Incorporation Of Rejection Criterion - A Novel Technique For Evaluating Semantic Segmentation Systems

Elhassan Mohamed
School of Engineering and Digital Art
University of Kent
Canterbury, Kent, UK
enrm4@kent.ac.uk

Konstantinos Sirlantzis
School of Engineering and Digital Art
University of Kent
Canterbury, Kent, UK
k.sirlantzis@kent.ac.uk

Gareth Howells
School of Engineering and Digital Art
University of Kent
Canterbury, Kent, UK
w.g.j.howells@kent.ac.uk

***Abstract*—Semantic segmentation 'SS' evaluation metrics are great tools to assess systems' performance in terms of pixels' accuracy and the alignment of segments. Standard metrics ignore pixels' confidence scores which can carry useful information. Pixels' scores represent the level of confidence of the system for assigning class labels to image pixels. However, it has not been utilised by any evaluating metric for semantic segmentation systems. We propose to incorporate pixels' confidence scores with existing metrics to gain better insights into systems' behaviours. Results show the usefulness of the introduced approach to utilise the pixels' scores in the evaluation process. Besides, using pixels' scores thresholding can help to enhance the system performance on a specific task or objects of a particular size.**

***Index Terms*—Convolutional Neural Network, Evaluation metrics, Pixels classification, Semantic segmentation, Thresholding**

I. Introduction

Pixels classification is the process of assigning a label to each pixel in an image from a predefined set of labels. It is a deep learning technique to semantically segment, hence the Semantic Segmentation 'SS' name, an image into an annotated scene where each pixel has a label. A group of pixels with the same label represents an object. SS tasks are treated as supervised learning problems for which classifiers are trained to fit the training data on the pixels level.

SS systems have seen rapid progress in the past few years, not only from an accuracy perspective but also in speed and real-time processing. These systems have many applications such as autonomous driving [1], medical applications [2], and general scenes understanding [3]. Many Convolutional Neural Networks 'CNN' for SS tasks have been proposed. These systems follow two main categories: the series architecture such as Fully Convolutional Networks [4] and the encoder-decoder architecture such as U-Net [2], and DeepLab [5]. The efficiency of a SS system can be measured by its performance on a target application. However, these kinds of benchmarking are flawed because of the inability to compare algorithm with each other due to the subjectivity of the measure. This leads to the introduction of many other general application-independent metrics.

Many metrics are introduced to evaluate different deep learning tasks. Accuracy, Average Precision 'AP', Bounding Box Intersection over Union 'BB_IoU' and Mask Intersection over Union 'Mask_IoU' are mainly used to evaluate Object Classification 'OC', Object Detection 'OD', Semantic Segmentation and Instance Segmentation 'IS' tasks, respectively. Other metrics, such as Panoptic Quality 'PQ' [6], is introduced to unify the evaluation of semantic and Instance segmentation. PQ can be used to assess the performance of a system on both stuff and things classes (stuff classes such as sky, grass, ... etc., while things classes such as cars, people, ...etc.) in a simple and informative manner. Unlike Panoptic Segmentation 'PS' and SS, IS incorporate Object (segment) confidence score in the AP metric calculation. Confidence scores are essential elements in the evaluation of any system. These scores add a further informative dimension to the downstream systems. Consequently, utilising them in SS systems can help to better assess these systems.

This paper is focused on SS tasks as we want to incorporate the pixels' confidence scores in the evaluation of SS systems to gain more understanding of their behaviours. Results show that pixels' confidence scores can affect the systems performance evaluations dramatically, as shown in the results section. It can also provide a deep understanding of the system's operation. The main contributions of this paper are as follows: we propose a new technique that can be incorporated with the existing SS evaluation metrics. This technique is based on a well-known idea of thresholding that has been used in many applications over the past years. However, introducing this technique to evaluate SS tasks can be considered as a novel contribution. Thresholding of pixels' confidence scores can contribute to the SS overall output. Consequently, it has a major impact on the evaluation metrics. Pixels thresholding is distinct from Mask_IoU or BB_IoU as it is computed on the pixel level and not on the Mask or Bounding Box object level as in the case of IS or OD, respectively. We use a standard dataset in the comprehensive ablation experiments to analyse the contribution of the new element (pixels' threshold) on the system's output and the evaluation metrics. The used dataset

is the standard Cambridge-driving Labeled Video Database 'CamVid' [7] [8].

The paper is organised as follows: Section 2 covers the methodology and standard SS evaluation metrics. Section 3 presents the details of the experiment setup, and section 4 presents and discusses the obtained results.

II. Methodology

Semantic Segmentation evaluation metrics such as accuracy, 'IoU' and BF score do not incorporate pixels scores into their calculations. Pixels scores reflect the degree of confidence a pixel belongs to a specific class from a set of predefined classes. For semantic segmentation tasks, all pixels of an image have to be assigned to one of the predefined classes, even though these pixels might not belong to any of the classes. For example, if the predefined classes for a particular semantic segmentation system do not include a 'car' class, but an image that needs to be classified by the system contains a car, the system will assign the car's pixels to any predefined class. Usually, these pixels will have a very low score. Nevertheless, these pixels contribute to the system's performance as the traditional evaluation metrics uses them during the evaluation process.

On the other hand, Pixels of objects of non-interest are usually kept unlabelled in the ground truth 'gTruth' data (undefined or void pixels). While these pixels should not be used for the system evaluation, some traditional evaluation metrics cannot exclude them.

We propose a novel evaluation technique that incorporates pixels threshold in the evaluation process. The method is similar to posterior probability in statistics at which we assign all the 'undefined' pixels in the gTruth data to an extra class called 'Reject' class. Usually, these pixels belong to objects of non-interest for the system, object borders or oversight pixels. In case of CamVid dataset, these pixels might belong to far objects or pavement borders (Fig 1b). Fig 1c shows that these pixels have the lowest classification score.

To compare the predicted pixels with the gTruth ones, we assign all of the predicted pixels below a predefined threshold to the 'Reject class'. If the trained system is robust enough, these low score pixels should belong to objects of non-interest. Objects that are not in the predefined classes or have not been seen by the system. Then we evaluate the predicted output of the system with the gTruth data at different threshold values to investigate the system's behaviour and the threshold impact on the overall system performance.

A. Semantic segmentation evaluation metrics

The performance of semantic segmentation systems can be evaluated using the following metrics: Accuracy, IoU and Mean BF score. Each metric reflects a specific quality of the system, such as the ability of the system to classify pixels correctly or the alignment of the predicted pixels with the gTruth one. However, the aforementioned metrics do not consider pixels' confidence scores in their calculations. The softmax layer of a typical semantic segmentation system based on convolutional layers outputs several scores for each pixel in the image corresponding to the number of classes. The highest value represents the class of that pixel. In case of uncertainty, and sometimes border pixels, even the highest score pixel across all classes has a low value. Nevertheless, they still contribute to the system performance.

Two types of accuracy can be calculated for a dataset: Global Accuracy 'GA' and Mean Accuracy 'MA'. GA is calculated regardless of the class as the ratio between correctly classified pixels to the total number of pixels (1). Whereas MA is the average accuracy of all classes in all images. Accuracy of each class can be calculated as the ratio of correctly classified pixels to the total number of pixels in that class using (2). A major limitation of GA and MA measures is the bias in the presence of imbalanced classes.

$$GA = \frac{TP + TN}{TP + FP + FN + TN} \tag{1}$$

Where TP, TN, FP, FN are True Positive, True Negative, False Positive, and False Negative, respectively.

$$Acc_{class} = \frac{TP}{TP + FN} \tag{2}$$

Similarly, Mean IoU for a dataset can be calculated as the average IoU of all classes in all images. IoU (also known as Jaccard index) for each class is the ratio between correctly classified pixels to the total number of predicted and gTruth pixels in that class (3). For disproportionately distributed classes, Weighted IoU can be reported. It is the standard IoU but weighted by the number of pixels of each class in the dataset. IoU metrics evaluate the amount of correctly classified pixels but do not reflect the boundaries quality, which can consider as a disadvantage. Trimap [9] is introduced to overcome this drawback by evaluating the segmentation accuracy around the segment boundaries using a predefined narrow band around the contours. Whereas pixels in this predefined band contribute to the accuracy calculations. It suggests to measure pixels' accuracy within a defined region around the object boundaries rather than considering all image pixels to better assess the system's ability to capture objects' boundaries. Yet, choosing the optimal band size is challenging and might vary depending on the application.

$$IoU_{class} = \frac{TP}{TP + FP + FN} \tag{3}$$

Information retrieval [10] approaches have used Precision-Recall curves as a standard evaluation metric. The metric was first used to evaluate edge detectors by Abdou and Pratt [11]. Precision measures the ratio of detections that are True Positive rather than False Positive (6). Whereas Recall measure the ratio of the detected True Positive rather than missed (7). The parametric Precision-Recall curve captures the trade-off between accuracy and noise while the detector threshold changes [12]. A permissible trade-off for a particular application between noise and accuracy can be defined by the relative cost α in the F1 score equation (4).

(a) gTruth with annotation. (b) Undefined pixels (blue). (c) Pixels classification scores.

Fig. 1: Undefined pixels result in low pixels' confidence scores.

$$F1_{score} = \frac{P \cdot R}{(1-\alpha) \cdot P + \alpha \cdot R} \tag{4}$$

F1 score calculates the weighted harmonic mean of Precision and Recall. The maximum F1 score, which is the point on the curve where the optimal detector threshold occurs, can be reported as an indication of the detector's performance [12]. In our experiments, we set α to 0.5. Thus, (4) can be simplified to (5). Also using (6) and (7), (5) can be simplified to (8).

$$F1_{score} = \frac{2 \cdot P \cdot R}{P + R} \tag{5}$$

Where P and R are the Precision and Recall, respectively.

$$P = \frac{TP}{TP + FP} \tag{6}$$

$$R = \frac{TP}{TP + FN} \tag{7}$$

So,

$$F1_{score} = \frac{TP}{TP + \frac{1}{2}(FP + FN)} \tag{8}$$

Trimap [9] does not fully capture the quality of the contours. Thus, a semantic segmentation contour-based accuracy metric called BF score is proposed [13]. BF score metric is inspired by boundary-based evaluation measure [14] [15] and F1 score [12]. The boundary-based evaluation measure and F1 score define a distance tolerance to decide if a match has happened between pixel boundary points in the prediction and gTruth images.

Boundary-based measure [14] calculates the minimum euclidean distance between two sets of points where the sets represent the boundaries of two segments(gTruth and prediction). Hence, the mean and the standard deviation is calculated from the distance distribution between the two sets. A small mean and standard deviation indicate high matching. Whereas the weighted harmonic mean of Precision and Recall is used in the case of F1 score [12] to estimate the point for the optimal detector threshold.

BF score [13] has extended F1 score to semantic segmentation tasks. The proposed metric (BF score) has been used to calculate one value per class to evaluate classes independently. BF score sets the distance tolerance to 0.75% of the length of the image diagonal. We have used the same ratio in our experiments.

Mean BF score or contour matching score, which measure the alignment of the predicted and gTruth boundaries, is the average BF score of all classes in all images for a dataset (10). Whereas Mean BF score of a class is the average F1 score (8) of that class overall images (9).

$$MeanBF_{\text{score}}^{\text{class}} = \frac{\sum F1_{\text{score}}^{\text{class}}}{\text{no\# of images}} \tag{9}$$

$$MeanBF_{\text{score}}^{\text{dataset}} = \frac{\sum_{classes} BF_{\text{score}}}{\text{no\# of images}} \tag{10}$$

IoU is the standard evaluation metric for PASCAL VOC challenge [16]. However, solely depending on a specific metric to assess a SS system is insufficient. Csurka et al. [13] argue that systems' parameters for a segmentation algorithm should be optimized one the target metric for fair comparisons as different segmentation algorithms can be optimal for different evaluation metrics. Besides, per-image metrics can provide more details of the system's performance and allow more detailed comparisons. Hence, Mean BF score is averaged over all images in a dataset. Additionally, in our experiments, we have reported accuracy and IoU for the dataset and for each class. We believe that these metrics are complementary to each other. Furthermore, incorporating pixels' confidence scores with these metrics can reveal another level of information of the system's performance.

B. Relation of the proposed thresholding technique to the existing metrics

The calculations of conventional SS evaluation metrics ignore pixels' confidence scores. We propose to incorporate pixel confidence scores with the calculation process of these metrics because of the important information that can be reflected by

these scores. First, we predefine a pixel score threshold. If the pixel's value after the softmax layer (the highest pixel value across all pixel classes) is below this threshold, its value is assigned to a 'Reject' class. Class 'Reject' cannot contribute to any of the evaluation metrics calculations. For a robust system, a low confidence pixel score usually represents a high uncertainty pixel. This pixel can be for a class of non-interest (i.e., has not been predefined for the task) or a pixel between the borders of predefined classes of interest. Lastly, we evaluate the system using the existing metrics.

After thresholding, predicted pixels corresponding to gTruth ones count towards TP, predicted pixels different from gTruth ones count towards FP, and unpredicted gTruth pixels count towards False Negative FN.

The novelty of the proposed technique is in assessing the system under different conditions (pixels' threshold values). SS systems under various conditions can behave differently. Consequently, the system behaviour should be well-investigated using many impacting factors. The most important impacting factor is the pixel itself. Thus, we believe the pixels' scores should contribute to the evaluation process of any system.

The segment matching for the IoU denoted by $\mathrm{IoU}(p, g)$ in (11) for the Panoptic Quality metric is different from the Semantic Segmentation IoU as the former is calculated between two segments (p and g for the predicted and gTruth segments, respectively). Whereas the latter is calculated based on the output pixel labels and completely ignore object-level labels. Also, the segment matching threshold of 0.5 used by PQ is distinct from the proposed pixel confidence thresholding. The former is computed on the segment matching IoU level (object level). In contrast, the latter is computed on the pixel confidence score level.

$$\mathrm{PQ} = \frac{\sum_{(p,g)\in TP} \mathrm{IoU}(p, g)}{|TP| + \frac{1}{2}|FP| + \frac{1}{2}|FN|} \tag{11}$$

The proposed process is simple and straight forward (Fig. 2). However, the added dimension of the pixels' confidence scores helps to exclude the contribution of a specific area in an image with respect to the overall performance of the system. As this area might be undefined by the annotator, yet, it still contributes to the metrics calculations, which is undesirable in many cases.

III. Experimental setup

In our experiments, we have reported GA, MA, Mean IoU, Weighted IoU and Mean BF score using four different pixels' threshold values (0.2, 0.4, 0.6, 0.8) that are monotonically increasing. The choice of these threshold values helps to capture the system's behaviour under a wide range of conditions. The evaluation metrics are calculated for the dataset and the individual classes.

A. Dataset

Cambridge-driving Labeled Video Database 'CamVid' [7] [8] is used to test the proposed evaluation technique. CamVid dataset has 701 images annotated on the pixel level for 32 classes. Images are captured outdoors from the perspective of a driving car. We group the 32 classes of the dataset into 11 classes for simplicity as some of the 32 original classes have very limited objects. These 11 classes are Building, Pole, Road, Pavement, Tree, Sign/Symbol, Car, Pedestrian, Bicyclist, Sky, and Fence.

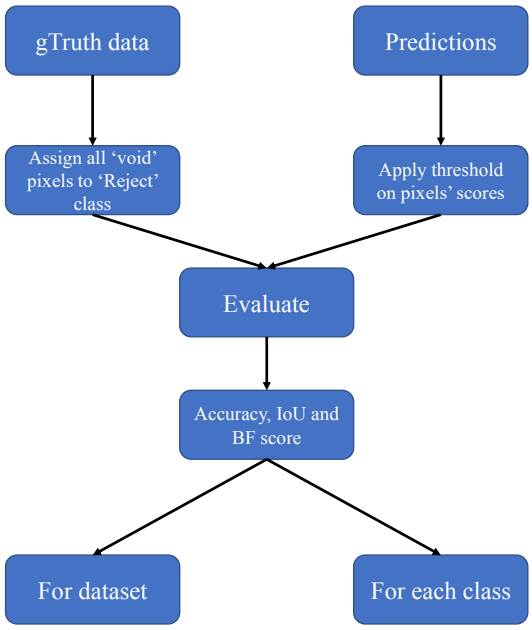

Fig. 2: Methodology

The dataset contains some undefined/void pixels which belong to non-of-interest objects or overlooked pixels. We also define an extra 'Reject' class for the purpose of our experiments for which we assign all the undefined pixels. Pixels distribution of the gTruth data is shown in Fig. 3 (gTruth).

The dataset is split randomly into 70% for training (491), 15% for validation (105 images) and 15% for testing(105 images).

B. System architecture

The neural network architecture that has been used for training is based on encoder-decoder DeepLab Version 3 plus 'DLV3+' [5] for semantic segmentation. The architecture's base network uses residual blocks that help the system to process high-resolution images (960×720×3 pixels) without losing information because of vanishing gradients. In addition, the system's decoder has a simple design but with high efficiency.

Very deep networks suffer from vanishing/exploding gradients [17] [18]. Residual blocks help to mitigate this problem by reusing the activations from previous layers until the adjacent layer learns its weights. This allows the network to learn more low-level features without being worried about performance degradation as it goes deeper. The elegance of this architecture

is that these short-cut connections do not add either extra parameters or computational complexity [19].

C. Training

The system is trained end-to-end using Stochastic Gradient Descent with 0.9 Momentum 'SGDM' as the training optimiser. A starting learning rate of 0.001 which is then dropped by a factor of 0.3 every ten epochs. To avoid sequence memorisation, training images are shuffled every epoch. Also, L2 regularisation is used to limit overfitting. To enhance the overall system accuracy, data augmentation is employed with X and Y translations. Additionally, different hyper-parameters and optimisation algorithms are tried to achieve the highest performance. Moreover, for reproducibility, systems are trained several times under the same configurations.

To avoid bias in favour of dominant classes, inverse frequency weighting is used to balance classes weightings. Image normalisation is employed to rescale all the pixels' values in the range of zero to one. The system is trained on relatively high-resolution images of 960×720×3 pixels, unlike the original implementation of DLV3+, which crops patches of 513 size from the PASCAL VOC dataset [16] images during the training and the testing processes. This approach, training on high-resolution images, is believed to enhance the system's ability to semantically segment small size objects alongside medium and large size ones. Also, this boosts the effectiveness of large rate atrous convolutions kernels as its weight will be applied to actual pixels and not to zero paddings.

IV. Results and discussion

A. CamVid dataset results

The trained system on the CamVid datasets has achieved a validation loss of 0.368. The distance tolerances of 0.75% of the length of the image diagonal for the BF score calculations is 9 pixels. Results show interesting behaviours of the evaluation metrics regarding different size objects using various threshold values. Thresholding has proved that the pixels' confidence scores greatly impact the system's performance concerning the datasets and the individual classes. Consequently, the proposed technique can be used to optimize the system on a specific application, task, or group of objects of interest.

Table I shows that the system's performance on the CamVid dataset varies under different thresholds. While applying no threshold, it has achieved the highest MA and Weighted IoU. Whereas higher threshold values have achieved better GA, Mean IoU and Mean BF score. Consequently, it is feasible to optimize the system on a specific evaluation metric for a specific application or challenge.

Similar observations can be extracted from Table II. Although applying no threshold has achieved the highest accuracy for all classes regardless of the object's size or pixels' distribution, higher threshold values have achieved better IoU and mean BF score.

Large size objects, and therefore high pixels' distribution (Fig. 3) such as Sky, Building, and Road, have achieved the best performance under no pixels' scores threshold. Large-medium and medium-sized objects, such as Tree and Pavement, have achieved better IoU and Mean BF score using moderate pixels' scores thresholding of 0.4 and 0.6. For medium-small and small size objects, which have the lowest pixels' distributions but vital to many applications such as Pole, SignSymbol, Fence, Car, Pedestrian, and Bicyclist, applying higher threshold values have achieved the best performance in terms of higher IoU and Mean BF scores.

Objects' sizes and pixels' frequencies have a great impact on the system's behaviour, consequently, a direct impact on the evaluation process. As an example, when the number of pixels that are assigned to the 'Reject' class increase due to the increase in the threshold values, IoU and mean BF score of things classes, that are mainly of medium and small sizes, increase (Fig. 3 and Table II).

On the other hand, large size and high pixels frequency classes (stuff classes) have performed best using low or no pixels' score threshold values. Thus, optimizing the network on a specific class or group of classes for a particular task is straight forward thanks to the thresholding technique.

Remarkable results are shown in Fig. 4 which depict the per-image IoU at different threshold values. The number of images that have achieved an overall IoU of more than 0.5 increases as we increase the threshold values. Consequently, pixels' threshold values can directly impact the classifier performance, which in this case can indicate the enhancement of the system performance on the IoU metric. The performance boost can be attributed to the high uncertainty of the undefined pixels and pixels at the object's borders that can be elevated using the appropriate pixels' threshold values to reduce the impact of fuzzy pixels quantitatively and qualitatively.

V. Conclusion

Pixels are the main building blocks of any image. Thus, we believe their confidence scores should contribute to the evaluation metrics. Nevertheless, pixels' scores have been overlooked by standard evaluation metrics. We have presented a novel technique that incorporates pixels' confidence scores in the evaluation process of semantic segmentation systems, which can add a further dimension to the evaluation metrics. The proposed technique is straight forward and has been applied to many statistical problems, which signifies its efficiency.

Results have shown the potential of the thresholding technique as it helps to suppress fuzzy pixels that do not belong to any classes of interest and emerge pixels that belong to classes of importance to the application. Furthermore, it can be concluded from the results that optimizing systems on large size objects (stuff classes) can be achieved using no or low pixels' threshold values. Whereas systems' performances on medium, small and tiny objects can be boosted using high pixels threshold values.

Acknowledgment

This work is supported by the Assistive Devices for empowering dis-Abled People through robotic Technologies

TABLE I: Evaluation Metrics Of The CamVid Test Set At Different Thresholds Values.

Metrics / Threshold	*Global Accuracy*	*Mean Accuracy*	*Mean IoU*	*Weighted IoU*	*Mean BFScore*
No threshold	0.895	**0.864**	0.667	**0.831**	0.690
0.2	0.895	0.864	0.667	0.831	0.690
0.4	0.900	0.858	0.672	0.831	0.697
0.6	0.929	0.815	**0.682**	0.817	**0.699**
0.8	**0.960**	0.731	0.663	0.772	0.660

TABLE II: Class Metrics Of The CamVid Test Set At Different Threshold Values.

Threshold Values	**No Threshold**			**0.2**			**0.4**			**0.6**			**0.8**		
Metrics / Class	***Acc***	***IoU***	***M.BF***	***Acc***	***IoU***	***M.BF***	***Acc***	***IoU***	***M.BF***	***Acc***	***IoU***	***M.BF***	***Acc***	***IoU***	***M.BF***
Sky	**0.940**	**0.908**	**0.905**	0.940	0.908	0.905	0.939	0.908	0.905	0.920	0.901	0.892	0.880	0.872	0.831
Building	**0.816**	**0.796**	**0.633**	0.816	0.796	0.633	0.809	0.791	0.615	0.753	0.742	0.518	0.654	0.650	0.385
Pole	**0.731**	0.240	0.578	0.731	0.240	0.578	0.721	0.249	0.588	0.635	0.294	0.630	0.476	**0.322**	**0.645**
Road	**0.941**	**0.928**	**0.817**	0.941	0.928	0.817	0.940	0.927	0.816	0.928	0.919	0.784	0.896	0.892	0.706
Pavement	**0.903**	0.741	0.750	0.903	0.741	0.750	0.899	0.742	**0.751**	0.865	**0.746**	0.749	0.782	0.718	0.696
Tree	**0.904**	0.780	0.722	0.904	0.780	0.722	0.901	**0.783**	**0.726**	0.859	0.778	0.707	0.760	0.723	0.598
SignSymbol	**0.766**	0.456	0.543	0.766	0.456	0.543	0.757	0.463	0.555	0.698	**0.496**	0.597	0.592	0.495	**0.633**
Fence	**0.806**	0.571	0.564	0.806	0.571	0.564	0.798	0.584	0.584	0.737	**0.605**	0.600	0.639	0.583	**0.608**
Car	**0.925**	0.804	0.760	0.925	0.804	0.760	0.919	0.808	0.767	0.888	**0.811**	**0.768**	0.829	0.788	0.725
Pedestrian	**0.859**	0.457	0.625	0.859	0.457	0.625	0.849	0.474	0.649	0.794	0.518	0.716	0.693	**0.533**	**0.719**
Bicyclist	**0.915**	0.656	0.555	0.915	0.656	0.555	0.909	0.665	0.598	0.885	0.695	0.669	0.834	**0.715**	**0.787**

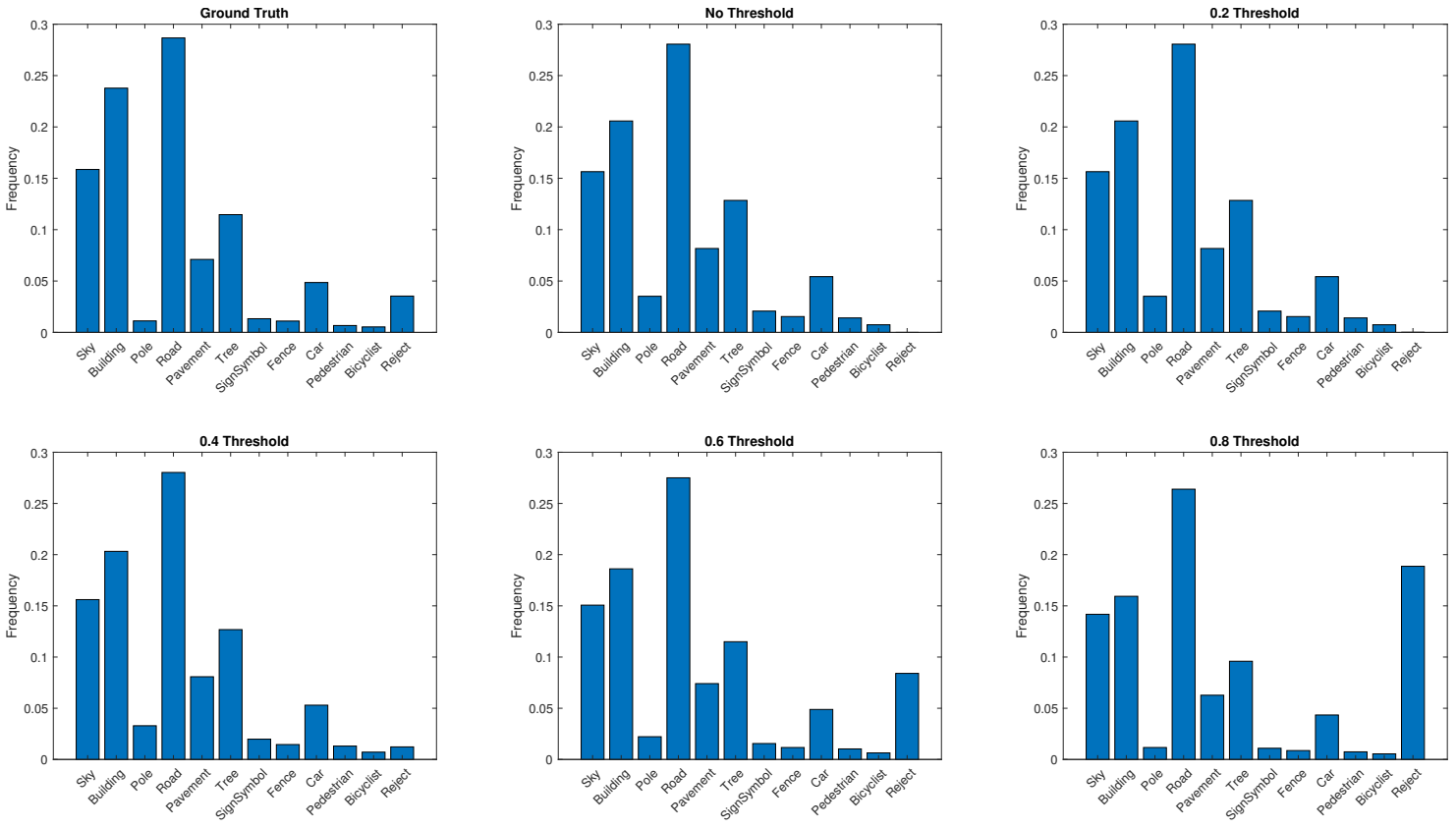

Fig. 3: Pixels distribution at different threshold values for the CamVid test set.

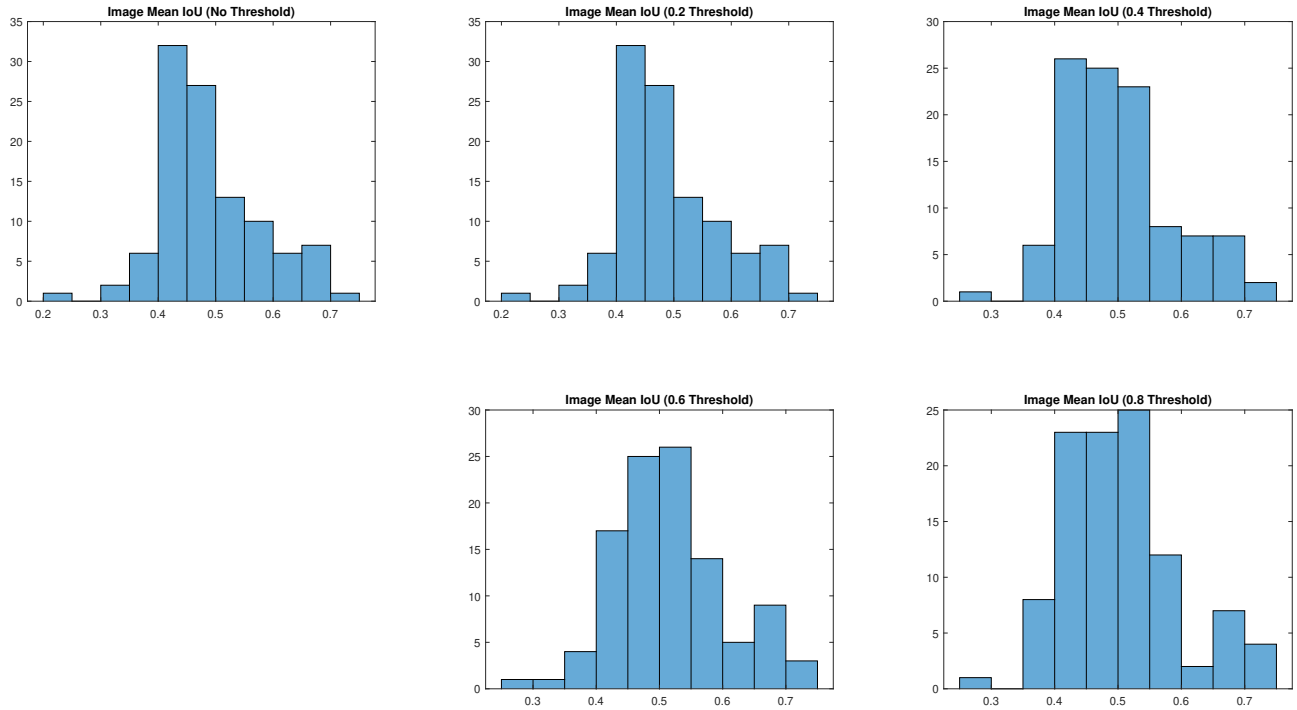

Fig. 4: Histogram of the per-image IoU for CamVid test set at different threshold values. The x-axis represents the IoU value and the y-axis represents the number of images.

(ADAPT) project. ADAPT is selected for funding by the INTERREG VA France (Channel) England Programme which is co-financed by the European Regional Development Fund (ERDF). The European Regional Development Fund (ERDF) is one of the main financial instruments of the European Unions (EU) cohesion policy.

REFERENCES

[1] R. Miyamoto, Y. Nakamura, M. Adachi, T. Nakajima, H. Ishida, K. Kojima, R. Aoki, T. Oki, and S. Kobayashi, "Vision-based road-following using results of semantic segmentation for autonomous navigation," in *2019 IEEE 9th International Conference on Consumer Electronics (ICCE-Berlin)*. IEEE, 2019, pp. 174–179.

[2] 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*. Springer, 2015, pp. 234–241.

[3] L.-J. Li, R. Socher, and L. Fei-Fei, "Towards total scene understanding: Classification, annotation and segmentation in an automatic framework," in *2009 IEEE Conference on Computer Vision and Pattern Recognition*. IEEE, 2009, pp. 2036–2043.

[4] E. Shelhamer, J. Long, and T. Darrell, "Fully convolutional networks for semantic segmentation," *IEEE transactions on pattern analysis and machine intelligence*, vol. 39, no. 4, pp. 640–651, 2017.

[5] 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)*, 2018, pp. 801–818.

[6] A. Kirillov, K. He, R. Girshick, C. Rother, and P. Dollár, "Panoptic segmentation," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2019, pp. 9404–9413.

[7] G. J. Brostow, J. Shotton, J. Fauqueur, and R. Cipolla, "Segmentation and recognition using structure from motion point clouds," in *ECCV (1)*, 2008, pp. 44–57.

[8] G. J. Brostow, J. Fauqueur, and R. Cipolla, "Semantic object classes in video: A high-definition ground truth database," *Pattern Recognition Letters*, vol. xx, no. x, pp. xx–xx, 2008.

[9] P. Kohli, P. H. Torr *et al.*, "Robust higher order potentials for enforcing label consistency," *International Journal of Computer Vision*, vol. 82, no. 3, pp. 302–324, 2009.

[10] C. J. V. Rijsbergen, "Information retrieval," in *Encyclopedia of GIS*, 1979.

[11] I. E. Abdou and W. K. Pratt, "Quantitative design and evaluation of enhancement/thresholding edge detectors," *Proceedings of the IEEE*, vol. 67, no. 5, pp. 753–763, 1979.

[12] D. R. Martin, C. C. Fowlkes, and J. Malik, "Learning to detect natural image boundaries using local brightness, color, and texture cues," *IEEE transactions on pattern analysis and machine intelligence*, vol. 26, no. 5, pp. 530–549, 2004.

[13] G. Csurka, D. Larlus, F. Perronnin, and F. Meylan, "What is a good evaluation measure for semantic segmentation?." in *BMVC*, vol. 27, no. 2013, 2013, pp. 10–5244.

[14] Q. Huang and B. Dom, "Quantitative methods of evaluating image segmentation," in *Proceedings., international conference on image processing*, vol. 3. IEEE, 1995, pp. 53–56.

[15] J. Freixenet, X. Muñoz, D. Raba, J. Martı, and X. Cufí, "Yet another survey on image segmentation: Region and boundary information integration," in *ECCV*, 2002.

[16] M. Everingham, L. Van Gool, C. K. Williams, J. Winn, and A. Zisserman, "The pascal visual object classes (voc) challenge," *International journal of computer vision*, vol. 88, no. 2, pp. 303–338, 2010.

[17] Y. Bengio, P. Simard, and P. Frasconi, "Learning long-term dependencies with gradient descent is difficult," *IEEE Transactions on Neural Networks*, vol. 5, no. 2, pp. 157–166, 1994.

[18] X. Glorot and Y. Bengio, "Understanding the difficulty of training deep feedforward neural networks," in *Proceedings of the Thirteenth International Conference on Artificial Intelligence and Statistics*, ser. Proceedings of Machine Learning Research, Y. W. Teh and M. Titterington, Eds., vol. 9. Chia Laguna Resort, Sardinia, Italy: PMLR, 13–15 May 2010, pp. 249–256. [Online]. Available: http://proceedings.mlr.press/v9/glorot10a.html

[19] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2016, pp. 770–778.