Increasing robustness of semantic segmentation by highlighting edges in training data

Hidde Huitema
1373005
Department of Electrical Engineering
Eindhoven University of Technology
Eindhoven, The Netherlands
h.r.huitema@student.tue.nl

***Abstract*—Convolutional Neural Networks have been shown to be very sensitive to noise and other image corruptions. Previous work has succesfully increased the robustness of CNN's to these kinds of noise by forcing them to recognise shapes instead of textures. This work proposes a simple data augmentation method that no longer relies on labelled target images to achieve this, but rather highlights the edges that are present in the image using a simple edge detector. This method shows a 4.5% improved dice score with respect to the baseline and a 2.5% improvement with respect to previous work.**

***Index Terms*—CNN, semantic segmentation, Robustness**

I. Introduction

Previous research has shown that Convolutional Neural Networks (CNN's) experience performance decreases when they are faced with corrupted input images [1]. It has even been shown that Neural Networks perform significantly worse than humans when faced with some image distortions [2]. In real world computer vision applications, these kinds of corruptions are unavoidable and it is therefore important to come up with ways to improve the robustness of CNN's to ensure acceptable performance in real world scenarios. A logical approach would be to train the models on noisy images, but this might lead to a performance decrease on high quality images and will lead to longer training times, since the models need to be trained on more data.

A significant amount of work has already been done to increase the robustness of CNN's by performing data augmentations. Hendrycks et al. [3] show that by combining different data augmentations (such as rotation, shift or cutouts), the robustness of a model can be significantly improved. Geirhos et al. [4] have shown that CNN's mostly learn to recognise texture of objects and that by forcing it to recognise shapes, the robustness of image classification is increased. Previous work has also shown that the robustness of a segmentation model can be increased by forcing the model to focus on shapes instead of textures during training [5]. This last work colours images based on the labels given in the targets (see Figure 1). This approach led to significant improvements in robustness, but heavily relies on accurate annotation of images and can not be used in self-supervised learning methods. The current approach aims to tackle these two problems and increase the shape bias of the model by highlighting the edges that are present on the image using a simple edge detector.

II. Methods

A. Baseline model and training strategy

The baseline model is a U-Net architecture with 6 encoder and decoder blocks. This model was implemented in Pytorch [6] and trained on the CityScapes dataset [7]. The images in this dataset are down scaled in each dimension by a factor 4 (from 1024x2048 pixels to 256x512 pixels) to allow for faster training. The model was trained with the Adam optimiser with an exponentially decaying learning rate. The Crossentropy loss function is used and the relative weights of each class are determined based on the number occurrences of each class. The most frequently occurring class has unit weight and the least occuring class has a weight of 1.5, this is done to deal with the significant class imbalance in the CityScapes dataset. The model is trained for 50 epochs until it is converged. A train:validation split of $0.9 : 0.1$ was used. All other models will use the exact same training strategy to make the results consistent. The code for this project is made available in a public Github [1] repository.

B. 'Paint by number' approach

The current proposed method that will be described below is very similar to the one used by Kamann and Rother [5], but does not rely on the labelled images. To evaluate the performance of the current approach, this method of 'painting by number' will also implemented and the results will be compared between the baseline model, the model trained on edge information and the model trained using the 'paint by number approach'. An example image of the latter approach can be seen in Figure 1. This 'paint by number' approach will be used as a second baseline.

C. Improvements

To improve the robustness of the model to image corruptions, the model will be forced to focus on learning shapes during the training process. To achieve this, an edge detector is applied to the image and these edges will be blended into the

[1] https://github.com/HiddeHuitema/5SLM0_Homework

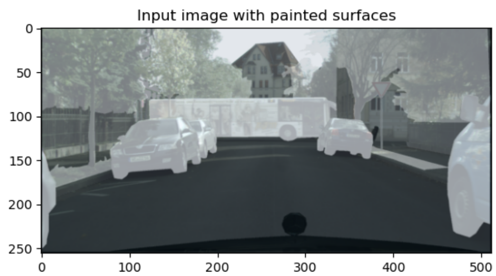

Fig. 1. Image with painted surfaces

original image, leading to an image with highlighted edges, as can be seen in Figure 2. In doing so, we assume that the edges in an image strongly correspond to the shapes of objects in this image. The edge information is obtained by calculating the pixel gradients in x and y direction for the three colour channels and taking their mean. This image with edge information is then blended into the original image according to:

$$I_{blend} = \alpha * I_{original} + (1 - \alpha) * I_{edges} \quad (1)$$

$I_{original}$ and I_{edges} refer to the original image and the image with edge information respectively. During training the value of α is randomly sampled for each batch of images with $0 \leq \alpha \leq 1$. In each batch, exactly half of the images are augmented with this method, the others are not augmented this way to ensure that the model performs well even on normal images without highlighted edges.

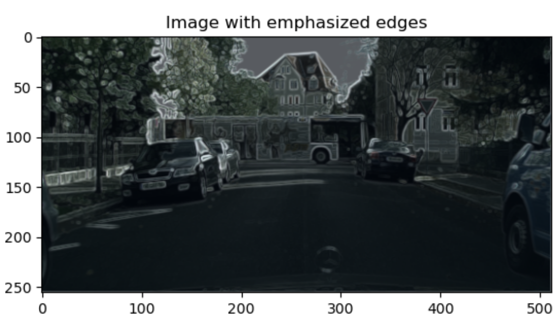

Fig. 2. Image with highlighted edges

III. Results

A. Baseline Results

To evaluate the performance of the model quantitatively, it is uploaded to a CodaLab competition. The results of the baseline model in the 'Peak Performance' and 'Robustness' categories can be seen in Table I. These results show that the baseline model performs poorly in segmenting smaller objects in the 'object' and 'human' categories. This is most likely due to the down scaling of the images, where the information of small objects is largely lost. Figure 3 shows the segmentation performance of the model on an image from the validation set. This result is in line with the dice scores from Table I; the model segments the road, sidewalk and buildings well, but it misses some finer details such as traffic lights. Figure 4 shows the performance of this baseline model with some Gaussian noise added to the input image. This nearly in perceivable amount of noise significantly impacts performance, where the image is not segmented properly anymore.

B. Paint by number approach

Figure 5 shows the model performance with the same amount of noise added as in Figure 4. This shows that the segmentation performance is still impacted significantly by a small amount of noise. Table I shows that this model performs slightly better than the baseline model in the peak performance category, with a slightly better dice score for almost all classes. A similar increase in performance can be seen in the robustness category, where the biggest improvements can be seen in the flat and nature categories.

C. Highlighted edges

Table I shows that the model trained on images with highlighted edges slightly outperforms the baseline model (an improvement of about 0.5%) in the Peak Performance benchmark. Figure 6 shows the segmentation results for this model. The segmentation is very similar to the one of the baseline model and this aligns with the scores from Table I. It scores better in all categories except for the vehicles. This is likely due to the fact that vehicles contain a lot of different edges, such as from windows and doors. This leads to an excessive amount of edge information for this category which has likely negatively impacted the performance.

This model still suffers the some effects of noise similar to the baseline model and the 'paint by number' trained model. However, this new augmentation strategy allows us to highlight the edges in the test images with the same strategy that was used during training, which significantly improves results, as can be seen in Figure 7. This immediately demonstrates one benefit of using this novel approach that does not depend on the target images and can therefore also be applied on line to improve performance in difficult circumstances.

For quantitative results, this model was uploaded to the robustness category of the CodaLab competition without the augmentations mentioned above. Without applying these augmentations on the test data, the model shows an increase in performance of about 2.5% compared to the 'paint by number' approach and a 4.5% increase in performance compared to the baseline model. The biggest improvements come from the flat and sky categories.

IV. Conclusion

In this work, a new data augmentation scheme was proposed to increase the robustness of segmentation models. This approach aims to force a CNN to learn the shape of objects instead of their texture, which essential for real life situations where noise makes textures unreliable. By highlighting the edges in an image (through calculating pixel gradients and blending them into the original image),

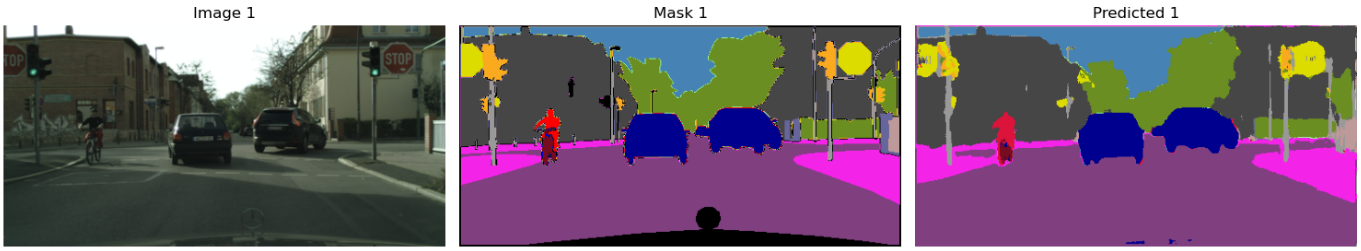

Fig. 3. Segmentation results for the baseline model

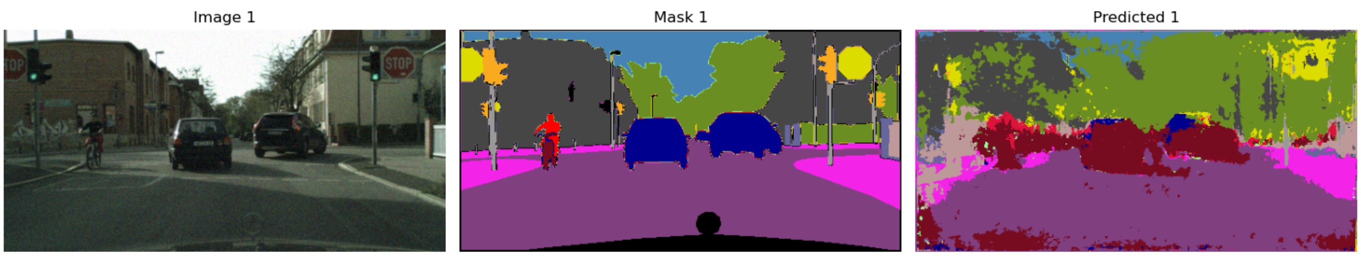

Fig. 4. Segmentation results for the baseline model with added noise

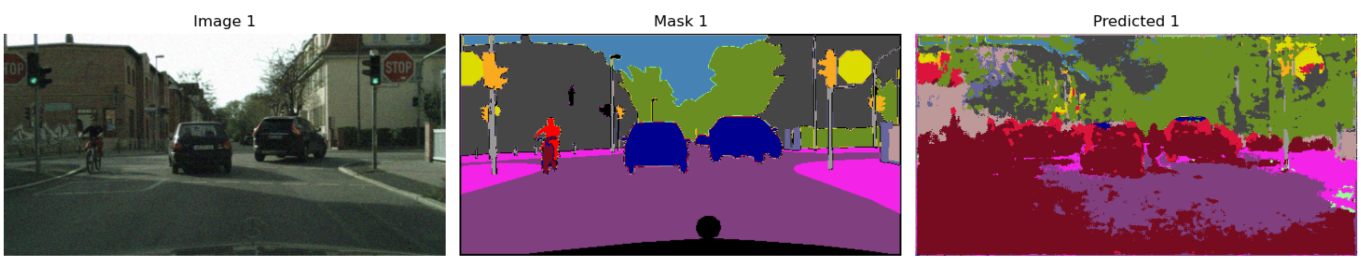

Fig. 5. Segmentation results for the model trained with ‘paint by number’ augmentations on a noisy image

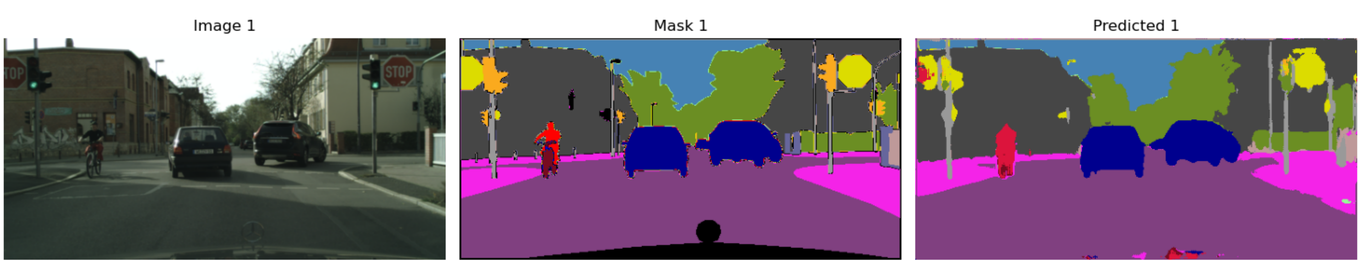

Fig. 6. Segmentation results for the model trained with highlighted edges

TABLE I
SCORES FOR MODELS TRAINED WITH DIFFERENT AUGMENTATIONS

Model	Category	Mean Dice	Dice flat	Dice construction	Dice object	Dice Nature	Dice sky	Dice human	Dice vehicle
Baseline	Peak Performance	0.420	0.87	0.77	0.12	0.80	0.78	0.23	0.68
	Robustness	0.312	0.72	0.65	0.09	0.57	0.59	0.17	0.43
Paint By Number	Peak Performance	0.423	0.87	0.78	0.13	0.81	0.80	0.28	0.64
	Robustness	0.317	0.77	0.65	0.10	0.61	0.60	0.17	0.44
Highlighted edges	Peak Performance	0.422	0.89	0.78	0.14	0.82	0.81	0.26	0.62
	Robustness	0.326	0.80	0.66	0.09	0.62	0.65	0.16	0.41

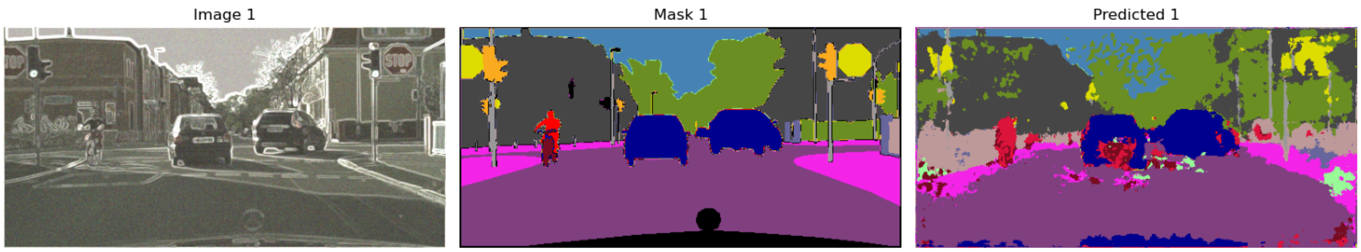

Fig. 7. Segmentation results for the model trained with highlighted edges augmentations

the robustness of the model was improved with respect to a baseline model as well as a similar data augmentation technique from literature. Another benefit of the current data augmentation strategy comes from the fact that we do not rely on the annotated target images to perform the augmentations, which allows this strategy to be implemented in self-supervised learning methods and makes it possible to apply the augmentations on-line in real world segmentation tasks to increase performance.

A. Limitations

One of the main limitations of this research is the performance of the baseline model. This baseline model did not have excellent segmentation results, which is likely the result of having to down scale the input images to allow for faster training. Since this baseline model does not have state of the art performance, it is hard to estimate how much the proposed method would increase the robustness of more state of the art models.
Another limitation is the fact that only one range for α was used, varying this range would allow us to try different strengths of augmentations and find an optimal range that provides the most robust model.

B. Future work

Future work can look into the limitations mentioned before by applying the data augmentation technique to train a state of the art model and training it with different ranges for α. Another interesting direction would be to apply it to a self-supervised learning method since the ability to apply this data augmentation strategy to self supervised approaches is one of its greatest benefits over previous work.

REFERENCES

[1] S. Dodge and L. Karam, "Understanding How Image Quality Affects Deep Neural Networks," 4 2016. [Online]. Available: http://arxiv.org/abs/1604.04004
[2] ——, "A Study and Comparison of Human and Deep Learning Recognition Performance Under Visual Distortions," 5 2017. [Online]. Available: http://arxiv.org/abs/1705.02498
[3] D. Hendrycks, N. Mu, E. D. Cubuk, B. Zoph, J. Gilmer, and B. Lakshminarayanan, "AugMix: A Simple Data Processing Method to Improve Robustness and Uncertainty," 12 2019. [Online]. Available: http://arxiv.org/abs/1912.02781
[4] R. Geirhos, P. Rubisch, C. Michaelis, M. Bethge, F. A. Wichmann, and W. Brendel, "IMAGENET-TRAINED CNNS ARE BIASED TOWARDS TEXTURE; INCREASING SHAPE BIAS IMPROVES ACCURACY AND ROBUSTNESS," Tech. Rep. [Online]. Available: https://github.com/rgeirhos/texture-vs-shape
[5] C. Kamann and C. Rother, "Increasing the Robustness of Semantic Segmentation Models with Painting-by-Numbers," in *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, vol. 12355 LNCS. Springer Science and Business Media Deutschland GmbH, 2020, pp. 369–387.
[6] J. Ansel, M. E. Yang, M. H. He, M. Natalia, G. Openai, A. Jain, M. M. Voznesensky, M. Bin, B. M. Peter, B. Quansight, D. Berard, M. Evgeni, B. Quansight, G. Chauhan, M. Anjali, C. Meta, M. Alban, D. Meta, Z. Devito, M. E. Ellison, M. W. Feng, M. J. Gong, I. Michael, G. Meta, B. Hirsh, M. S. Huang, M. Kshiteej, K. Quansight, L. Kirsch, M. M. Lazos, M. Mario, L. Quansight, Y. Liang, M. J. Liang, M. Yinghai, L. Meta, C. K. Luk, M. B. Maher, M. Y. Pan, C. Puhrsch, M. Matthias, R. Meta, M. Marcos, Y. Siraichi, Q. Helen, S. Meta, M. Suo, M. Phil, T. Openai, E. Wang, I. X. Wang, M. William, W. Meta, S. Zhang, M. Xu, Z. Meta, K. Zhou, R. Zou, M. A. Mathews, M. G. Chanan, M. P. Wu, M. Soumith, and C. Meta, "PyTorch 2: Faster Machine Learning Through Dynamic Python Bytecode Transformation and Graph Compilation Mark Saroufim," 2024. [Online]. Available: https://doi.org/10.1145/3620665.3640366
[7] M. Cordts, M. Omran, S. Ramos, T. Rehfeld, M. Enzweiler, R. Benenson, U. Franke, S. Roth, B. Schiele, D. A. R&d, and T. U. Darmstadt, "The Cityscapes Dataset for Semantic Urban Scene Understanding," Tech. Rep. [Online]. Available: www.cityscapes-dataset.net