Deep Image Segmentation based on Mutual Information

Leonardo Capozzi and Tiago Gonçalves
Faculdade de Engenharia
Universidade do Porto
Porto, Portugal
{up201503708, up201607753}@fe.up.pt

Abstract

The use of deep neural networks has achieved many advancements in many different areas, namely computer vision. Information theory concepts such as the Kullback-Leibler Divergence is often used in deep learning methodologies as optimisation criterion since it quantifies the difference between two probability distributions. Image segmentation is a computer vision problem where the goal is to classify individual pixels of an image. It has many real-world applications such as self-driving cars, medical imaging and object detection, to name a few. In this paper, we present a comparative study of two methodologies based on mutual information applied to the task of deep image segmentation. We use the Cityscapes Dataset and PASCAL Visual Object Classes (VOC) 2012 databases to benchmark the performance of the algorithms and compare them with a baseline, showing that there can be some advantages to this type of regularisation.

1. Introduction

The use of deep neural networks has brought a lot of advancements in various areas of research, namely computer vision. Image segmentation is a computer vision problem where the goal is to classify individual pixels of an image as belonging to a certain class. This allows to detect the location of certain objects/features in an image. It has many real-world applications, such as self-driving cars (e.g., pedestrian detection, navigable surface), traffic control systems, object detection, recognition tasks, medical imaging (e.g., tumour boundary), and many others. Over the years, numerous algorithms have been developed to address this problem, with most of them relying on more "handcrafted" features. These include methods such as thresholding, histogram-based bundling, k-means clustering and Markov random fields [15]. In recent years, deep learning methodologies have brought performance improvements compared to methods that did not use deep learning. Information theory is based on probability and statistics and is often used to study the information of the distributions of random variables. For instance, in deep learning, concepts such as the Kullback-Leibler Divergence, which quantifies the difference between two probability distributions, are widely used as optimisation criteria.

In this paper, we present a comparative study of two methodologies based on mutual information applied to the task of deep image segmentation. In [27], Zhao *et al.* propose an image segmentation methodology that uses a clustering loss based on mutual information. This loss explicitly enforces consistency in the prediction of unlabeled images and simultaneously optimises the network to correctly predict the labels of annotated images. In [25], Zhang *et al.* propose a deep mutual learning strategy where a collection of student networks learn simultaneously with help from each other. This is a variation of the model distillation technique, where a powerful large network is used to transfer knowledge to a smaller and more computationally efficient network.

Besides the Introduction, the rest of this paper is organised as follows: section 2 presents the main concepts of the field of deep learning for image segmentation and its relationship with information theory, section 3 describes the data sets and the details of the implementation of the experiments, section 4 shows the results and exposes a brief discussion and section 5 concludes this paper and proposes future research directions. The code related to this paper is available in a GitLab repository[1].

2. Related Work

2.1. Deep Learning for Image Segmentation

One of the first successful approaches of deep learning for image segmentation was the Fully Convolutional Network (FCN) proposed by Long *et al.* [14]. FCN contains only convolutional layers, thus allowing the inputs to

[1] https://gitlab.inesctec.pt/tiago.f.goncalves/image-segmentation-mutual-information

be of arbitrary size. In their implementation, the authors replaced all the fully-connected layers of several state-of-the-art models for image classification (*e.g.*, VGG16 [22], GoogLeNet [23]) by convolutional layers. Hence, instead of outputting classification scores, the model outputs a spatial segmentation map. Despite its predictive performance, architectures based only on the FCN had several limitations, namely: 1) being slow during real-time inference, 2) ignoring the global context information and 3) being difficult to adapt to three-dimensional (3D) images. Therefore, most of the work proposed after the FCN is built on top of it and often addresses at least one of its limitations [16]. An interesting example is the ParseNet proposed by Liu *et al.* [13]. This model added global pooling operations to produce augmented feature maps and to add information about the global context during training. Another interesting class of deep learning models for image segmentation relies on the application of encoder-decoder architectures. The first model was proposed by Noh *et al.* and was one of the pioneers in the application of the transposed convolution (also known as *deconvolution*). He used the structure of the VGG16 and built a deep neural network composed of an encoder with convolutional layers to extract features and a decoder that applies transposed convolutions and unpooling operations to achieve a pixel-wise classification mask in the end. This paved the way to more complex models such as SegNet [1] and U-Net [21]. Proposed by Badrinarayanan *et al.*, SegNet proposed a novel strategy for the decoder to upsample the feature maps from the encoder by using pooling indices computed in the max-pooling step of the encoder to perform non-linear upsampling, thus removing the need for learning how to upsample. U-Net was proposed by Ronneberger *et al.* for the segmentation of images of electron microscopy (see Figure 1). This architecture has the main advantage of being capable to learn from few annotated images effectively because it also relies on the use of data augmentation. Inspired by the FCN, the encoder of the U-Net performs down-sampling operations and extracts features through several convolutional layers; the main goal of this module is to capture context. On the other hand, the decoder, which is symmetric to the encoder, performs up-sampling operations through transposed convolutions, thus reducing the number of feature maps and increasing the resolution of the outputs. In the end, a final convolution is employed to obtain a pixel-wise segmentation map. Another interesting novelty brought by the U-Net is the concatenation of feature maps generated by the encoder to the feature maps generated by the decoder: these "connections" are called *skip connections* and facilitate the proper flow of information, thus allowing the model to train with fewer problems and to avoid gradient-vanishing problems [11].

Later, Lin *et al.* proposed the Feature Pyramid Network (FPN), an architecture that could deal with multi-scale features [12] and Zhao *et al.* developed the Pyramid Scene Parsing Network (PSPN), a multi-scale network that also addressed the learning of global context representation [26]. Recently, another operation has been introduced in CNNs: the dilated convolution (also known as *atrous* convolution), which introduced a new hyper-parameter, the dilation rate. The nature of this operation allows us to enlarge the receptive field of the deep neural networks with no increase in computational cost. For instance, the state-of-the-art DeepLabV3+ model is based on the use of this operation along with an encoder-decoder architecture and the Atrous Spatial Pyramid Pooling (ASPP) operation, which allows us to extract features at multiple sampling rates and to capture context information at multiple scales [3].

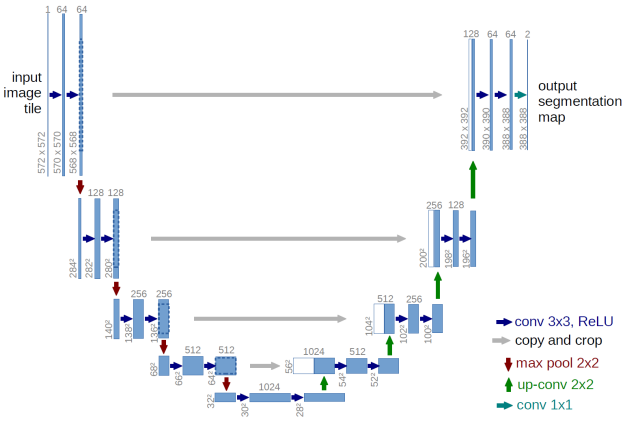

Figure 1. U-Net architecture, from [21].

2.2. Concepts of Information Theory

We present the fundamental concepts of information theory, needed to understand the intuition behind our analysis. All the concepts are presented as in [5].

2.2.1 Information

Claude Shannon derived a way of measuring information content called "self-information". Formally, it is defined as:

$$I_X(x) = -log_2[p_X(x)] \tag{1}$$

where X is a random variable with probability mass function $p_X(x)$.

Shannon's definition of information follows several axioms:

1. An event with a probability of 100% is completely unsurprising and therefore yields no information.

2. The lower the probability of an event the more information it yields, as it is more surprising.

3. The total amount of information of two independent events measured separately is the sum of the information of the individual events.

2.2.2 Entropy

The definition of entropy is the expected information content of random variable X. It is defined as:

$$H(X) = -\sum_{x} p_X(x) log_2[p_X(x)] \tag{2}$$

The joint entropy of two random variables X and Y is defined as:

$$H(X,Y) = -\sum_{x,y} p(x,y) log_2[p(x,y)] \tag{3}$$

The conditional entropy is defined as the entropy of random variable X given Y, and is written as:

$$H(X|Y) = -\sum_{y} p(y) \sum_{x} p(x|y) log_2[p(x|y)] \tag{4}$$

2.2.3 Mutual Information

Mutual information refers to the amount of information that can be obtained about one random variable by observing another random variable. The mutual information between two random variables X and Y is given as:

$$I(X;Y) = \sum_{x,y} p(x,y) log_2 \frac{p(x,y)}{p(x)p(y)} \tag{5}$$

Mutual information is also related to entropy in the following ways:

$$I(X;Y) = H(X) - H(X|Y) \tag{6}$$

$$I(X;Y) = H(X) + H(Y) - H(X,Y) \tag{7}$$

2.2.4 Kullback-Leibler Divergence

The Kullback-Leibler Divergence measures how one probability distribution is different from another. When two distributions have the same quantity of information their relative entropy is equal to 0.

Given two discrete probability distributions P and Q, with the same probability space X, the relative entropy from Q to P is defined as:

$$D_{KL}(P \parallel Q) = \sum_{x \in X} P(x) log(\frac{P(x)}{Q(x)}) \tag{8}$$

2.3. Mutual Information and Deep Learning

An interesting approach to the application of information theory concepts to the field of deep learning is related to model distillation methodologies. Despite their predictive performance, state-of-the-art deep learning methods are often very large in depth or width and contain a large number of parameters [25]. This may represent a disadvantage in terms of execution time or storage memory. To overcome this problem, several research lines on smaller and simpler models have been developed, such as model compression [7], pruning [10], binarisation [19] or model distillation [8]. The main assumption behind the application of model distillation is that small networks have the same *representation capacity* as large networks but may be more difficult to optimise. The distillation approach starts with the learning of a complex and powerful network (*i.e.*, the *teacher* network). Then, the objective is to train a smaller network (*i.e.*, the *student* network) to mimic the teacher, which, according to the literature [20], may be easier than learning the target function directly. Another approach that is directly related to model distillation is *mutual learning*. While in model distillation the main objective is that the student can learn from a pre-trained network (one-way knowledge transfer), in mutual learning, both the networks are training and the main goal is that both mimic each other.

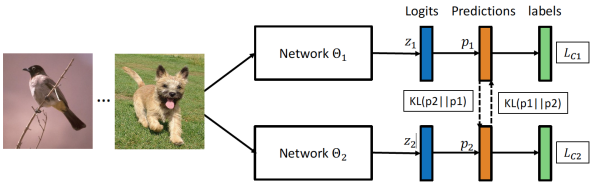

Figure 2. Deep mutual learning methodology, proposed [25].

Zang *et al.* [25] explored this strategy using information theory concepts (see Figure 2). In their approach, they train two networks, N_1, N_2. Let p_1 and p_2 be the predictions of N_1 and N_2, respectively. To measure the match of these network's predictions, they add a term in the loss based on the Kullback-Leibler Divergence. For instance, during the training of N_1, the goal is to minimise the Kullback-Leibler Divergence from p_1 to p_2, $D_{KL}(p_2 \parallel p_1)$, whereas, during the training of N_2, the goal is to minimise the Kullback-Leibler Divergence from p_2 to p_1, $D_{KL}(p_1 \parallel p_2)$. The networks are trained in a jointly manner [25].

Recently, Zhao *et al.* proposed a *region mutual information* loss for semantic segmentation [27]. In their work, they state that most segmentation models use pixel-wise loss functions to optimise the weights, which end up ignoring the dependencies between the pixels of an image. To overcome this issue, they model the image as a multi-dimensional distribution that encodes the relationship be-

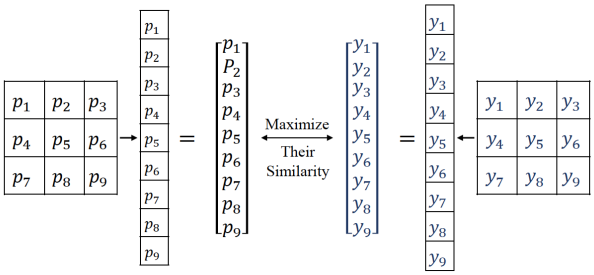

Figure 3. The intuition behind the RMI loss, from [27].

tween pixels. This is achieved by representing a given pixel by itself and its neighbours. The main goal is to maximise the mutual information between the multi-dimensional distributions of the prediction and ground-truth, however, since the real-value of mutual information may be hard to compute, the authors derive a lower bound of the mutual information and maximise this lower bound to maximise the real value of the mutual information [27] (see Figure 3). A similar methodology has also been proposed by Peng *et al.*, for the regularisation of semi-supervised segmentation models [18].

3. Methodology

3.1. Data

We performed experiments with two benchmark data sets, explained below.

3.1.1 Cityscapes Dataset

The Cityscapes Dataset[2] is a database composed of urban street scenes recorded from 50 different cities [4]. The train, validation and test splits are composed of 2975, 500 and 1525 images, respectively. This data set has 19 annotated classes.

3.1.2 PASCAL Visual Object Classes (VOC) 2012

The PASCAL Visual Object Classes (VOC) 2012[3] is a database composed of several visual object classes in realistic scenes. It was created for the Visual Object Classes Challenge 2012 (VOC2012) [6]. The train and validation splits are composed of 1464 and 1449 images (test split is not available). This data set has 21 annotated classes.

3.2. Implementation

This comparative study is composed of three use-cases, explained below. We used U-Net in all experiences because it is a model with relatively few parameters and does not require too many computational resources. Moreover, it was also our intention to study if we could achieve better results with this model through the introduction of information theory concepts in the training regularisation. The algorithms were implemented using Python 3.7 and the deep learning library PyTorch [17]. The U-Net implementation used in this report is based in the implementation by Buda *et al.* [2][4].

3.2.1 Data Processing

This data processing strategy is employed in all the use-cases: 1) the input images are read and are given to the models in RGB format; 2) the images are then resized into the size 234×234; 3) a random-crop is then applied to the images and they reach the final size of 224×224. Since this is a segmentation task, our labels are also images (in this case, each class has its segmentation mask), so the strategy that is employed for the input images is also performed for the labels. In the end, the input data has the shape $[b, 3, 224, 224]$ and the output data has the shape $[b, nc, 224, 224]$, where b is the batch size and nc is the number of output classes.

3.2.2 U-Net: Baseline

The baseline model is trained for 200 epochs, with batch size 8. The loss function is given by the cross-entropy and the model's parameters are optimised using Adaptive Momentum (Adam) [9] optimiser with learning rate 1×10^{-3}. The data processing strategy is employed as described in 3.2.1. The best model in training is saved according to a decrease in the loss value.

3.2.3 U-Net: Region Mutual Information Loss

In this use-case, we apply the loss proposed by [27] and described in 2.3. The model with region mutual information loss is trained for 200 epochs, with batch size 2. The loss function is given by $\mathcal{L} = \lambda L_{\text{BCE}} + (1-\lambda) L_{\text{RMI}}$, where L_{BCE} is the binary cross-entropy loss, L_{RMI} is the region mutual information loss and $\lambda = 0.5$ is a weight coefficient applied to the loss terms. The parameters of the model are optimised using Adam optimiser with learning rate 1×10^{-4}. The data processing strategy is employed as described in 3.2.1. The best model in training is saved according to a decrease in the loss value.

3.2.4 Deep Mutual Learning: DeepLabV3 and U-Net

In this use-case we apply the training strategy proposed by [25] and described in 2.3. We employ a joint training

[2] https://www.cityscapes-dataset.com
[3] http://host.robots.ox.ac.uk/pascal/VOC/voc2012
[4] https://pytorch.org/hub/mateuszbuda_brain-segmentation-pytorch_unet/

strategy with a *teacher* model and a *student model*. In this work, we used the DeepLabV3 architecture as the teacher model and U-Net as the student model. This pipeline is trained for 200 epochs, with batch size 10. The loss function of the teacher model is given by $\mathcal{L}_{\text{teacher}} = L_{\text{CE}} + \mathrm{D}_{KL}(p_{\text{student}} \parallel p_{\text{teacher}})$, where L_{CE} is the cross-entropy loss and $\mathrm{D}_{KL}(p_{\text{student}} \parallel p_{\text{teacher}})$ is the Kullback–Leibler divergence from the teacher predictions, p_{teacher}, to the student predictions p_{student}. The loss function of the student model is given by $\mathcal{L}_{\text{student}} = L_{\text{CE}} + \mathrm{D}_{KL}(p_{\text{teacher}} \parallel p_{\text{student}})$, where L_{CE} is the cross-entropy loss and $\mathrm{D}_{KL}(p_{\text{teacher}} \parallel p_{\text{student}})$ is the Kullback–Leibler divergence from the student predictions, p_{student}, to the teacher predictions p_{teacher}. Both the teacher and student models' parameters are optimised using Adam optimiser with learning rate 1×10^{-4}. The best models in training are saved according to a decrease in their respective loss values.

3.3. Performance Metrics

To assess the predictive performance of the deep learning algorithms we use two performance metrics: 1) intersection over union (IoU) and 2) accuracy. Let TP, TN, FP and FN be the number of true positive pixels, the number of true negative pixels, the number of false-positive pixels and the number of false-negative pixels, respectively. In a binary setting, the IoU and the accuracy can be computed according to Equation 9 and Equation 10.

$$\text{IoU} = \frac{TP}{TP + FP + FN} \tag{9}$$

$$\text{Accuracy} = \frac{TP + TN}{TP + FP + FN + TN} \tag{10}$$

In a multi-class setting, the IoU can be computed by adding all class IoUs (*i.e.*, for each class, sum all TP) and divide it by the sum of all class unions (*i.e.*, the sum of TP, FP and FN over all classes). The accuracy can be computed by summing the number of pixels that were correctly classified (*i.e.*, the pixels where the *ground truth* class and *predicted* class match) divided by the total amount of pixels in the image.

4. Results and Discussion

4.1. U-Net: Baseline

Table 1 presents the accuracy and IoU values obtained for this use-case in both data sets and Figure 4 and Figure 5 present some examples of predicted masks for the Cityscapes Dataset and the PASCAL VOC 2012 data sets, respectively. In this use-case, we note the clear differences in performance between the two data sets, since the Cityscapes Dataset achieves better results than PASCAL VOC 2012.

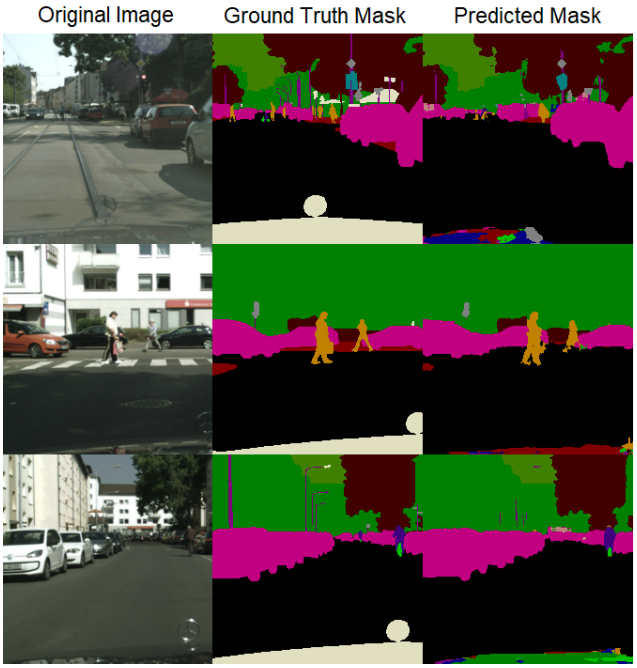

Figure 4. Examples of predicted masks for the Cityscapes Dataset using "U-Net: Baseline".

Figure 5. Examples of predicted masks for the PASCAL VOC 2012 using "U-Net: Baseline".

4.2. U-Net: Region Mutual Information Loss

Table 2 presents the accuracy and IoU values obtained for this use-case in both data sets and Figure 6 and Figure 7 present some examples of predicted masks for the

Table 1. Results obtained for the "U-Net: Baseline" use-case.

Dataset	Intersection over Union (IoU)	Accuracy
Cityscapes Dataset	0.5931	0.9388
PASCAL VOC 2012	0.1965	0.7367

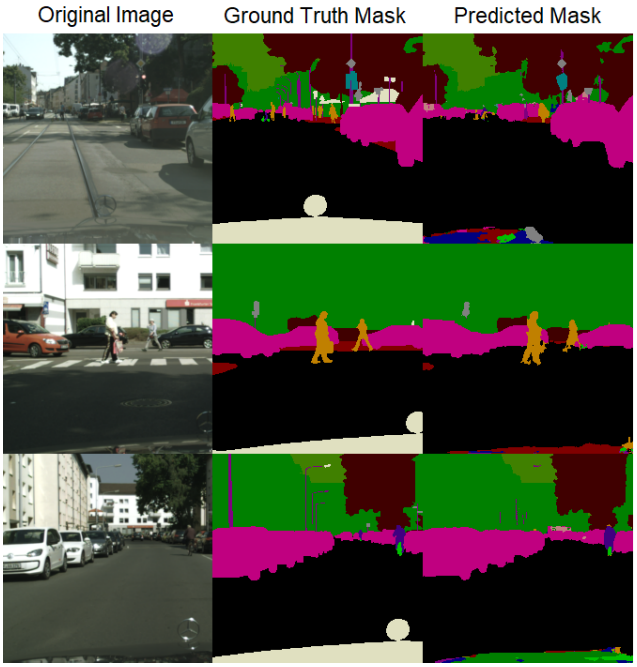

Figure 6. Examples of predicted masks for the Cityscapes Dataset using "U-Net: Region Mutual Information Loss".

Figure 7. Examples of predicted masks for the PASCAL VOC 2012 using "U-Net: Region Mutual Information Loss".

Cityscapes Dataset and the PASCAL VOC 2012 data sets, respectively. In this use-case, we note that the predictive performance does not significantly change in the Cityscapes Dataset, however, it achieves worse results in the PASCAL VOC 2012, when compared to the baseline, which is counter-intuitive.

4.3. Deep Mutual Learning: DeepLabV3 and U-Net

Table 3 presents the accuracy and IoU values obtained for this use-case in both data sets and Figure 8 and Figure 9 present some examples of predicted masks for the Cityscapes Dataset and the PASCAL VOC 2012 data sets, respectively. In this use-case, we note that the predictive performance does not significantly change in the Cityscapes Dataset, however, it achieves the best result in the PASCAL VOC 2012, when compared to the previous use-cases. Assuming that the PASCAL VOC 2012 is difficult to learn (this assumption comes from the results obtained for the PASCAL VOC 2012 in the previous use-cases), this is an interesting result, since it suggests that minimising the Kullback-Leibler Divergence between the predictions of student network and the predictions of the teacher network may be beneficial.

5. Conclusions and Future Work

This work presented a comparative study between methodologies based on the mutual information applied to deep learning for image segmentation. We used U-Net as the baseline algorithm and we studied three different use-cases: 1) baseline U-Net; 2) U-Net with RMI loss; 3) U-Net as a student network in a deep mutual learning strategy. Results suggest that there are gains in applying a deep mutual learning strategy, as can be observed in the results obtained for the PASCAL VOC 2012. Surprisingly, the application of the RMI loss does not seem to bring improvements to the predictive performance of the U-Net model in these data sets. Although this can be due to an unknown hyper-parameter that still needs further tuning, some authors suggest that some of the reported results regarding the benefits of using mutual information to optimise deep learning algorithms, may not be attributed to the properties of mutual information alone [24].

Hence, further work should be devoted to the evaluation of different deep image segmentation architectures on the same use-case and to the study of different mutual information properties that could be incorporated in the regularisation of different deep learning algorithms to assess if these

Table 2. Results obtained for the "U-Net: Region Mutual Information Loss" use-case.

Dataset	Intersection over Union (IoU)	Accuracy
Cityscapes Dataset	0.5244	0.9345
PASCAL VOC 2012	0.0607	0.6893

Table 3. Results obtained for the "Deep Mutual Learning: DeepLabV3 and U-Net" use-case.

Dataset	Intersection over Union (IoU)	Accuracy
Cityscapes Dataset	0.5619	0.9342
PASCAL VOC 2012	0.2203	0.7539

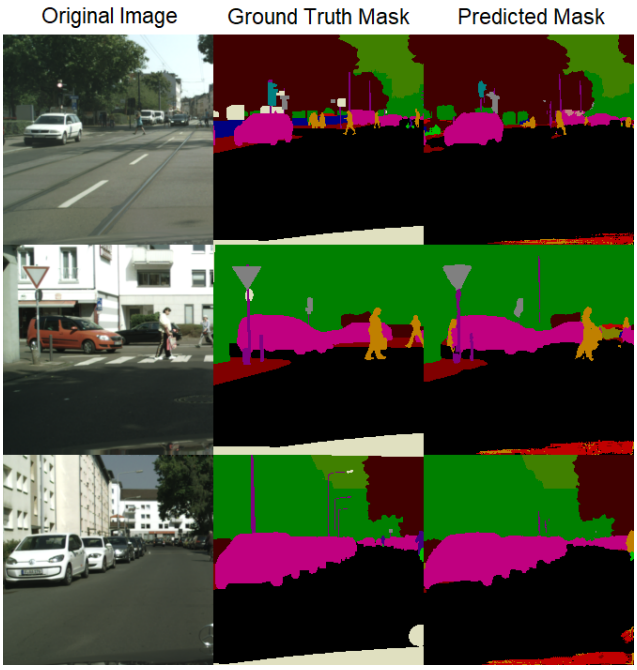

Figure 8. Examples of predicted masks for the Cityscapes Dataset using "Deep Mutual Learning: DeepLabV3 and U-Net".

Figure 9. Examples of predicted masks for the PASCAL VOC 2012 using "Deep Mutual Learning: DeepLabV3 and U-Net".

methodologies are data set dependent or task-dependent.

References

[1] V. Badrinarayanan, A. Kendall, and R. Cipolla. Segnet: A deep convolutional encoder-decoder architecture for image segmentation. *IEEE transactions on pattern analysis and machine intelligence*, 39(12):2481–2495, 2017. 2

[2] M. Buda, A. Saha, and M. A. Mazurowski. Association of genomic subtypes of lower-grade gliomas with shape features automatically extracted by a deep learning algorithm. *Computers in Biology and Medicine*, 109, 2019. 4

[3] L.-C. Chen, Y. Zhu, G. Papandreou, F. Schroff, and H. Adam. Encoder-decoder with atrous separable convolution for semantic image segmentation. In *Proceedings of the European conference on computer vision (ECCV)*, pages 801–818, 2018. 2

[4] M. Cordts, M. Omran, S. Ramos, T. Rehfeld, M. Enzweiler, R. Benenson, U. Franke, S. Roth, and B. Schiele. The cityscapes dataset for semantic urban scene understanding. In *Proc. of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2016. 4

[5] T. M. Cover. *Elements of information theory*. John Wiley & Sons, 1999. 2

[6] M. Everingham, L. Van Gool, C. K. I. Williams, J. Winn, and A. Zisserman. The PASCAL Visual Object Classes Challenge 2012 (VOC2012) Results. http://www.pascal-network.org/challenges/VOC/voc2012/workshop/index.html. 4

[7] S. Han, H. Mao, and W. J. Dally. Deep compression: Compressing deep neural networks with pruning, trained quantization and huffman coding. *arXiv preprint arXiv:1510.00149*, 2015. 3

[8] G. Hinton, O. Vinyals, and J. Dean. Distilling the knowledge in a neural network. *arXiv preprint arXiv:1503.02531*, 2015. 3

[9] D. P. Kingma and J. Ba. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*, 2014. 4

[10] H. Li, A. Kadav, I. Durdanovic, H. Samet, and H. P. Graf. Pruning filters for efficient convnets. *arXiv preprint*

arXiv:1608.08710, 2016. 3

[11] H. Li, Z. Xu, G. Taylor, C. Studer, and T. Goldstein. Visualizing the loss landscape of neural nets. *arXiv preprint arXiv:1712.09913*, 2017. 2

[12] T.-Y. Lin, P. Dollár, R. Girshick, K. He, B. Hariharan, and S. Belongie. Feature pyramid networks for object detection. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 2117–2125, 2017. 2

[13] W. Liu, A. Rabinovich, and A. C. Berg. Parsenet: Looking wider to see better. *arXiv preprint arXiv:1506.04579*, 2015. 2

[14] J. Long, E. Shelhamer, and T. Darrell. Fully convolutional networks for semantic segmentation. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 3431–3440, 2015. 1

[15] S. Minaee, Y. Boykov, F. Porikli, A. Plaza, N. Kehtarnavaz, and D. Terzopoulos. Image segmentation using deep learning: A survey. *CoRR*, abs/2001.05566, 2020. 1

[16] S. Minaee, Y. Y. Boykov, F. Porikli, A. J. Plaza, N. Kehtarnavaz, and D. Terzopoulos. Image segmentation using deep learning: A survey. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2021. 2

[17] A. Paszke, S. Gross, S. Chintala, G. Chanan, E. Yang, Z. DeVito, Z. Lin, A. Desmaison, L. Antiga, and A. Lerer. Automatic differentiation in pytorch. 2017. 4

[18] J. Peng, M. Pedersoli, and C. Desrosiers. Mutual information deep regularization for semi-supervised segmentation. In *Medical Imaging with Deep Learning*, pages 601–613. PMLR, 2020. 4

[19] M. Rastegari, V. Ordonez, J. Redmon, and A. Farhadi. Xnor-net: Imagenet classification using binary convolutional neural networks. In *European conference on computer vision*, pages 525–542. Springer, 2016. 3

[20] A. Romero, N. Ballas, S. E. Kahou, A. Chassang, C. Gatta, and Y. Bengio. Fitnets: Hints for thin deep nets. *arXiv preprint arXiv:1412.6550*, 2014. 3

[21] O. Ronneberger, P. Fischer, and T. Brox. U-net: Convolutional networks for biomedical image segmentation. In *International Conference on Medical image computing and computer-assisted intervention*, pages 234–241. Springer, 2015. 2

[22] K. Simonyan and A. Zisserman. Very deep convolutional networks for large-scale image recognition. In *International Conference on Learning Representations*, 2015. 2

[23] C. Szegedy, W. Liu, Y. Jia, P. Sermanet, S. Reed, D. Anguelov, D. Erhan, V. Vanhoucke, and A. Rabinovich. Going deeper with convolutions. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 1–9, 2015. 2

[24] M. Tschannen, J. Djolonga, P. K. Rubenstein, S. Gelly, and M. Lucic. On mutual information maximization for representation learning. *arXiv preprint arXiv:1907.13625*, 2019. 6

[25] Y. Zhang, T. Xiang, T. M. Hospedales, and H. Lu. Deep mutual learning. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 4320–4328, 2018. 1, 3, 4

[26] H. Zhao, J. Shi, X. Qi, X. Wang, and J. Jia. Pyramid scene parsing network. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 2881–2890, 2017. 2

[27] S. Zhao, Y. Wang, Z. Yang, and D. Cai. Region mutual information loss for semantic segmentation. *arXiv preprint arXiv:1910.12037*, 2019. 1, 3, 4