Effect of Image Quality in Computer Vision for Semantic Segmentation of Road Images

Cristian Anghel
University of Twente
P.O. Box 217, 7500AE Enschede
The Netherlands
c.anghel@student.utwente.nl

ABSTRACT

Convolutional neural networks have real practical application potential, such as autonomous driving, but they are known to be sensitive to image degradation. The focus of this research is to give insight into the robustness of the current state-of-the-art model for semantic segmentation against corruptions likely to be encountered in real settings, specifically compression, motion blur and Poisson (shot) noise. In a safety-critical application, the precise semantic segmentation of certain instance classes, for example persons or vehicles, can be considered more important than others, such as vegetation or the sky, which is why the robustness of individual instance classes is also assessed with the intent to determine model deployability.

Keywords

Semantic segmentation, computer vision, convolutional neural network, road images, degradation, image quality

1. INTRODUCTION

Autonomous vehicles are heavily reliant on camera systems which help determine the surroundings of a vehicle. Cameras are a very common and practical way of capturing visual representations of the world, thus they are the predominant tool used in autonomous vehicles. Capturing images in traffic is a necessity for autonomous driving, but simply capturing images is not enough as a computer system would not make sense of plain images. One way of enabling any computer to make sense of the surroundings is by inputting these images into a convolutional neural network (CNN). The output generated by the CNN can then be used by an embedded system to make decisions. However, a CNN's performance has been shown to be sensitive to image degradation of any kind [6, 1, 11, 19], which is precisely what can happen in real-life scenarios. This sensitivity to image degradation of CNNs has been studied for some applications of computer vision, specifically for face detection [17, 21] — although these studies focus on object detection models — and weed mapping [13]. The robustness of semantic segmentation models has also been studied in [14], although this study analyses the robustness to extreme cases of degradation. There is no research, however, on the robustness of semantic segmentation models that conclude some estimates on their deployability in real environments, which is what the focus of this study will be. Aside from this, in order to offer more insight into their deployability, this study will also focus on analysing what instances are easier to segment under various types of corruptions. For example, vehicles are easily distinguishable from bicycles, and implicitly segmented better, but it may differ under some level of corruption. It is important to analyse these possibilities as it is essential in a safety-critical system that the precise segmentation of critical classes, such as humans, vehicles, roads, cyclists is more reliable than non-critical classes such as vegetation or the sky.

In order to address the issues described above, the following research questions are proposed:

RQ1: *Which image degradation type has the most negative impact on the performance of a CNN?*

RQ2: *How much image degradation can a CNN tolerate before the performance is too heavily impacted?*

RQ3: *Which class instances are more robust against image corruption?*

Image degradation can be of various types, such as compression, reduced resolution, blur, distortion, image noise, haziness and illumination, all of which impact the performance of a CNN, some more than others, but only a few are relevant in the context of road images, specifically:

1. compression
2. motion blur
3. shot noise

Compression is a relevant type of degradation because the internal computers of autonomous vehicles have a finite amount of processing capabilities. Due to the large number of cameras present on a vehicle, the amount of data collected at any given time can potentially exceed the rate at which it is processed, which is why compression is required. Compression reduces the total size of data, which helps to keep the total amount and processing speed on par.

Motion blur is the most probable type of blur present in images captured in traffic as an autonomous vehicle taking a capture would most likely be in motion. If, however, the vehicle is stationary, it is equally likely that subjects in the image would be in motion instead. Regardless of the case, both increase the likelihood of motion blur presence in images.

36th Twente Student Conference on IT Febr. 4th, 2022, Enschede, The Netherlands.

Shot noise exists because of the discrete nature of photons. The amount of photons captured by an optical device varies over time. In bright environments, this variation is negligible, but in low-light environments it becomes significant as the number of photons emitted is reduced.

This research aims to give insight into the effect image quality has on the ability of a CNN to semantically segment images with precision by simulating the degradation types enumerated above at different intensities that either reflect possible real-life occurrences or extreme cases. Besides, individual class robustness against the aforementioned corruptions will also be analyzed for the purpose of deriving real-setting deployability estimations.

The paper is structured as follows. Section 2 introduces in detail previous work on robustness studies and their findings. This is followed by section 3, which describes the experimental setup of this research. Section 4 details the three experiments that were performed in order to answer the research questions. Results are presented and discussed to some extent in section 5, and the research concludes with section 6.

2. RELATED WORK

The impact of image quality on the performance of a CNN has been studied before for face detection [17, 21], weed mapping [13] and general model robustness [14]. In this section, I describe the experimental setups and the respective findings only of [14, 13] as these are more closely related to the scope of this research.

2.1 Weed mapping study

This study analyzes the impact of image quality on the performance of a CNN for three main tasks, namely object detection, semantic segmentation and instance segmentation. However, I only focus on the semantic segmentation aspect as it aligns with the scope of this research. The model used for carrying out their experiments is DeepLabV3 [2], with ResNet-50 as the backbone architecture. The weights used with this model were pretrained by a subset of COCO train 2017 images on 20 classes.

No pre-existent dataset was used in this study, instead, image collection and annotation was done independently, and managed to collect a total of 2485 images. Training and testing was performed on normal images, but in order to study the impact of image quality on the performance, various types of degradation at various intensities were simulated and used for testing.

The conclusion drawn from these experiments is that image denoising — which reduces the noise, but can introduce blur — does not significantly impact the performance, regardless of intensity. Contrary to image denoising, there was a notable difference in performance for images with reduced resolution, showing that the model is sensitive to such a degradation. Overexposure and motion blur are concluded to only have a slight effect on the performance. Gaussian blur greatly impacted the visual quality of an image, but did not affect the performance too heavily, except for the greatest intensity used. Moreover, Gaussian noise at low levels appears to not affect performance in any significant way, except for some specific type of weeds. The model is thus considered to be resistant to noise at low levels, with only significant performance drops observed at greater noise levels.

Except for motion blur, none of the other degradation types align with the degradation types proposed for my research, but they offer nevertheless a good insight into the robustness of DeepLabV3 to various types of degradation and serve as a good guideline for organizing and conducting my own experiments.

2.2 Benchmarking the robustness of semantic segmentation models

The purpose of this study is to assess the robustness of DeepLabV3+ against various types of image degradation, but also to analyze how individual architectural properties of the model, such as atrous convolutions or atrous spatial pyramid pooling enhance the robustness against image degradation. Thus, two experiments are proposed. First is to benchmark DeepLabV3+ with a wide variety of network backbones on images with various types of degradation of different intensities. Second is to modify architectural properties of DeepLabV3+ one by one and again evaluate the robustness for this ablated model. The first experiment of robustness evaluation is quite extensive and compares performance of the model with ResNet-50, ResNet-101 [10], Xception-41 [4], Xception-65, Xception-71 and MobileNet-V2 [18] as network backbones under many types of degradation, some of which include motion, defocus and gaussian blur, impulse, shot and speckle noise and JPEG compression. Training and evaluation of the models were conducted on PASCAL VOC 2012 [7], Cityscapes [5] and ADE20K [20].

The study concludes that regardless of the network backbone used, blur is handled quite well and does not impact the performance significantly. On the other hand, noise appears to have a substantial impact on segmentation performance, but how much it impacts the performance is also dependent on the backbone used. MobileNet-V2, for example, which is a lightweight backbone performs significantly worse than other heavyweight backbones, such as ResNet or Xception. Lastly, JPEG compression also severely decreases performance, but this is again dependent on the network used.

While the first experiment of this study is very similar to the ones conducted in my paper, one important distinction is that I also try to determine an estimation in the deployability of the model in real settings by simulating corruptions also simulated in this study.

3. EXPERIMENTAL SETUP

This section introduces the model that was chosen to perform the experiments to be introduced in section 4, some key characteristics that make it the preferred choice, the network backbones, the dataset and the metric used for evaluating the model.

3.1 Model

The model employed for this research is DeepLabV3+ [3], regarded as the state-of-the-art model for semantic image segmentation at the time of writing this paper. DeepLabV3+ is an improvement of DeepLabV3 [2], as the former model uses a novel encoder-decoder structure, which actually employs the latter model as the encoder module, and a simple and effective decoder module. The purpose of an encoder is to progressively reduce the feature maps, enabling the extraction of higher semantic information, whereas the purpose of a decoder is to progressively recover spatial information lost while encoding. Figure 2 shows the structure of DeepLabV3+ and some of its characteristics that are going to be explained in the following sub-sections.

Figure 1: Sample compressed images with different quality values

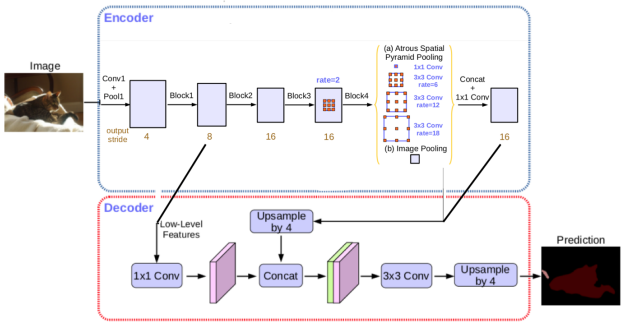


Figure 2: DeepLabV3+ model structure

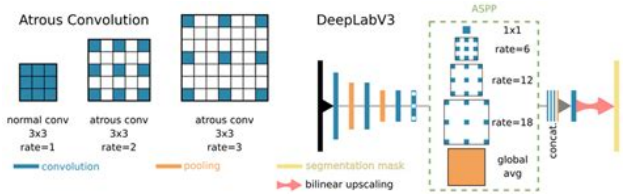


Figure 3: Atrous convolution (left side) samples with atrous rate = 1, 2 and 3; ASPP (right side) with 4 parallel atrous convolutions with rates = 1, 6, 12, 18

3.1.1 *Atrous Convolution*

Atrous (dilated) convolution [16, 8] is a tool that allows for the control of resolution of feature maps extracted by deep convolutional neural networks (DCNN), such as ResNet [10], and the adjustment of the filter's field-of-view, a technique that permits the capture of multi-scale information without impacting the computational complexity. It is similar to a standard convolution, except for the fact that the filter is up-sampled.

A visualization of atrous convolution can be observed in Figure 3(left side).

3.1.2 *Atrous Spatial Pyramid Pooling*

Atrous Spatial pyramid pooling (ASPP) [15, 9] is a technique also employed to extract multi-scale information. DeepLabV3+ incorporates this technique by performing four parallel atrous convolutions that can handle image segmentation at different scales. After all the atrous convolutions are finished, the results are concatenated and a final 1 x 1 convolution is applied on it to extract the final output.

Figure 3 displays on the right side a visualization of ASPP with rates 1, 6, 12 and 18. It can be noticed in the figure that aside from the parallel atrous convolutions, there is also a global average pooling (GAP). Global average pooling essentially averages the output of each feature maps of previous layer, reducing the amount of data significantly. This technique replaces the fully connected final layers paradigm. It is beneficial because it speeds up the training process of a model while also making it more robust against spatial translations of images.

3.2 Backbones

The network backbones used in this research are ResNet-101 [10] and MobileNet [12]. ResNet-101 is a heavyweight backbone which offers high precision at the cost of speed, while MobileNet is a lightweight backbone which opposed to ResNet-101, offers speed at the cost of precision. Training a model requires a significant amount of time and powerful hardware, which were not available, thus pretrained models on both of the backbones were used instead. The pretrained models were obtained from GitHub[1], with DeepLabV3+ on ResNet-101 backbone performing a score of 76.2% and DeepLabV3+ on MobileNet backbone performing a score of 72.1%, with both models being trained and evaluated on Cityscapes [5].

3.3 Dataset

While training and evaluation of a model was not performed individually, a dataset was still required in order to simulate corruptions and use them to evaluate the model on them. Cityscapes [5] dataset contains images captured in traffic and is therefore suitable for the purpose of this research. This dataset comprises 3475 diverse high-quality images split into 2975 images for training and 500 images for evaluation, and is divided into 19 instance classes.

3.4 Metrics

The most common metric used in semantic segmentation tasks is mean Intersection-over-Union (mIoU). This is also the metric employed for measuring the model performance in this research.

4. EXPERIMENTS

This section describes the experiments proposed for evaluating the robustness of the model and backbones introduced in Section 3.1 and Section 3.2.

4.1 Compression experiment

There are two possible types of compression, lossless and lossy. Lossless compression is a technique that does not lose any data in the compression process, whereas lossy compression suffers from loss of data depending on the amount of compression applied. For this experiment, however, only lossy compression will be used. A lossless compression technique would obviously be more desirable in the scenario of autonomous driving as it does not affect the data itself, only the size. However, it is expected that the results in this case would be identical, or remarkably similar at the very least to the results of uncompressed data. Moreover, a lossy compression technique should not be excluded as a good substitute for lossless compression

[1]https://github.com/VainF/DeepLabV3Plus-Pytorch

(a) Sample motion blurred images with different filter sizes

(b) Sample noisy images with different variations in photons per pixel

Figure 4: Examples of motion blurred and noisy images with the chosen values for the experiments

as it can usually reduce the size even further while keeping the loss of data to a minimum.

Having established the type of compression to be used, the specific amount of compression to be simulated for the experiment can be introduced. In JPEG (lossy) compression, the lowest compression ratio results in the highest possible quality ($Q = 100$) and the highest ratio results in the lowest possible quality ($Q = 1$). The compression ratio can vary from one image to another even if the same quality value is used, thus for consistency and simplicity, the level of compression used in this experiment will be denoted by the quality value Q. Five different levels of compression were generated for each image in Cityscapes in order to asses performance, specifically $Q = 90$ (high quality), 70 (medium quality), 50 (medium-low quality), 30 (low quality), 10 (very-low quality). A quality level $Q = 90$ and $Q = 70$ as seen in Figure 1 are more reflective of real-life compression applications as they reduce the size and mostly maintains the integrity of the data, compared to $Q = 30$ and 10, but these values were selected so as to also consider edge cases.

4.2 Motion blur experiment

Motion blur can have an aesthetic look in certain circumstances, but it most certainly is an undesired effect in mobile segmentation tasks. Blurring obscures the precise location of a subject in an image, which in turn impacts the ability of a CNN to semantically segment the image. This effect can happen because the instances in images captured on roads would most likely be moving. In the same fashion as described in the compression experiment, five distinct intensities of motion blurring were generated for each image contained in the dataset. In order to generate motion blur, a filter (n*n matrix) of varying sizes — generally odd sizes — is convoluted across the image. The filter sizes chosen for this experiment are $F = 3, 7, 11, 15, 19$. Figure 4a contains examples of motion blurred images with filters of the chosen sizes. Images blurred with filter sizes $F = 3$ and 7 are again more reflective of a real-life scenarios while filter sizes $F = 15$ and 19 are edge cases, with $F = 11$ being an in-between case.

4.3 Shot noise experiment

Shot noise can appear in images due to the nature of photons and their independent occurrence of each other. As the occurrence is independent of one another, the amount of photons captured by an optical device at any given time can be approximated by the Poisson distribution. As mentioned before, in bright environments the variation in photon emissions and capture is insignificant and does not result in any noticeable noise. In low-light environments, however, when the amount of photons emitted is small enough such that uncertainties described by the Poisson distribution can occur, the variation becomes significant and results in noise being generated. In other words, shot noise is a variation in the amount of photons detected on a spot, which when translated into an image results in pixels being inadequately color-coded. As it was the case for the previous experiments, shot noise of five distinct intensities were generated for each image contained in the dataset. The specific values in photons variation per pixel chosen for this experiment are $\sigma = 100$, 250, 500, 750, 1000. Figure 4b contains sample images with simulated shot noise for every variation in the number of photons per pixel chosen.

Q	avg. size reduction
90	20.15%
70	35.15%
50	44.37%
30	55.42%
10	76.91%

Table 1: Average image size reduction for each level of compression applied

5. RESULTS AND DISCUSSION

5.1 Effect of compression

The lowest level of compression applied, $Q = 90$, is found to have a very small impact on the performance of both models, resulting in a mIoU score reduction of only 0.84% for ResNet-101 and 1.41% for MobileNet (Figure 5). While this level of compression does not affect the performance significantly, it does reduce image size by 20.15% (Table 1) on average, which means that if the model-dependent performance loss is considered acceptable, it has a high applicability potential in real settings.

Increasing the amount of compression to $Q = 70$ has a more noticeable, although still relatively small, effect on ResNet-101 of 3.19% mIoU score reduction compared to

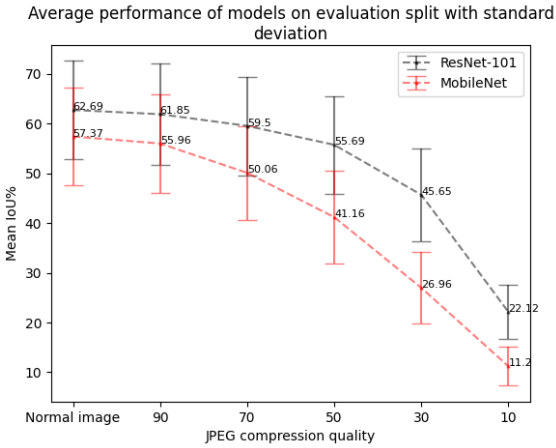


Figure 5: Average performance of ResNet-101 and MobileNet on evaluation split with standard deviation under compression

Backbone	Q	mIoU%	Perf. impact	std. dev
ResNet-101	none	62.69%	0%	9.86%
	90	61.85%	0.84%	10.23%
	70	59.5%	3.19%	9.92%
	50	55.69%	7%	9.83%
	30	45.65%	17.04%	9.25%
	10	22.12%	40.57%	5.5%
MobileNet	none	57.37%	0%	9.75%
	90	55.96%	1.41%	9.9%
	70	50.06%	7.31%	9.43%
	50	41.16%	16.21%	9.31%
	30	26.96%	30.41%	7.21%
	10	11.20%	46.17%	3.9%

Table 2: Results of ResNet-101 and MobileNet under different levels of compression

the normal image, and an even greater average size reduction of 35.15%. The same cannot be said about MobileNet, which suffers a rather significant 7.31% mIoU score reduction. Applying even more compression naturally impacts the performance even further as there is increasingly more loss of detail and high-frequency information in the image. The results for the remaining compression levels are summarised in Table 2. The output generated under various amounts of compression can be visualized in Figure 16 for ResNet-101 and Figure 17 for MobileNet (Appendix).

Individual class robustness against compression can be visualized in Figure 6 and Figure 7. The best performing classes in both cases are road, vegetation, buildings, car and sky. DeepLabV3+ performs very well on the road class, with ResNet-101 performing marginally better than MobileNet for $Q = 90$ and 70. For higher compression values, compared to ResNet-101, MobileNet suffers from a greater reduction in performance, but also from a higher variability in precision. The same comparison holds for the sky, vegetation, car and building classes. The behaviour of the other critical classes, such as person, bicycle, traffic sign and traffic light is different between the models. ResNet-101 performs noticeably better on these classes overall, although not at the level of some non-critical classes, and also exhibits a higher degree of robustness compared to MobileNet. Some of these classes likely perform worse because they occupy a smaller area of the image compared to road, vegetation or sky classes.

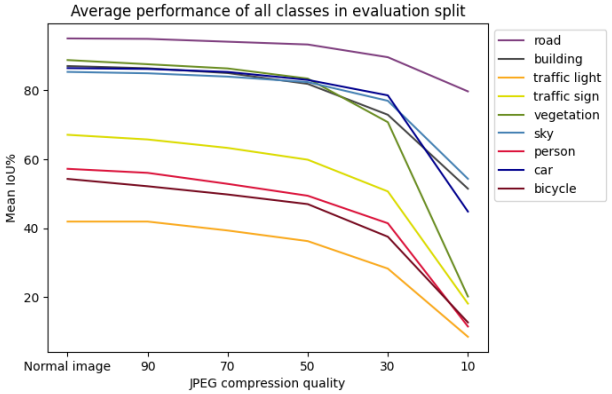


Figure 6: Average performance of critical and best performing classes with ResNet-101 under compression

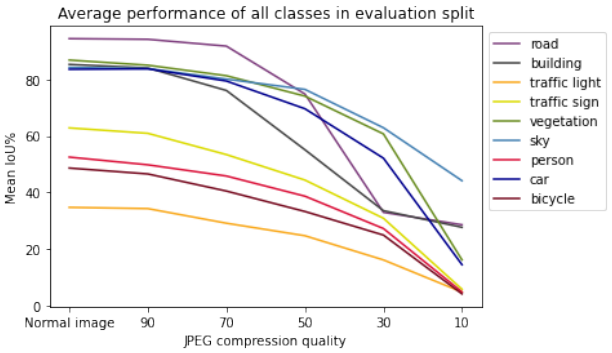


Figure 7: Average performance of critical and best performing classes with MobileNet under compression

However, for the lowest amount of compression tested, the performance of most classes is unaffected, and in the case of ResNet-101, even extreme amounts of compression can be tolerated.

5.2 Effect of motion blur

Motion blur has a minimal effect on performance at the lowest intensity as depicted in Figure 8. A filter of size 3 affects ResNet-101 performance by 1.12% and MobileNet by 0.96%, which shows that, although MobileNet does not perform at the level of ResNet-101, it handles very low levels of motion blur slightly better. Already from the second lowest motion blur intensity tested, the impact on performance is significant, resulting in a mIoU score decrease of 8.12% for ResNet-101 and 10.71% for MobileNet. Contrary to the findings in [13], where the motion blur filter of size 7 is found to have little effect on performance for weed mapping, in this case it highlights the sensitivity to motion blur and that the effect is significant. This is likely caused by the obscuring of fine details and precise location of instances. The complete results are summarised in Table 3. Compared to compression, motion blur does not have any practical use in real settings. Combining this with the discovery that both models only tolerate very low intensities of this type of degradation, if any model were to be deployed, measures to remove as much of it as possible should be in place.

The average performance of some classes is illustrated in Figure 9 and Figure 10. The road class is shown to be the most robust against all intensities of motion blur for both models.

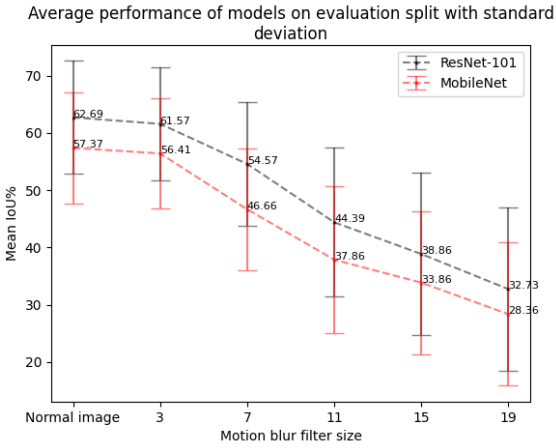


Figure 8: Average performance of ResNet-101 and MobileNet on evaluation split with standard deviation on blurred images

Backbone	F	mIoU%	Perf. impact	std. dev
ResNet-101	none	62.69%	0%	9.86%
	3	61.57%	1.12%	9.89%
	7	54.57%	8.12%	10.77%
	11	44.39%	18.3%	13.01%
	15	38.86%	23.83%	14.18%
	19	32.73%	29.96%	14.19%
MobileNet	none	57.37%	0%	9.75%
	3	56.41%	0.96%	9.68%
	7	46.66%	10.71%	10.63%
	11	37.86%	19.51%	12.77%
	15	33.86%	23.51%	12.52%
	19	28.36%	29.01%	12.48%

Table 3: Results of ResNet-101 and MobileNet evaluation split under motion blur

The only notable difference between the models in segmentation performance is the vegetation and sky classes. MobileNet outperforms ResNet-101 by a large margin on vegetation, and in the case of sky class, it is the exact opposite, with ResNet-101 outperforming MobileNet. However, this is not as relevant as performing better on critical classes, which in both cases are observed to perform worse than non-critical classes. Compared to the previous effects of compression, where most classes were shown to exhibit robustness when using ResNet-101, in the case of motion blur that does not hold.

5.3 Effect of shot noise

Although the performance impact caused by the lowest noise intensity ($\sigma = 1000$) is greater for both ResNet-101 and MobileNet than the lowest intensities used in the other experiments ($Q = 90$ and $F = 3$), the overall impact is gentler. The lowest noise intensity results in a 2.68% score reduction for ResNet-101 and 3.07% for MobileNet (Figure 11). The following two intensities do not affect the performance of ResNet-101 significantly, only that of MobileNet, which suffers a 6.56% reduction. This is evidence that DeepLabV3+ is fairly resistant to shot noise, with the obvious exception of extreme levels. As before, the complete results are summarised in Table 4.

Individual average class performance is presented in Figure 12a and Figure 12b. As it was the case for previous results, the road class is not affected in any significant way when

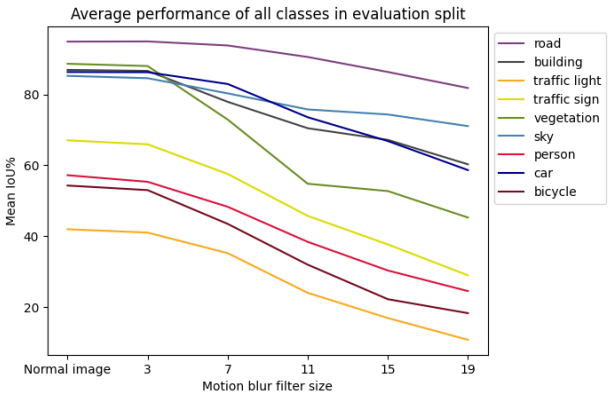


Figure 9: Average performance of critical and best performing classes with ResNet-101 on motion blurred images

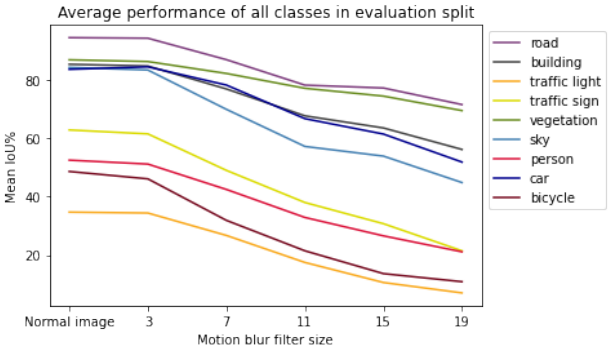


Figure 10: Average performance of critical and best performing classes with MobileNet on motion blurred images

using DeepLabV3+ with ResNet-101. In comparison to previous results where MobileNet performance on the road class suffered a noticeable drop under moderate amounts of compression and motion blur, such an impact is hardly noticeable for noise corruption. This also holds for all the other classes, which shows that both models can in fact tolerate moderate and even high amounts of noise better than compression or motion blur.

5.4 Performance on specific environments

Also depicted in Figure 5 is the standard deviation of model performance for a given compression level.

The standard deviation suggests in this case that there is a relatively high variability (9.86% for normal image) in segmentation performance. Considering that the evaluation split is further divided into three other distinct splits with images from Frankfurt, Lindau and Munster, the performance of each individual split was also analysed in an attempt to uncover the source of the aforementioned variability, but also to study general model deployability. Figure 13, 14 and 15 (Appendix) illustrate the average performance of ResNet-101 on images contained in the Frankfurt, Lindau and Munster splits, and Table 5 summarizes the results. In both cases, the city splits Frankfurt and Munster perform relatively similar, within 4% of the evaluation split average, but it is the Lindau city split that not only performs significantly worse, but also has a significantly higher standard deviation compared to the others. Similar results were obtained for MobileNet and the other experiments. This is a strong indication that there are some environments in which the model performs worse than expected, although this assumption can only be con-

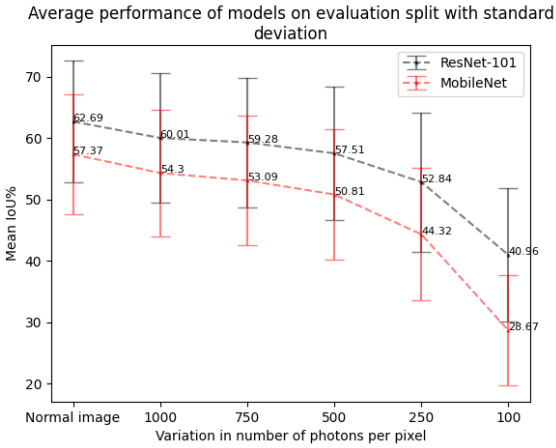


Figure 11: Average performance of ResNet-101 and MobileNet on evaluation split with standard deviation on noisy images

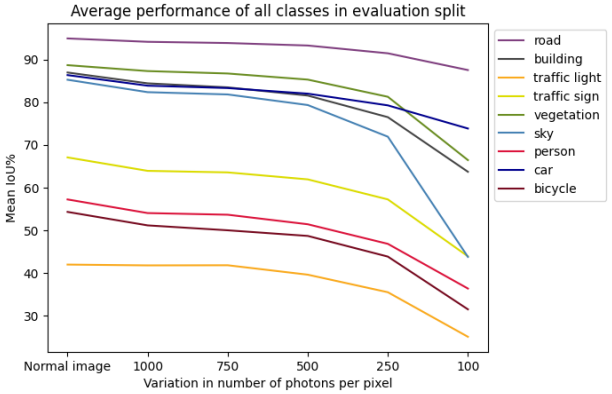
Backbone	σ	mIoU%	Perf. impact	std. dev.
ResNet-101	none	62.69%	0%	9.86%
	1000	60.01%	2.68%	10.54%
	750	59.28%	3.41%	10.52%
	500	57.51%	5.18%	10.81%
	250	52.84%	9.85%	11.31%
	100	40.96%	21.73%	10.84%
MobileNet	none	57.37%	0%	9.75%
	1000	54.3%	3.07%	10.3%
	750	53.09%	4.28%	10.49%
	500	50.81%	6.56%	10.61%
	250	44.32%	13.05%	10.77%
	100	28.67%	28.7%	9.0%

Table 4: Results of ResNet-101 and MobileNet evaluation split under shot noise

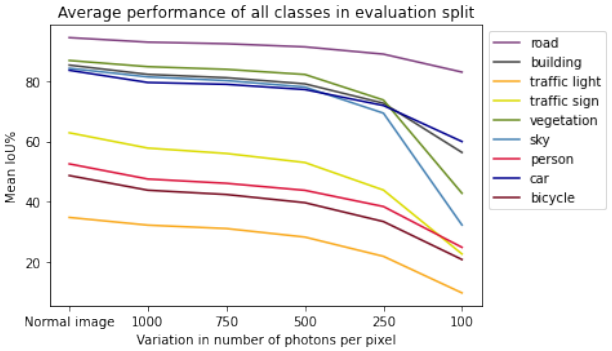
fidently made with regards to the road class as it is the only one that can differ significantly from one environment to another.

6. CONCLUSION

This study presents a robustness analysis of DeepLabV3+ with two distinct backbones against compression, motion blur and shot noise. The most impactful at low intensities is motion blur, which affects the performance of both models significantly, revealing that it is not well tolerated. At high intensities, the impact of compression surpasses that of motion blur, but in general, compression is tolerated better. Shot noise does not affect the performance significantly, except for the two highest levels tested. The road class is, in general, segmented with the highest precision by a noticeable margin, regardless of the degradation or intensity, with the only exception being MobileNet under very high compression. It's performance is then followed by vegetation, building, car and sky classes. Out of these, only two can be confidently considered to be critical classes in the context of autonomous driving, namely road and car. The rest are not as relevant for the safety of traffic participants. The remaining critical classes, bicycle, person, traffic sign and traffic light perform worse than the aforementioned ones. Also, instance class robustness is highly dependent on degradation type, but also the model employed.



(a) Average performance of critical and best performing classes with ResNet-101 on noisy images



(b) Average performance of critical and best performing classes with MobileNet on noisy images

Figure 12: Average performance of critical and best performing classes with ResNet-101 and MobileNet on noisy images

7. FUTURE WORK

Compression, motion blur and shot noise were each simulated independently. In a real-setting, it is likely that these types of degradation would occur simultaneously, and it is unknown whether the impact would be even greater or remain the same. Simulating a wider variety of degradations would also be interesting to analyze.

As for the individual instance classes, it would be worth researching if training with fewer instance classes influences the precise segmentation of critical classes. The pretrained models used throughout this research were trained on 19 classes, some of which would not occur that often. Merging related instances together, such as trucks and cars, thus reducing the number of classes, may positively influence the performance overall.

8. REFERENCES

[1] A. Azulay and Y. Weiss. Why do deep convolutional networks generalize so poorly to small image transformations? *arXiv preprint arXiv:1805.12177*, 2018.

[2] L.-C. Chen, G. Papandreou, F. Schroff, and H. Adam. Rethinking atrous convolution for semantic image segmentation. *arXiv preprint arXiv:1706.05587*, 2017.

[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.

[4] F. Chollet. Xception: Deep learning with depthwise separable convolutions. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 1251–1258, 2017.

[5] 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.

[6] S. Dodge and L. Karam. Understanding how image quality affects deep neural networks. In *2016 eighth international conference on quality of multimedia experience (QoMEX)*, pages 1–6. IEEE, 2016.

[7] M. Everingham, L. Van Gool, C. K. I. Williams, J. Winn, and A. Zisserman. The pascal visual object classes (voc) challenge. *International Journal of Computer Vision*, 88(2):303–338, June 2010.

[8] A. Giusti, D. C. Cireşan, J. Masci, L. M. Gambardella, and J. Schmidhuber. Fast image scanning with deep max-pooling convolutional neural networks. In *2013 IEEE International Conference on Image Processing*, pages 4034–4038. IEEE, 2013.

[9] K. Grauman and T. Darrell. The pyramid match kernel: Discriminative classification with sets of image features. In *Tenth IEEE International Conference on Computer Vision (ICCV'05) Volume 1*, volume 2, pages 1458–1465. IEEE, 2005.

[10] 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 (CVPR)*, June 2016.

[11] D. Hendrycks and T. Dietterich. Benchmarking neural network robustness to common corruptions and perturbations. *arXiv preprint arXiv:1903.12261*, 2019.

[12] A. G. Howard, M. Zhu, B. Chen, D. Kalenichenko, W. Wang, T. Weyand, M. Andreetto, and H. Adam. Mobilenets: Efficient convolutional neural networks for mobile vision applications. *arXiv preprint arXiv:1704.04861*, 2017.

[13] C. Hu, B. B. Sapkota, J. A. Thomasson, and M. V. Bagavathiannan. Influence of image quality and light consistency on the performance of convolutional neural networks for weed mapping. *Remote Sensing*, 13(11), 2021.

[14] C. Kamann and C. Rother. Benchmarking the robustness of semantic segmentation models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 8828–8838, 2020.

[15] S. Lazebnik, C. Schmid, and J. Ponce. Beyond bags of features: Spatial pyramid matching for recognizing natural scene categories. In *2006 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'06)*, volume 2, pages 2169–2178. IEEE, 2006.

[16] G. Papandreou, I. Kokkinos, and P.-A. Savalle. Modeling local and global deformations in deep learning: Epitomic convolution, multiple instance learning, and sliding window detection. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 390–399, 2015.

[17] M. Rezaei, E. Ravanbakhsh, E. Namjoo, and M. Haghighat. Assessing the effect of image quality on ssd and faster r-cnn networks for face detection. In *2019 27th Iranian Conference on Electrical Engineering (ICEE)*, pages 1589–1594, 2019.

[18] M. Sandler, A. Howard, M. Zhu, A. Zhmoginov, and L.-C. Chen. Mobilenetv2: Inverted residuals and linear bottlenecks. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 4510–4520, 2018.

[19] I. Vasiljevic, A. Chakrabarti, and G. Shakhnarovich. Examining the impact of blur on recognition by convolutional networks. *arXiv preprint arXiv:1611.05760*, 2016.

[20] B. Zhou, H. Zhao, X. Puig, S. Fidler, A. Barriuso, and A. Torralba. Scene parsing through ade20k dataset. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 633–641, 2017.

[21] Y. Zhou, D. Liu, and T. Huang. Survey of face detection on low-quality images. In *2018 13th IEEE international conference on automatic face & gesture recognition (FG 2018)*, pages 769–773. IEEE, 2018.

APPENDIX

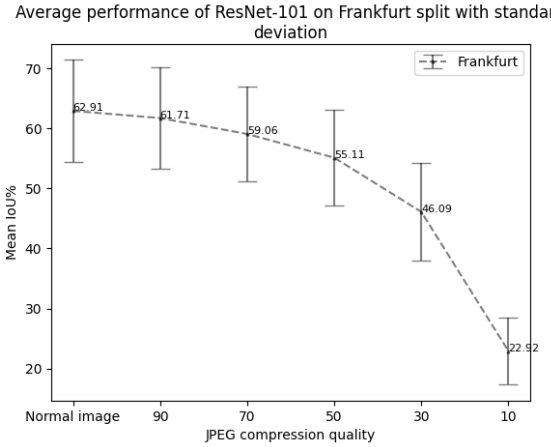


Figure 13: Average performance on Frankfurt data split with ResNet-101

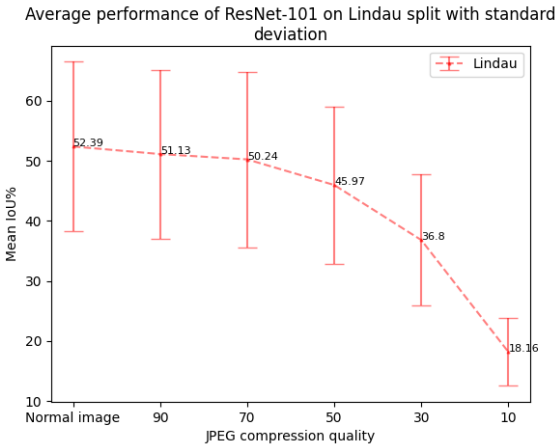


Figure 14: Average performance on Lindau data split with ResNet-101

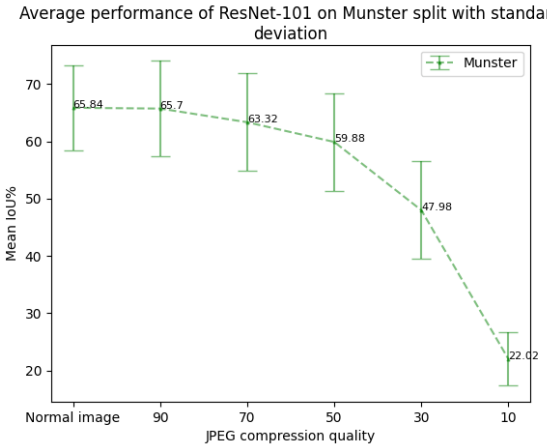
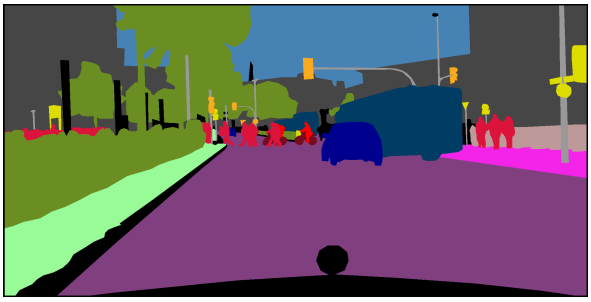


Figure 15: Average performance on Munster data split with ResNet-101

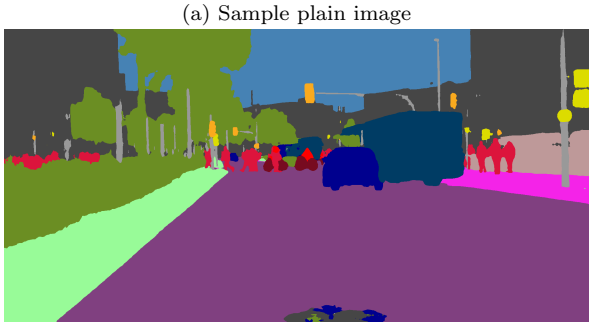
City split	Q	mIoU%	Perf. impact	std. dev
Frankfurt	none	62.91%	0%	8.48%
	90	61.71%	1.2%	8.51%
	70	59.06%	3.85%	7.81%
	50	55.11%	7.8%	7.94%
	30	46.09%	16.82%	8.12%
	10	22.52%	40,39%	5.61%
Lindau	none	52.39%	0%	14.11%
	90	51.13%	1.26%	14.05%
	70	50.24%	2.15%	14.63%
	50	45.97%	6.32%	13.07%
	30	36.8%	15.59%	10.94%
	10	18.16%	34.23%	5.64%
Munster	none	65.84%	0%	7.46%
	90	65.7%	0.14%	8.31%
	70	63.32%	2.52%	8.52%
	50	59.88%	5.96%	8.52%
	30	47.98%	17.86%	8.52%
	10	22.02%	43.82%	4.65%

Table 5: Results of ResNet-101 on Frankfurt, Lindau and Munster data splits under compression

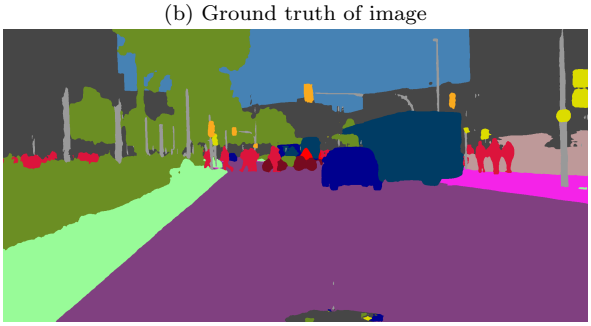
(a) Sample plain image



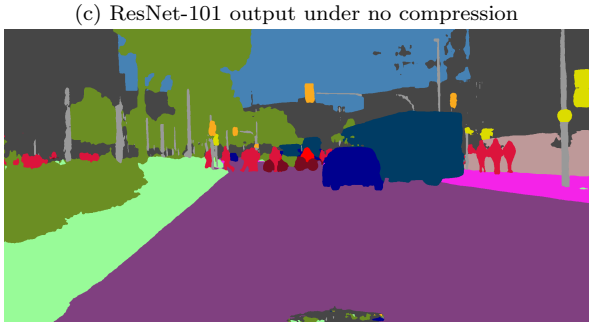
(b) Ground truth of image



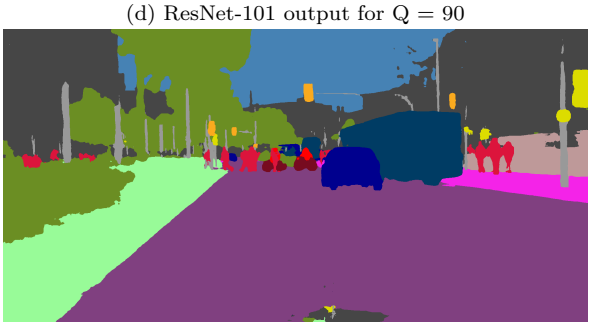
(c) ResNet-101 output under no compression



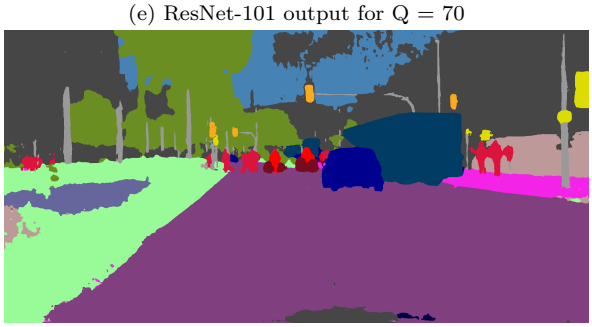
(d) ResNet-101 output for Q = 90



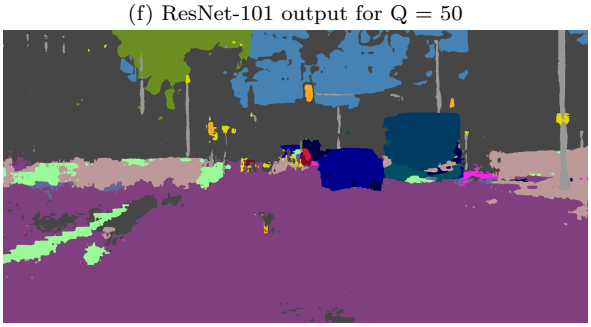
(e) ResNet-101 output for Q = 70



(f) ResNet-101 output for Q = 50



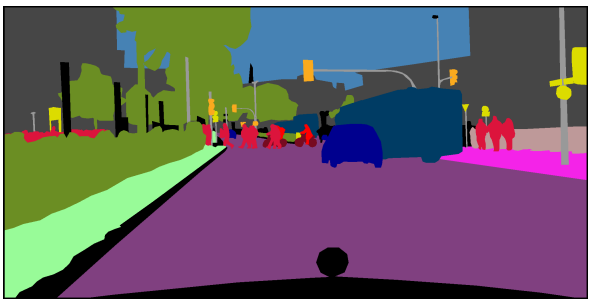
(g) ResNet-101 output for Q = 30



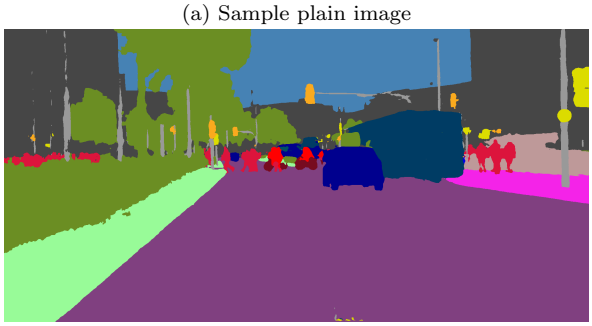
(h) ResNet-101 output for Q = 10

Figure 16: Output of ResNet-101 under various amounts of compression

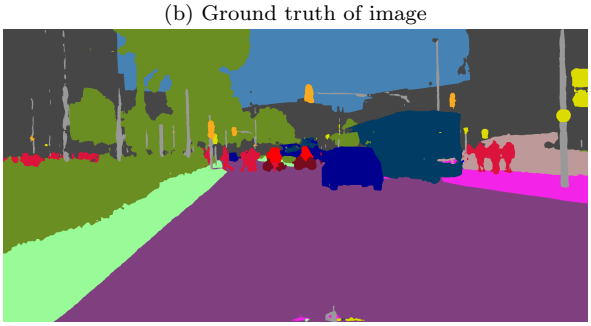
(a) Sample plain image



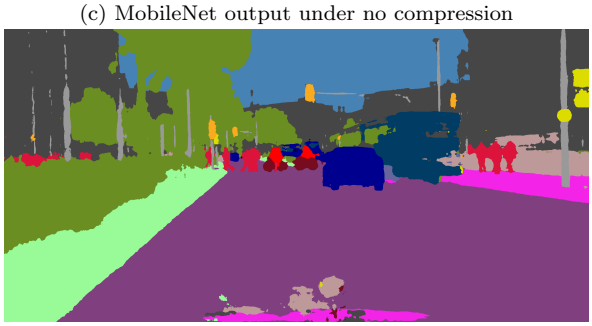
(b) Ground truth of image



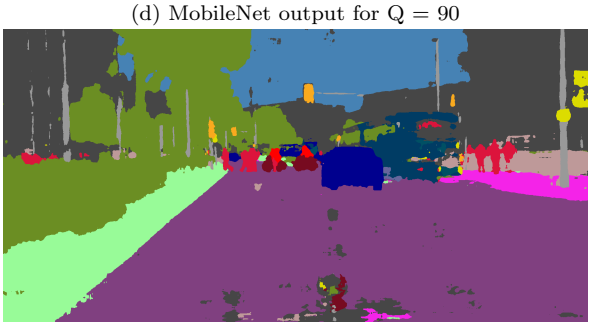
(c) MobileNet output under no compression



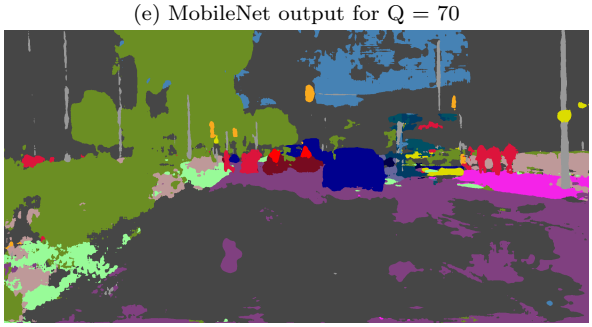
(d) MobileNet output for Q = 90



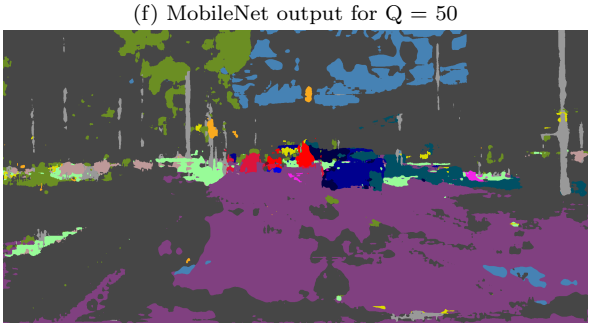
(e) MobileNet output for Q = 70



(f) MobileNet output for Q = 50



(g) MobileNet output for Q = 30



(h) MobileNet output for Q = 10

Figure 17: Output of MobileNet under various amounts of compression