Hochschule
Bonn-Rhein-Sieg
University of Applied Sciences

b-it Bonn-Aachen International Center for Information Technology

R&D Project

Semantic Segmentation using Resource Efficient Deep Learning

Naresh Kumar Gurulingan

Submitted to Hochschule Bonn-Rhein-Sieg,
Department of Computer Science
in partial fullfilment of the requirements for the degree
of Master of Science in Autonomous Systems

Supervised by

Prof. Dr Paul G. Plger
M. Sc. Deebul Nair

August 2018

I, the undersigned below, declare that this work has not previously been submitted to this or any other university and that it is, unless otherwise stated, entirely my own work.

Date	Naresh Kumar Gurulingan

Abstract

Your abstract

Acknowledgements

Thanks to ....

Contents

List of Figures

List of Tables

1

Introduction

1.1 Motivation

1.1.1 ...

1.1.2 ...

1.2 Challenges and Difficulties

1.2.1 ...

1.2.2 ...

1.2.3 ...

1.3 Problem Statement

1.3.1 ...

1.3.2 ...

1.3.3 ...

2

State of the Art

2.1 ....

Use as many sections as you need in your related work to group content into logical groups

Don't forget to correctly cite your sources [?].

2.2 Limitations of previous work

3

Methodology

3.1 Semantic segmentation architectures:

Inline with the goal of the project to use resource efficient deep learning in terms of inference time and storage memory, the deepLab v3+ model with mobileNetv2 and xception backbones was chosen.

4 Solution

Your main contributions go here

4.1 Proposed algorithm

4.2 Implementation details

5

Evaluation

Implementation and measurements.

6

Results

Since the major contribution of this work is the creation of the dataset, the experiments are focussed on validating the effectiveness of the dataset.

6.0.1 Results on the complete validation set

Deeplabv3+ with both the mobileNet backbone and the xception backbone are evaluated on all variants of variety of backgrounds and white backgrounds dataset. From 6.1, it is evident that the Mean IOU obtained on each variant is dependent on the properties of objects in the variant. The atWork_full variant treats all the 18 objects in the dataset as different classes. As a result, for instance, m20 and m30 have different labels despite the fact that the two objects only differ in size and slightly in color. The segmentation model is thus forced to distinguish between such objects. Since the objects occur in the dataset in arbitrary scales and are subject to differences in illumination, the real world differences between such similar objects become insignificant in the dataset. Thus, the Mean IOU obtained in the atWork_full variant is indeed the lowest as expected. The two variants atWork_size_invariant and atWork_similar_shapes combine objects which are similar. As a result, the segmentation model achives better Mean IOU on these variants. The atWork_binary variant requries the segmentation model to only distinguish foreground from background leading to the highest MIOU. From 6.2, deepLabV3+ with the xception backbone, evidently, also follows a similar trend like deepLabv3+ with mobileNet backbone.

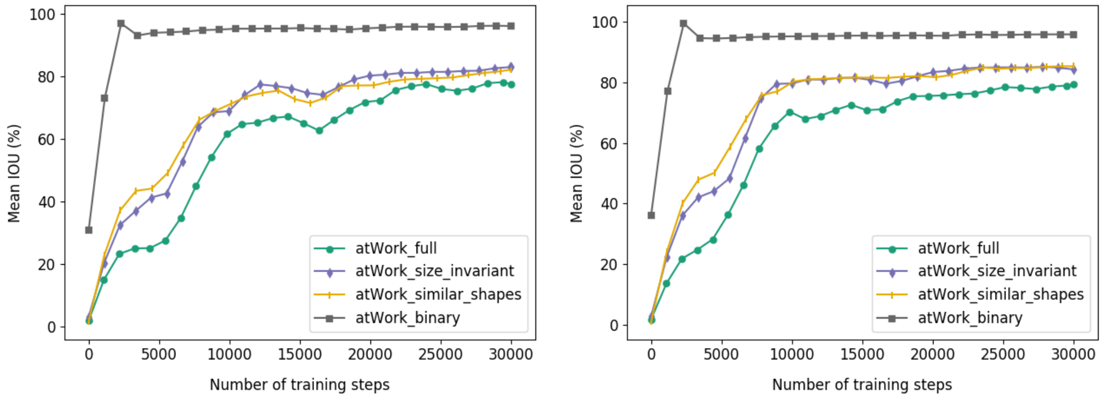
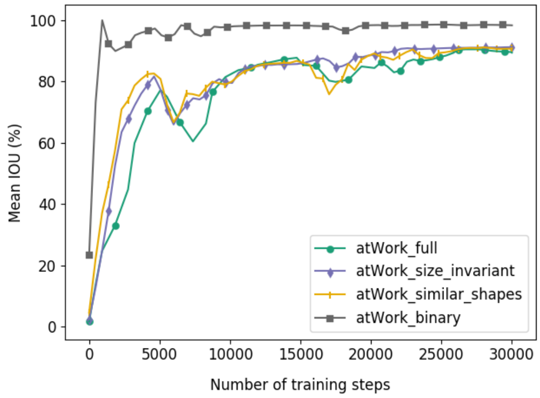


Figure 6.1: Mean IOU of deeplabv3+ with **mobileNet backbone** on variety of backgrounds dataset and white backgrounds dataset is shown. Mean IOU on the 4 variants of the variety of backgrounds dataset(left): atWork_full = 77.47%, atWork_size_invariant = 83.10%, atWork_similar_shapes = 82.10% and atWork_binary = 96.06%. Mean IOU on the 4 variants of the white backgrounds dataset(right): atWork_full = 79.26%, atWork_size_invariant = 84.29%, atWork_similar_shapes = 85.33% and atWork_binary = 95.83%.

6.0.2 Results on only the real validation set

Variant	Real training data	Variety of backgrounds all training data	White backgrounds all training data	Variety of backgrounds artificial training data
atWork_full	83.21	71.72	70.8	40.0
atWork_size_invariant	85.01	80.08	77.12	47.76
atWork_similar_shapes	79.83	77.33	76.47	43.31
atWork_binary	94.33	93.01	90.17	43.29

Table 6.1: This table summarizes the results obtained when validating only on the real validation data. The first column denotes the variant. The remaining columns denote on what data was the deepLabv3+ with mobileNet backbone model trained on. All the Mean IOUs are in percentage.



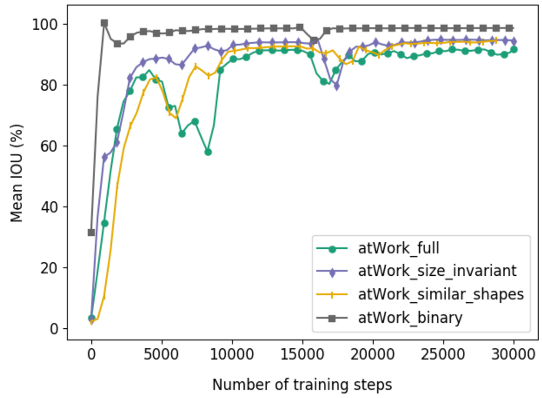


Figure 6.2: Mean IOU of deeplabv3+ with **xception backbone** on variety of backgrounds dataset and white backgrounds dataset is shown. Mean IOU on the 4 variants of the variety of backgrounds dataset(left): atWork_full = 89.38%, atWork_size_invariant = 91.19%, atWork_similar_shapes = 90.81% and atWork_binary = 98.31%. Mean IOU on the 4 variants of the white backgrounds dataset(right): atWork_full = 91.59%, atWork_size_invariant = 94.27%, atWork_similar_shapes = 94.33% and atWork_binary = 98.47%.

6.0.3 Performance on individual classes

6.0.4 Learning rate

6.0.5 Class balancing

6.0.6 Performance of quantized models

6.0.7 Transfer learning

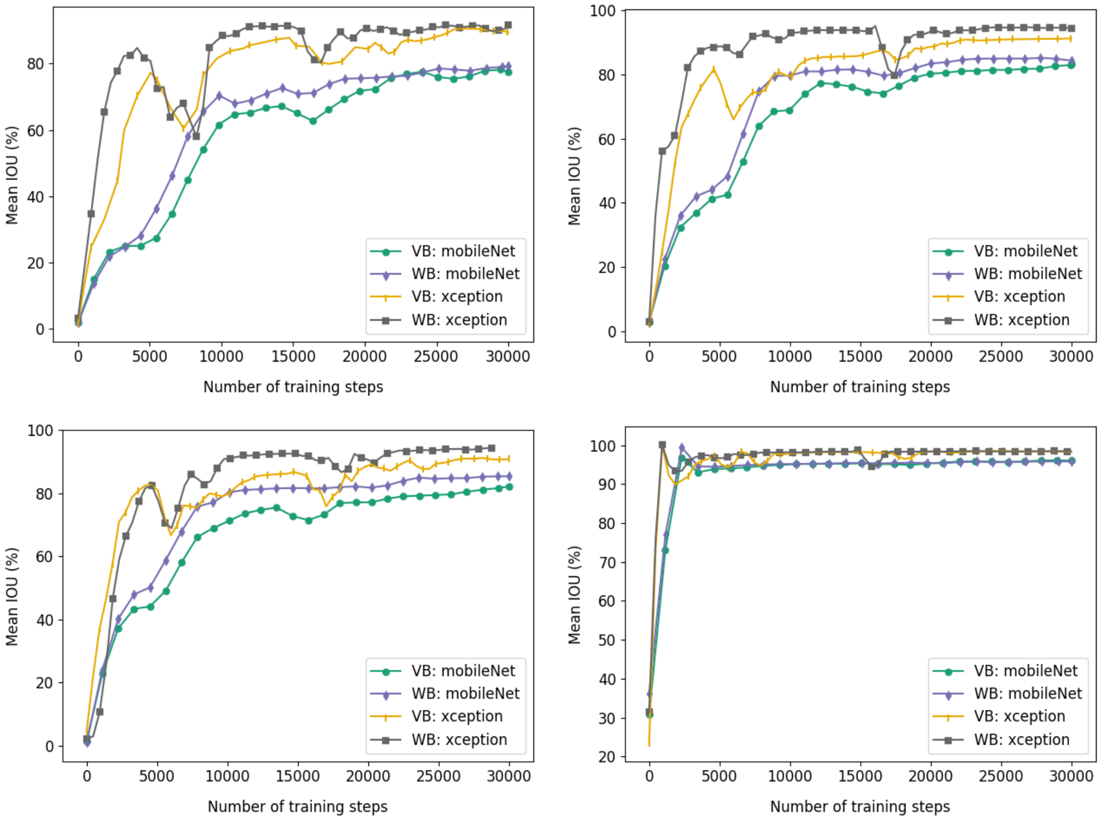


Figure 6.3: Comparision of Mean IOU obtained by deepLabv3+ using mobileNet backbone Vs xception backbone on all 4 variants. VB denotes variety of backgrounds dataset and WB denotes white backgrounds dataset. Top left: atWork_full variant, top right: atWork_size_invariant, bottom left: atWork_similar_shapes and bottom left: atWork_binary.

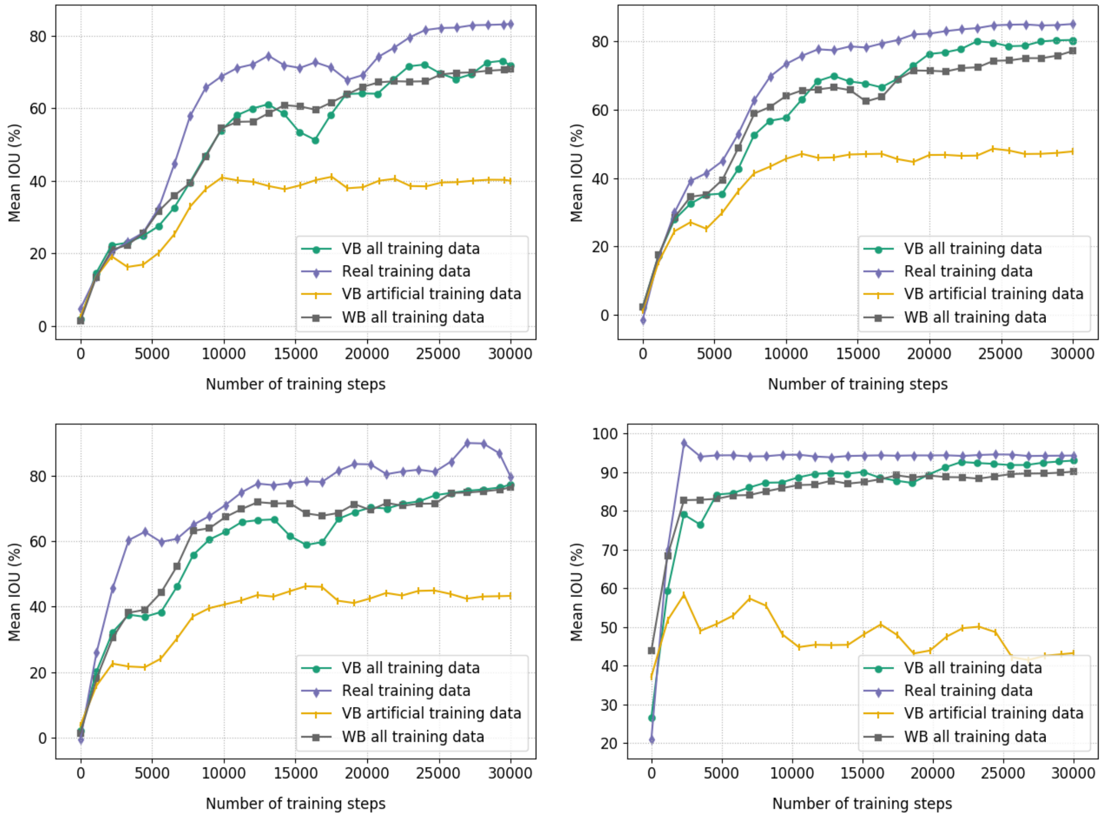


Figure 6.4: Mean IOU on all 4 variants obtained by deepLabv3+ with mobileNet backbone when validated only on the real validation data. VB stands for variety of backgrounds dataset and WB stands for white backgrounds dataset. Top left: atWork_full variant, top right: atWork_size_invariant, bottom left: atWork_similar_shapes and bottom left: atWork_binary. The Mean IOUs are tabulated in 6.1

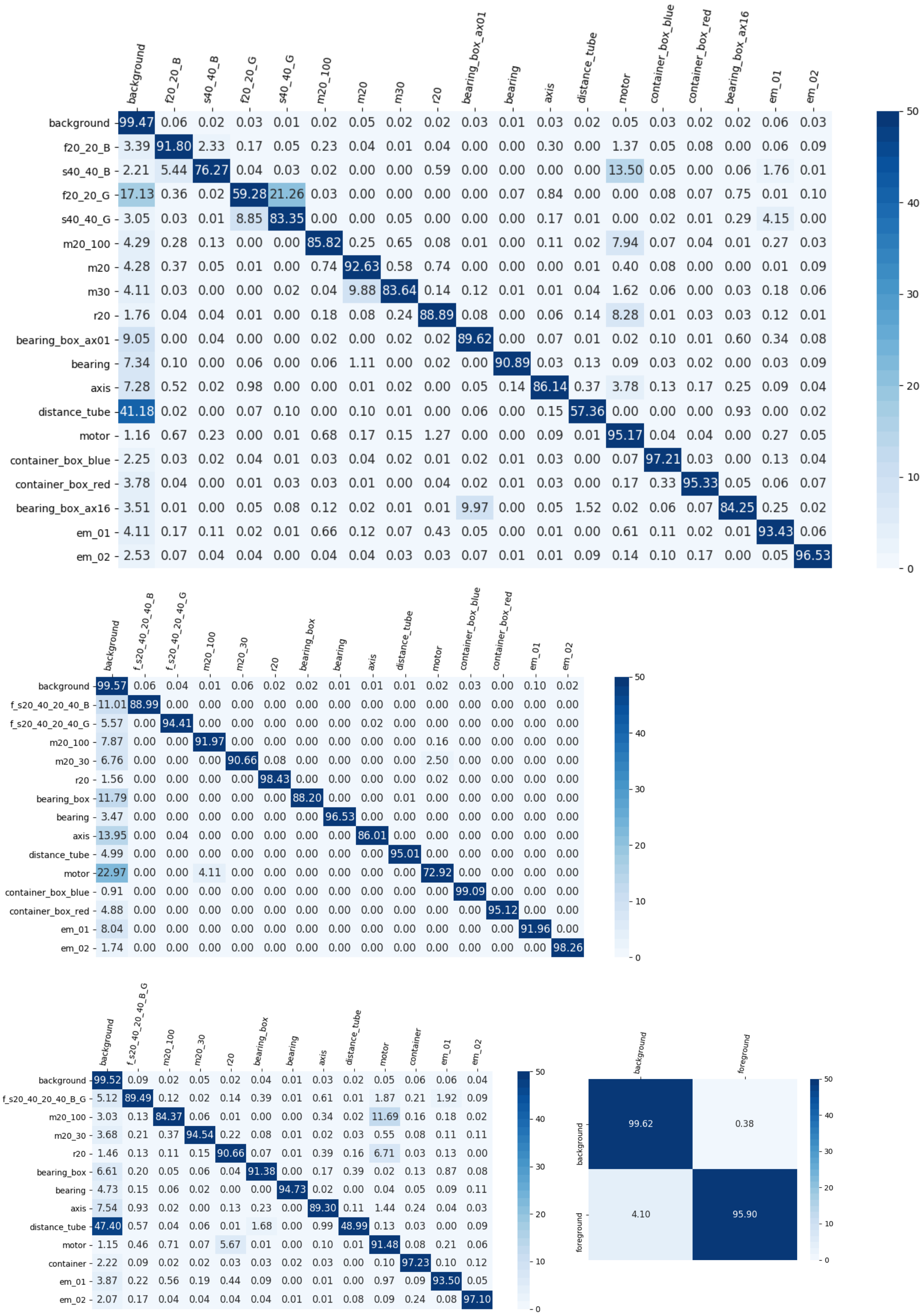


Figure 6.5: Confusion matrix based on number of classified pixels on all 4 variants of the variety of backgrounds dataset. The number of pixels in each row is normalized by the total number of pixels in the row. First row: atWork_full variant, second row: atWork_size_invariant, third row left: atWork_similar_shapes and third row left: atWork_binary.

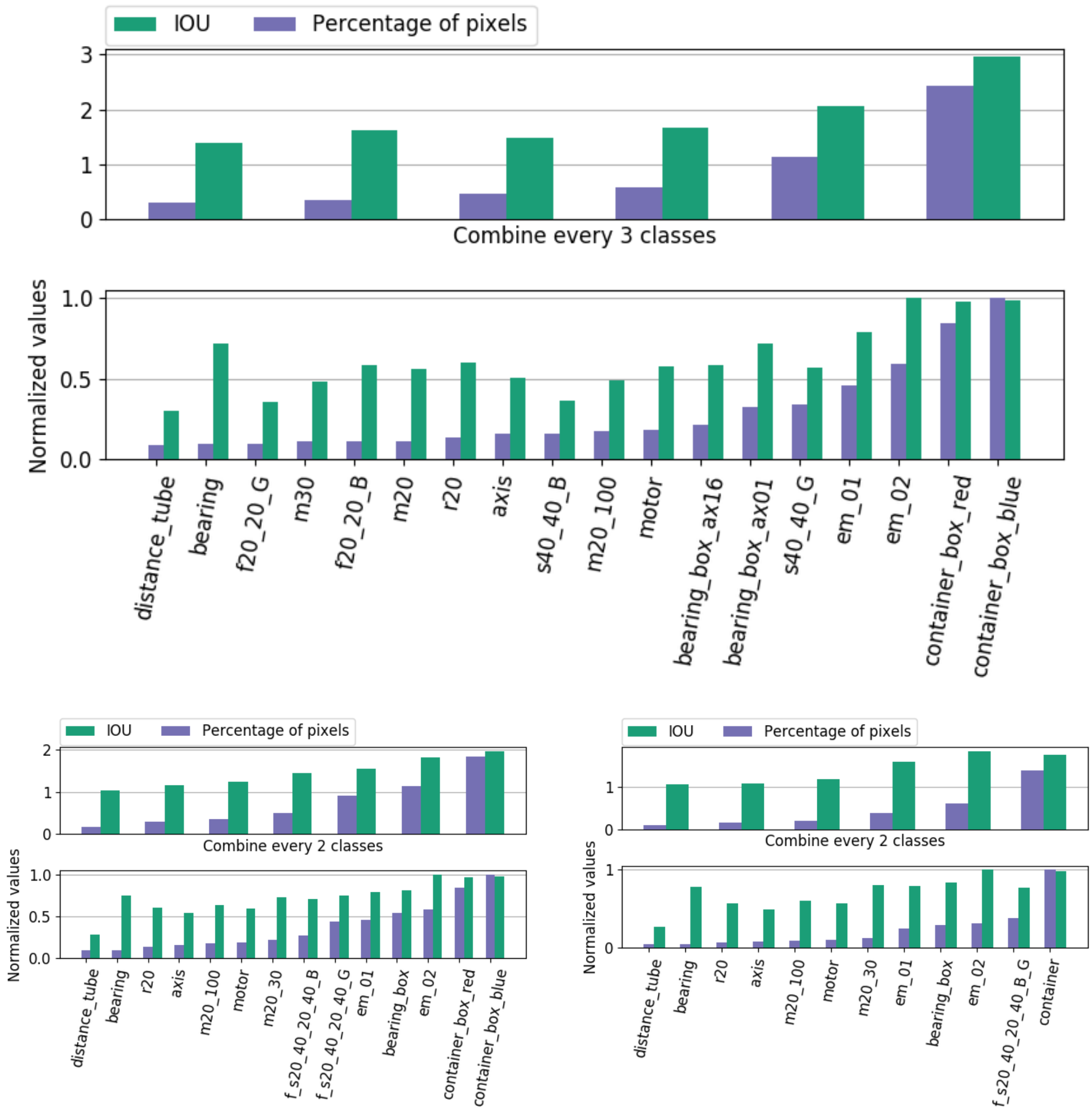


Figure 6.6: Individual class IOUs plotted with the percentage of pixels occupied on all 4 variants of the variety of backgrounds dataset. The number of pixels in each row is normalized by the total number of pixels in the row. Top left: atWork_full variant, top right: atWork_size_invariant, bottom left atWork_similar_shapes and bottom left atWork_binary.

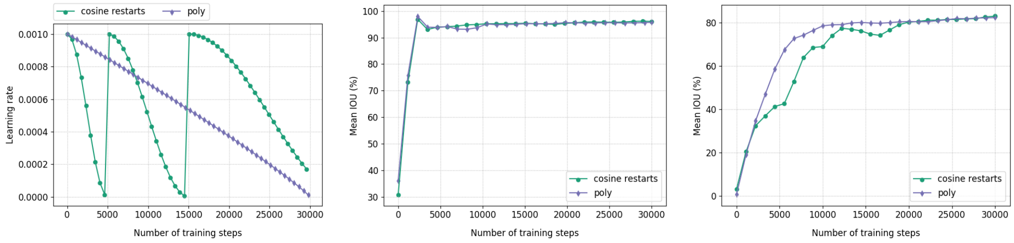


Figure 6.7: Learning rate decay with two different policies 1. cosine restarts and 2. poly is compared. Left: learning rate over 30000 steps with the two decay policies. Middle: Mean IOU on the validation set of atWork_binary variant is 96.06 % with cosine restarts and 95.75 % with poly. Right: Mean IOU on the validation set of atWork_size_invariant variant is 83.1 % with cosine restarts and 82.24 % with poly.

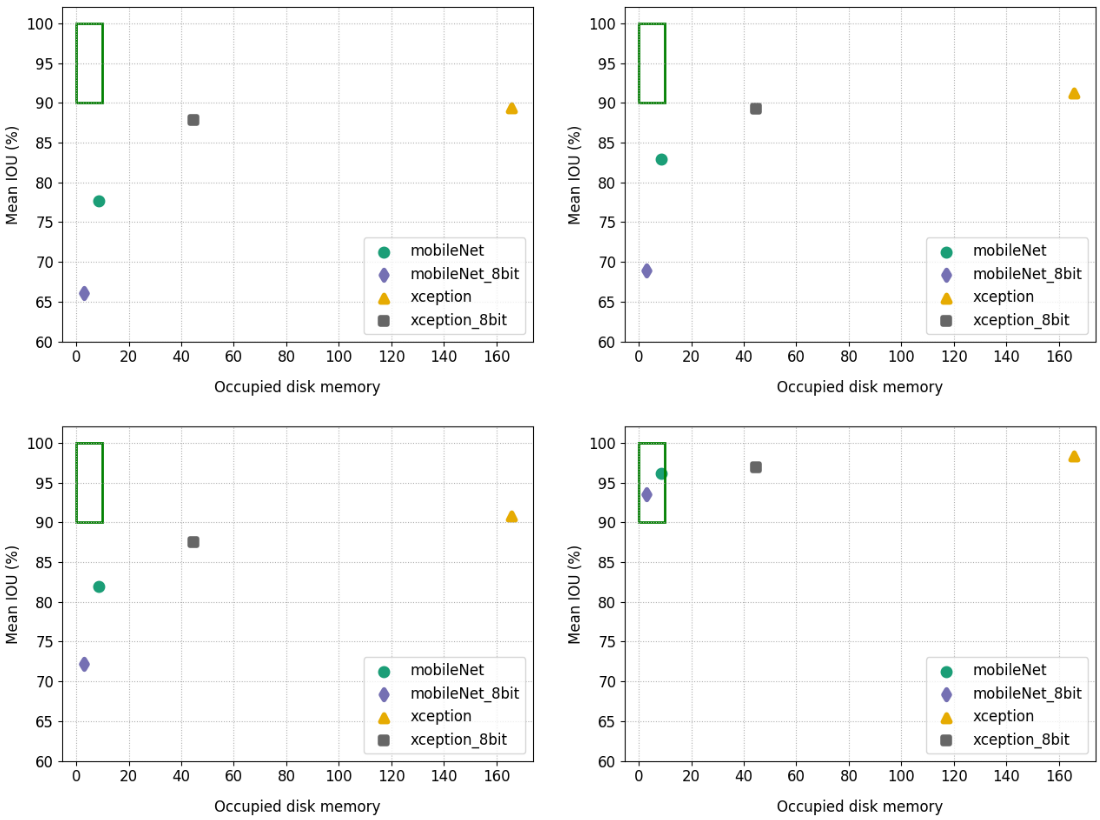


Figure 6.8: .

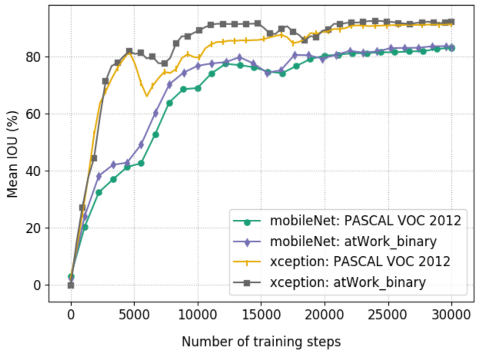


Figure 6.9: Size invariant;mobileNet: PASCAL VOC 2012 = 83.1, mobileNet: atWork_binary = 83.26, xception: PASCAL VOC 2012 = 91.19, xception: atWork_binary = 92.14

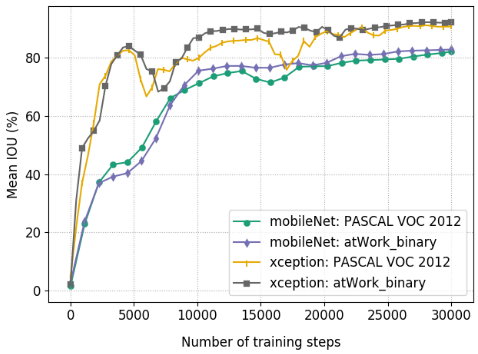


Figure 6.10: Similar shapes;mobileNet: PASCAL VOC 2012 = 82.1, mobileNet: atWork_binary = 82.8, xception: PASCAL VOC 2012 = 90.81, xception: atWork_binary = 92.15

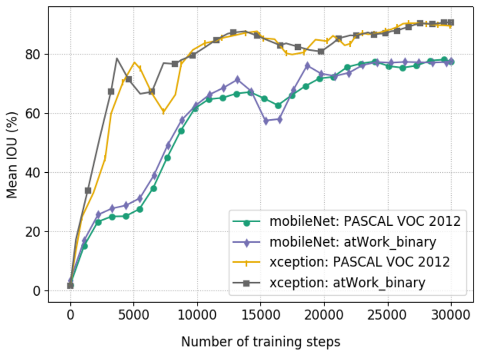


Figure 6.11: Full;mobileNet: PASCAL VOC 2012 = 77.47, mobileNet: atWork_binary = 77.73, xception: PASCAL VOC 2012 = 89.38, xception: atWork_binary = 90.64

7

Conclusions

7.1 Contributions

7.2 Lessons learned

7.3 Future work

A

Design Details

Your first appendix

B

Parameters

Your second chapter appendix