground truth markups). Thus, SEMEDA allows to better capture the structure of objects, as well as to refine the inter-class boundaries in the segmentation masks.

Table 5
Results (%mIoU) on Cityscapes and VOC 2012 validation sets with Deeplab v3+ pre-trained on ImageNet (with no additional pre-training on COCO or JFT). For both backbone networks, SEMEDA substantially enhances the performance.

Backbone	Method	VOC 2012	Cityscapes
ResNet-101	PPCE	74.4	72.4
	SEMEDA	**75.3**	**73.6**
Xception-71	PPCE	77.1	75.5
	SEMEDA	**78.8**	**77.3**

Table 6
Results (F_1-Score) for different face subparts on HELEN dataset with Deeplab v3+/Xception-71 models. SEMEDA provides better results on all classes.

	Brows	Eyes	Mouth	Overall
PPCE	74.76	82.66	87.8	85.84
SEMEDA	**75.42**	**83.5**	**90.37**	**87.36**

4.4. Comparison to leading methods

We compare our model with leading semantic segmentation methods on two datasets: Pascal VOC 2012 *test* set and Cityscapes *test* set. After finding the best model variant on val set, we then further fine-tune the model on the $train + val$ set. Our proposed SEMEDA attains the test set performance of 86.0% on VOC 2012 and 77.1% on Cityscapes, as shown in Tables 7 and 8, which

Table 7
Test set results on Pascal VOC 2012 (mIoU).

Method	mIoU
LRR 4x ResNet-CRF [40]	79.3
Deeplabv2 [9]	79.7
SegModel [41]	81.8
Deep Layer Cascade [42]	82.7
TuSimple [43]	83.1
Large Kernel Matters [44]	83.6
RefineNet [45]	84.2
PSPNet [46]	85.4
Deeplabv3 [39]	85.7
EncNet [47]	85.9
Deeplabv3+(reproduced)	85.0
Semeda (ours)	**86.0**

Fig. 10. Examples of predicted segmentation masks on HELEN dataset, and comparison between a baseline Deeplab v3+ model trained with PPCE and SEMEDA. SEMEDA produces better shaped predictions compared with the baseline.

Table 8
Test set results on Cityscapes (mIoU).

Method	Coarse	mIoU
Deeplabv2-CRF [9]	×	70.4
Deep Layer Cascade [42]	×	71.1
ML-CRNN [48]	×	71.2
LRR-4x [40]	√	71.8
RefineNet [45]	×	73.6
FoveaNet [49]	×	74.1
PSPNet [46]	×	76.3
Deeplabv3+(reproduced)	×	75.4
Semeda (ours)	×	**77.1**

are comparable with other leading methods. On both datasets, SEMEDA is better than the reproduced Deeplab v3+ model trained with PPCE loss. It is worth noting that, due to hardware limitations, we were unable to reproduce the results published in the Deeplab v3+ paper [35]. The biggest difference is that we have to use small batch size with fixed batch norm parameters during training. The importance of large batch size for training Deeplab models has been experimentally validated in [39]. The original Deeplab v3+ experiments [35] have set batch size at 16 to train the batch norm parameters, while in our experiments, we could only set batch size at 4 on VOC 2012 and at 2 on Cityscapes with fixed batch norm parameters. Other training strategies described in [35,39] such as decreasing output stride to 4 for training, duplicating the images that contain hard classes (on VOC 2012) and fine-tuning with coarse annotated dataset (Cityscapes) were also omitted for the same reason.

4.5. Qualitative assessment

To understand what SEMEDA network has learned, Fig. 6 illustrates the semantic edge embeddings learned by the first two CNN layers of the SEMEDA network. These embeddings are visually similar to traditional edge maps (c1,c2,c3), except that the filters of the SEMEDA network encompasses inter-class relationships in a one-vs-one fashion (d1,d2,d3), instead of simply separating object and other classes in a one-vs-all setting, as it is the case with simpler edge detector such as Sobel kernels. These richer embeddings can more efficiently encompass the structure in the segmentation masks.

To better illustrate the effectiveness of our proposed method, we showcase the entropy maps of the predictions in Fig. 7, illustrating the confidence of the predictions (the higher the entropy score, the less robust the classification is). Thus, with our method, high entropy activations (red points) only occur along the very boundaries of objects, whereas the baseline method produces high entropy scores both inside and at the boundary of objects. These results qualitatively prove that our approach not only refines the accuracy on the boundary pixels, but also makes the predictions within the object more certain and more uniform.

Figs. 8, 9, 10 show segmentation masks outputted with Deeplab v3+/Xception 71 trained with SEMEDA and with PPCE only (baseline). For each image, the segmentation mask provided by the network is overlayed with the input image. As observed, predictions generally conform better geometric edges when SEMEDA is added. For example, in the first image of VOC 2012 (Fig. 8), SEMEDA corrects wrong predictions (class dog (purple) confused with class horse (pink)) and produces better shaped predictions. This improvement is consistent on other datasets where predictions of different classes such as person, sidewalk and brows, are better conformed with geometric edges. As stated above, SEMEDA allows to better capture the structure of the segmented objects, by putting more emphasis on the inter-class boundaries, as well as to avoid discontinuities (e.g. holes) inside the objects. Notice, for instance, how fine-grained elements such as traffic signs, tree leaves or people shapes are better captured with edge-aware loss on Cityscapes (Fig. 9), and how well the segmentation fits the objects on VOC 2012 (Fig. 8) and the faces on HELEN (Fig. 10).

5. Conclusion

In this paper, we proposed a new learning strategy for semantic segmentation. Our approach leverages a semantic edge-aware loss for implicitly integrating structural information into segmentation predictions. It consists of training a semantic edge detection (SEMEDA) network to map segmentation masks to the corresponding edge maps. The predictions outputted by the segmentation network can then be optimized (via the proposed SEMEDA loss) in the embedding space of the semantic edge detection network, similarly to perceptual losses.

Through extensive evaluation on several datasets with very different application contexts, we showed that SEMEDA significantly improves the overall performance of semantic segmentation networks in all tested hyperparameter configurations, segmentation network architectures, backbone networks, and pre-training strategies. More precisely, we showed that SEMEDA works by enforcing inter-class boundary structure as well as avoiding holes in the segmented objects. In addition, SEMEDA does not require any additional annotation and negligible computational overhead, thus can be straightforwardly combined with traditional losses for improving the performance of any semantic segmentation network.

The proposed work leads us to rethink how structural information, such as semantic edge detection can be integrated into existing segmentation architectures for enhanced precision, beyond merely treating semantic edge detection and segmentation in a naive multi-task fashion. As such, it opens up a new space for designing edge-enhanced semantic segmentation architectures. SEMEDA naturally fits into the various applications of semantic segmentation, such as autonomous driving or medical imaging systems. In addition, SEMEDA could be adapted without bells and whistles to closely related domains, such as instance segmentation or body part detection. Last but not least, we showed in this work that the embeddings of the SEMEDA network contains rich information related to the structure of the objects. Hence, these embeddings could be used in the context of semantic segmentation on videos, for instance to generate better quality super-trajectories (extending [50]) by matching similar objects with enforced semantic edge-related structure constraints.

Declaration of Competing Interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

References

[1] J. Long, E. Shelhamer, T. Darrell, Fully convolutional networks for semantic segmentation, IEEE Trans. Patt. Anal. Mach. Intell. 39 (4) (2017).
[2] C. Dong, C.C. Loy, K. He, X. Tang, Image super-resolution using deep convolutional networks., IEEE Trans. Patt. Anal. Mach. Intell. 38 (2) (2016).
[3] Z. Cheng, Q. Yang, B. Sheng, Deep colorization, in: Proceedings of IEEE International Conference on Computer Vision, pp.415–423, 2015.
[4] R. Zhang, P. Isola, A. Efross, Colorful image colorization, in: Proceedings of European Conference on Computer Vision, pp.649–666, 2016.
[5] J. Johnson, A. Alahi, L. Fei-Fei, Perceptual losses for real-time style transfer and super-resolution, in: Proceedings of European Conference on Computer Vision, 2016, pp. 694–711.
[6] J. Shen, X. Dong, J. Peng, X. Jin, L. Shao, F. Porikli, Submodular function optimization for motion clustering and image segmentation, IEEE Trans. Neur. Netw. Learn. Syst. 30 (9) (2019) 2637–2649.
[7] J. Shen, J. Peng, X. Dong, L. Shao, F. Porikli, Higher order energies for image segmentation, IEEE Trans. Image Process. 26 (10) (2017) 4911–4922.
[8] L.C. Chen, G. Papandreou, I. Kokkinos, K. Murphy, A.L. Yuille, Semantic image segmentation with deep convolutional nets and fully connected crfs., in: International Conference on Learning Representations, 2015.
[9] L. Chen, G. Papandreou, I. Kokkinos, K. Murphy, A. Yuille, Deeplab: semantic image segmentation with deep convolutional nets, atrous convolution, and fully connected crfs, IEEE Trans. Patt. Anal. Mach. Intell. 40 (2016) 834–848.
[10] H. Noh, S. Hong, B. Han, Learning deconvolution network for semantic segmentation, in: Proceedings of IEEE International Conference on Computer Vision, 2015, pp. 1520–1528.
[11] Y. Wang, J. Liu, Y. Li, J. Fu, M.Xu, H. Lu, Hierarchically supervised deconvolutional network for semantic video segmentation, Pattern Recognit 64 (2017) 437–445.
[12] M.D. Zeiler, G.W. Taylor, R. Fergus, Adaptive deconvolutional networks for mid and high level feature learning, in: Proceedings of IEEE International Conference on Computer Vision, 2011, pp. 2018–2025.
[13] V. Badrinarayanan, A. Kendall, R. Cipolla, Segnet: a deep convolutional encoder-decoder architecture for image segmentation, IEEE Trans Pattern Anal Mach Intell 39 (12) (2017) 2481–2495.
[14] O. Ronneberger, P. Fisher, T.Brox, U-net: Convolutional networks for biomedical image segmentation, in: Medical Image Computing and Computer-Assisted Intervention, 2015, pp. 234–241.
[15] Y. Pang, Y. Li, J. Shen, L. Shao, Towards bridging semantic gap to improve semantic segmentation, in: The IEEE International Conference on Computer Vision (ICCV), 2019.
[16] Z. Zhang, H. Fu, H. Dai, J. Shen, Y. Pang, L. Shao, Et-net: A generic edge-attention guidance network for medical image segmentation, in: D. Shen, T. Liu, T.M. Peters, L.H. Staib, C. Essert, S. Zhou, P. Yap, A. Khan (Eds.), Medical Image Computing and Computer Assisted Intervention - MICCAI 2019 - 22nd International Conference, Shenzhen, China, October 13–17, 2019, Proceedings, Part I, Lecture Notes in Computer Science, 11764, Springer, 2019, pp. 442–450, doi:10.1007/978-3-030-32239-7_49.
[17] H. Zhao, J. Shi, X. Qi, J. Jia, Pyramid scene parsing network, in: Proceedings of IEEE Conference on Computer Vision and Pattern Recognition, 2017, pp. 6230–6239.
[18] J. Fu, J. Liu, H. Tian, Y. Li, Y. Bao, Z. Fang, H. Lu, Dual attention network for scene segmentation, in: The IEEE Conference on Computer Vision and Pattern Recognition, 2019.
[19] Z. Huang, X. Wang, L. Huang, C. Huang, Y. Wei, W. Liu, Ccnet: Criss-cross attention for semantic segmentation, in: The IEEE International Conference on Computer Vision, 2019.
[20] H. Zhao, Y. Zhang, S. Liu, J. Shi, C.C. Loy, D. Lin, J. Jia, PSANet: Point-wise spatial attention network for scene parsing, in: Proceedings of European Conference on Computer Vision, 2018, pp. 270–286.
[21] P. Zhang, W.Liu, H. Wang, Y. Lei, H. Lu, Deep gated attention networks for large-scale street-level scene segmentation, Patt. Recognit. 88 (2018) 702–714.
[22] T.-Y. Lin, P. Goyal, R. Girshick, K. He, P. Dollar, Focal loss for dense object detection, in: International Conference on Computer Vision, 2017.
[23] M. Berman, A. Triki, M. Blaschko, The lovász-softmax loss: A tractable surrogate for the optimization of the intersection-over-union measure in neural networks, in: Proceedings of IEEE Conference on Computer Vision and Pattern Recognition, 2018, pp. 4413–4421.
[24] G. Nagendar, D. Singh, V. Balasubramanian, C. Jawahar, Neuroiou: Learning a surrogate loss for semantic segmentation, in: British Machine Vision Conference, 2018.
[25] C.H. Sudre, W. Li, T. Vercauteren, S. Ourselin, M.J. Cardoso, Generalised dice overlap as a deep learning loss function for highly unbalanced segmentations, in: Deep Learning in Medical Image Analysis and Multimodal Learning for Clinical Decision Support - Third International Workshop, DLMIA 2017, and 7th International Workshop, ML-CDS 2017, Held in Conjunction with MICCAI 2017, Québec City, QC, Canada, September 14, 2017, Proceedings, 10553, Springer, 2017, pp. 240–248.
[26] G. Csurka, D. Larlus, F. Perronnin, What is a good evaluation measure for semantic segmentation? in: British Machine Vision Conference, 2013.
[27] S. Chen, C. Ding, M. Liu, Dual-force convolutional neural networks for accurate brain tumor segmentation, Patt. Recognit. 88 (2018) 90–100.

[28] D.-P. Fan, C. Gong, Y. Cao, B. Ren, M.-M. Cheng, A. Borji, Enhanced-alignment measure for binary foreground map evaluation, in: Proceedings of the 27th International Joint Conference on Artificial Intelligence, in: IJCAI18, AAAI Press, 2018, p. 698704.
[29] D.-P. Fan, M.-M. Cheng, Y. Liu, T. Li, A. Borji, Structure-measure: A New Way to Evaluate Foreground Maps, ICCV, 2017.
[30] R. Margolin, L. Zelnik-Manor, A. Tal, How to evaluate foreground maps, in: 2014 IEEE Conference on Computer Vision and Pattern Recognition, 2014, pp. 248–255.
[31] P. Luc, C. Couprie, S. Chintala, J. Verbeek, Semantic segmentation using adversarial networks, in: Conference on Neural Information Processing Systems, Workshop on Adversarial Training, 2016.
[32] M. Zhen, J. Wang, L. Zhou, T. Fang, L. Quan, Learning fully dense neural networks for image semantic segmentation, in: AAAI Conference on Artificial Intelligence, 2019.
[33] Y. Liu, M. Cheng, J. Bian, L. Zhang, P. Jiang, Y. Cao, Semantic edge detection with diverse deep supervision, CoRR abs/1804.02864 (2018).
[34] K. Simonyan, A. Zisserman, Very deep convolutional networks for large-scale image recognition, in: International Conference on Learning Representations, 2015.
[35] L.C. Chen, Y. Zhu, G. Papandreou, F. Schroff, H. Adam, Encoder-decoder with atrous separable convolution for semantic image segmentation, in: Proceedings of European Conference on Computer Vision, 2018, pp. 833–851.
[36] M. Everingham, S.A. Eslami, L. Van Gool, C.K. Williams, J. Winn, A. Zisserman, The pascal visual object classes challenge: aretrospective, Int. J. Comput. Vis. 111 (1) (2015) 98–136.
[37] B. Hariharan, P. Arbeláez, L. Bourdev, S. Maji, J. Malik, Semantic contours from inverse detectors, in: Proceedings of IEEE International Conference on Computer Vision, 2011, p. 991998.
[38] L.C. Chen, J.T. Barron, G. Papandreou, K. Murphy, A.L. Yuille, Semantic image segmentation with task-specific edge detection using cnns and a discriminatively trained domain transform, in: Proceedings of IEEE Conference on Computer Vision and Pattern Recognition, 2016, pp. 4545–4554.
[39] L.C. Chen, G. Papandreou, F. Schroff, H. Adam, Rethinking atrous convolution for semantic image segmentation, arXiv preprint arXiv:1706.05587, 2017.
[40] G. Ghiasi, C.C. Fowlkes, Laplacian pyramid reconstruction and refinement for semantic segmentation, in: Computer Vision - ECCV 2016 - 14th European Conference, Amsterdam, The Netherlands, October 11–14, 2016, Proceedings, Part III, 9907, Springer, 2016, pp. 519–534.
[41] F. Shen, R. Gan, S. Yan, G. Zeng, Semantic segmentation via structured patch prediction, context crf and guidance crf, in: 2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2017, pp. 5178–5186.
[42] P.L.C.C.L. Xiaoxiao Li Ziwei Liu, X. Tang, Not all pixels are equal: Difficulty-aware semantic segmentation via deep layer cascade, in: Proceedings of IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2017.
[43] P. Wang, P. Chen, Y. Yuan, D. Liu, Z. Huang, X. Hou, G. Cottrell, Understanding convolution for semantic segmentation, in: 2018 IEEE Winter Conference on Applications of Computer Vision (WACV), 2018, pp. 1451–1460.
[44] C. Peng, X. Zhang, G. Yu, G. Luo, J. Sun, Large kernel mattersimprove semantic segmentation by global convolutional network, in: Conference on Computer Vision and Pattern Recognition, 2017.
[45] G. Lin, A. Milan, C. Shen, I. Reid, Refinenet: Multi-path refinement networks for high-resolution semantic segmentation, in: Conference on Computer Vision and Pattern Recognition, 2017.
[46] H. Zhao, J. Shi, X. Qi, X. Wang, J. Jia, Pyramid scene parsing network, CVPR, 2017.
[47] H. Zhang, K.J. Dana, J. Shi, Z. Zhang, X. Wang, A. Tyagi, A. Agrawal, Context encoding for semantic segmentation, in: 2018 IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2018, Salt Lake City, UT, USA, June 18–22, 2018, 2018, pp. 7151–7160.
[48] H. Fan, X. Mei, D. Prokhorov, H. Ling, Multi-level contextual rnns with attention model for scene labeling, IEEE Trans. Intell. Transp. Syst. 19 (11) (2018) 3475–3485.
[49] X. Li, Z. Jie, W. Wang, C. Liu, J. Yang, X. Shen, Z. Lin, Q. Chen, S. Yan, J. Feng, Foveanet: Perspective-aware urban scene parsing, in: The IEEE International Conference on Computer Vision (ICCV), 2017.
[50] W. Wang, J. Shen, F. Porikli, R. Yang, Semi-supervised video object segmentation with super-trajectories, Trans. Patt. Analy. Mach. Intell. 41 (4) (2018).

Yifu Chen received a M.Sc in Probability and Finance from Sorbonne University and he received the Engineering Diploma in applied mathematics from Ecole Polytechnique in 2016. He is currently a Ph.D. student in the Machine Learning and Information Access of Sorbonne University, under the supervision of Matthieu Cord. His research focuses on the semantic image segmentation and visualisation and explanation of deep neural networks.

Arnaud Dapogny is a computer vision researcher at Datakalab in Paris, France. He obtained an engineering degree from SUPELEC engineering school and a masters degree from Sorbonne University in 2013, as well as a Ph.D in computer vision for face analysis in 2016. He also worked as a post-doctoral fellow in Sorbonne University. His research interests include computer vision for face analysis (facial expressions, face recognition, landmark alignment) as well as scene understanding and deep generative models.

Matthieu Cord is a full professor at the Computer Science Laboratory (LIP6) of Sorbonne University since 2006. He is also working part-time at the valeo.ai research laboratory. He is a laureate of a chair of research and teaching in artificial intelligence from the national french government program on AI 2020 entitled VISADEEP: Towards visual reasoning in deep learning. He is an honorary member of the Institut Universitaire de France (2009) and served from 2015 to 2018 as AI expert at CNRS and ANR. His research expertise includes computer vision, machine learning and artificial intelligence. He is the author of more than 150 international scientific publications on visual information retrieval, pattern recognition using deep learning, and multimodal vision and language understanding.