Sematic Segmentation using Resource Efficient Deep Learning

Naresh Kumar Gurulingan,
M. Sc. Deebul Nair, Prof. Dr. Paul G. Plöger

Hochschule Bonn-Rhein-Sieg

November 28, 2018

Hochschule
Bonn-Rhein-Sieg
University of Applied Sciences

Table of Contents

Table of Contents

Semantic segmentation

Divide an input image into different regions which contain a desired object or background.

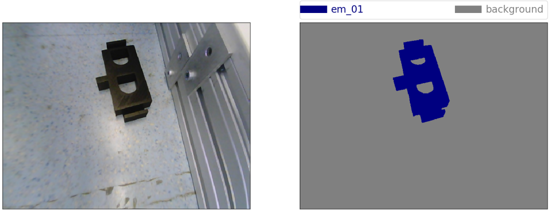


Figure 1: Left: Input image; Right: Segmentation result.

Applications

a Autonomous cars
b Robotics

(a) Street scene [1]

(b) Indoor scene [2]

Table of Contents

Dataset

Objects in the dataset

- First row from left: distance_tube, m20, bearing, axis, r20, m30, m20_100, motor, bearing_box_ax16, bearing_box_ax01, f20_20_B, f20_20_G.
- Second row from left: em_01, s40_40_B, s40_40_G, em_02, container_box_red, container_box_blue.

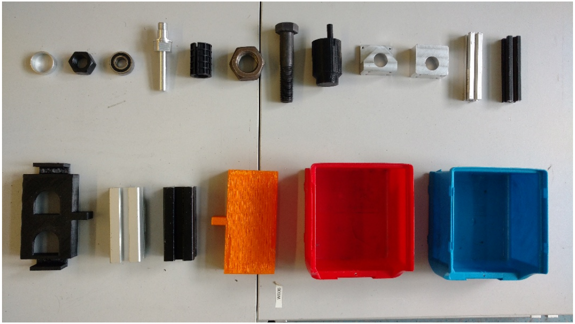


Figure 3: 18 objects in the dataset.

Annotation process

MATLAB ImageLabeler

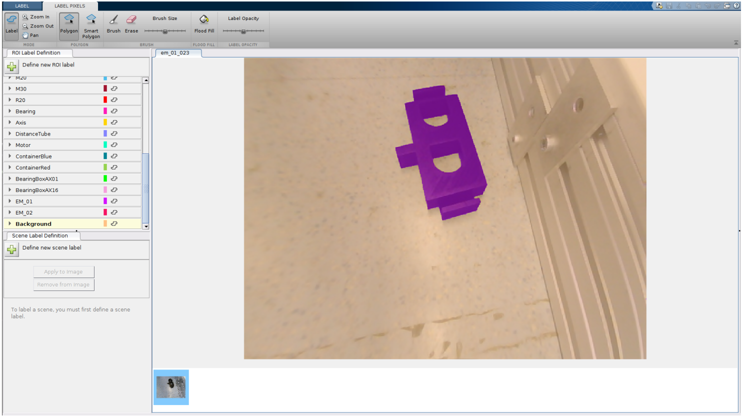


Figure 4: A sample object being labeled in ImageLabeler.

Artificial image generation

Motivation

- For an image containing 1 desired object, roughly 4 minutes was spent for manual annotation.
- Capturing diverse real-world variations is time consuming.

Process

- Collect RGB intensity values of objects using manual annotation.
- Create a list of all the collected objects.
- For an artificial image, select a background image and random objects from the list of objects.
- Place the selected objects at random locations and at random scales.
- Correspondingly generate semantic labels and object detection labels.

Artificial image generation

Sample result

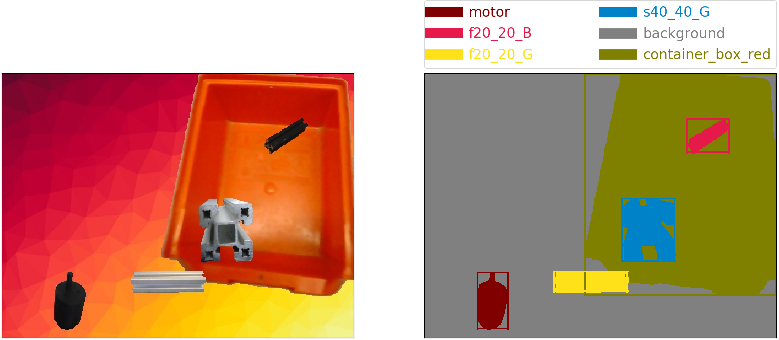


Figure 5: Sample results produced by the artificial image generation algorithm.

Dataset variants

Motivation

- Inability to distinguish size.

(a) m20 [3]

(b) m30 [3]

Figure 6: m30 is larger than m20

- Inability to distinguish shape.

(a) ax16

(b) ax01

(c) ax16

(d) ax01

Figure 7: Bearing_box ax16 and ax01 are distinguishable in a viewpoint from the side ((a), (b)) but are indistinguishable in a viewpoint from the top ((c), (d)).

Dataset variants

Number of classes

Variant	Number of classes including background
atWork_full	19
atWork_size_invariant	15
atWork_similar_shapes	13
atWork_binary	2

Table 1: Number of classes in dataset variants.

Number of images

	Training	Validation	Test
Real Images	396	72	69
Artificial Images	7104	870	870
Total Images	7500	942	939

Table 2: Number of images in the dataset.

Dataset analysis

- Percentage of pixels of an object $= \frac{NP_o}{NP_s}$.
  $NP_o =$ Number of pixels occupied by the object in the training set.
  $NP_s =$ Total number of pixels in the training set.

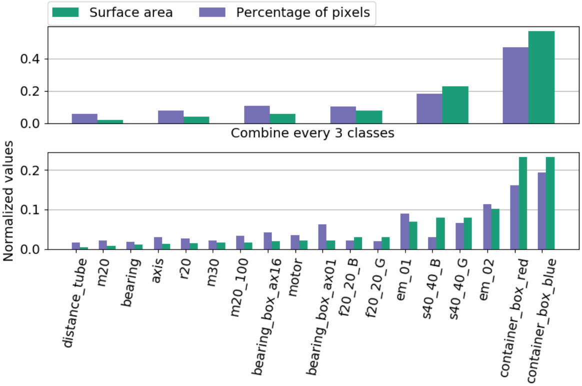


Figure 8: Percentage of pixels vs corresponding real-world surface area

Table of Contents

Local and Global context

- Two different objects are similar in local context.
- Global context required to classify pixels.

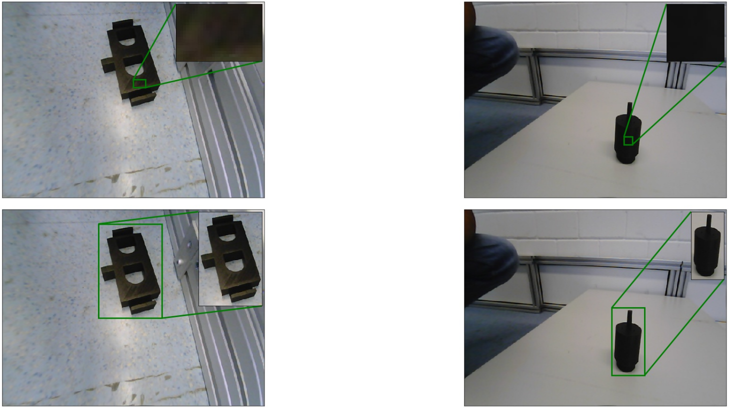
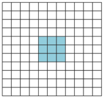


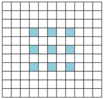
Figure 9: Row 1: local context, row 2: global context.

Atrous convolution and ASPP

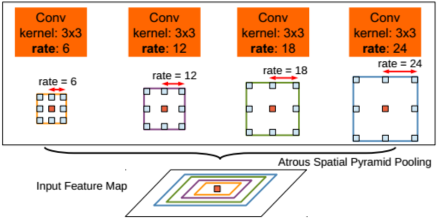
- Increasing the atrous rate (r) increases the receptive field.
- Atrous spatial pyramid pooing (ASPP) on the same input feature map to gather different contexts.



(a) r = 1



(b) r = 2



(c) ASPP

Figure 10: 3×3 atrous convolution with three different atrous rates [4]. Atrous spatial pyramid pooling (ASPP) and atrous convolution [5].

Architecture of DeepLabv3+

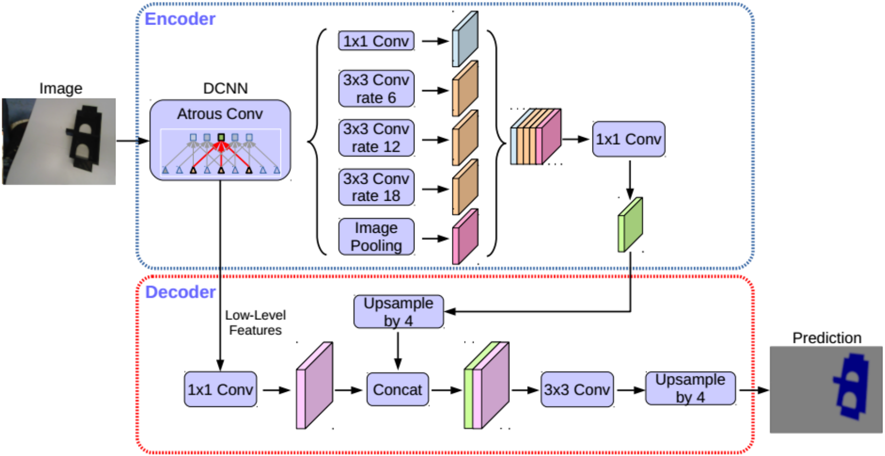


Figure 11: An illustration of DeepLabv3+ architecture. The encoder extracts features at different scales and the decoder refines object boundary delineation [6].

Depthwise seperable convolutions

- First depthwise convolution, then pointwise convolution.
- Row 2 to row 5: depthwise convolution, 6th row: pointwise convolution.

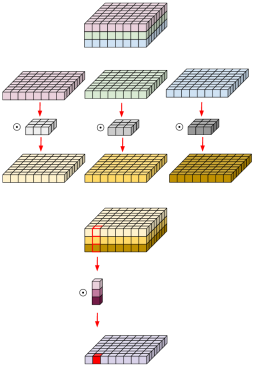


Figure 12: Depthwise seperable convolution [7].

MobileNetv2 and Xception

- Both MobileNetv2 and Xception use depthwise separable convolutions.
- Module: An arrangement of separable convolutions and pooling operations.
- The two networks use different modules.
- Differ in the number of trainable parameters.
- DeepLabv3+ with MobileNetv2 has 2.11 Million trainable parametres and with Xception has 41.05 Million trainable parameters.
- With the Xception encoder, DeepLabv3+ has a higher learning capacity.

Table of Contents

Comparing encoders

- Per class IOU $= \frac{ground\_truth \cap prediction}{ground\_truth \cup prediction}$
- mIOU = mean of all class IOUs.
- DeepLabv3+ with Xception encoder achieves higher mIOU on all four dataset variants.

Dataset variant	**mIOU in %**	
	MobileNetv2	Xception
atWork_full	77.47	89.63
atWork_size_invariant	83.10	92.47
atWork_similar_shapes	82.10	90.71
atWork_binary	96.06	98.68

Table 3: This table lists the mIOU obtained by DeepLabv3+ with MobileNetv2 and Xception encoders on 4 dataset variants.

Comparing dataset variants

- Background/foreground segmentation leads to the highest mIOU.
- Treating all objects as different classes leads to the lowest mIOU.
- Combining objects similar in shape, size or color improves mIOU.

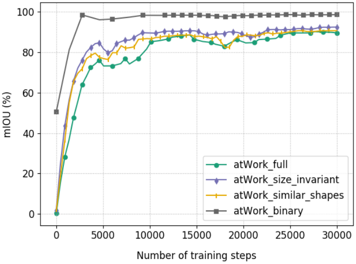


Figure 13: mIOU (%) vs Number of training steps
DeepLabv3+ with Xception encoder on all dataset variants

Comparing individual classes

- 9.88 % of pixels belonging to m30 is classified as m20.
- 0.58 % of pixels belonging to m20 is classified as m30 [Slide 11].

	background	f20_20_B	s40_40_B	f20_20_G	s40_40_G	m20_100	m20	m30	r20	bearing_box_ax01	bearing	axis	distance_tube	motor	container_box_blue	container_box_red	bearing_box_ax16	em_01	em_02
background	99.47	0.06	0.02	0.03	0.01	0.02	0.05	0.02	0.02	0.03	0.01	0.03	0.02	0.05	0.03	0.02	0.02	0.06	0.03
f20_20_B	3.39	91.80	2.33	0.17	0.05	0.23	0.04	0.01	0.04	0.00	0.00	0.30	0.00	1.37	0.05	0.08	0.00	0.06	0.09
s40_40_B	2.21	5.44	76.27	0.04	0.03	0.02	0.00	0.00	0.59	0.00	0.00	0.00	0.00	13.50	0.05	0.00	0.06	1.76	0.01
f20_20_G	17.13	0.36	0.02	59.28	21.26	0.03	0.00	0.00	0.00	0.00	0.07	0.84	0.00	0.00	0.08	0.07	0.75	0.01	0.10
s40_40_G	3.05	0.03	0.01	8.85	83.35	0.00	0.00	0.05	0.00	0.00	0.00	0.17	0.01	0.00	0.02	0.01	0.29	4.15	0.00
m20_100	4.29	0.28	0.13	0.00	0.00	85.82	0.25	0.65	0.08	0.01	0.00	0.11	0.02	7.94	0.07	0.04	0.01	0.27	0.03
m20	4.28	0.37	0.05	0.01	0.00	0.74	92.63	0.58	0.74	0.00	0.00	0.00	0.01	0.40	0.08	0.00	0.00	0.01	0.09
m30	4.11	0.03	0.00	0.00	0.02	0.04	9.88	83.64	0.14	0.12	0.01	0.01	0.04	1.62	0.06	0.00	0.03	0.18	0.06
r20	1.76	0.04	0.04	0.01	0.00	0.18	0.08	0.24	88.89	0.08	0.00	0.06	0.14	8.28	0.01	0.03	0.03	0.12	0.01
bearing_box_ax01	9.05	0.00	0.04	0.00	0.00	0.02	0.00	0.02	0.02	89.62	0.00	0.07	0.01	0.02	0.10	0.01	0.60	0.34	0.08
bearing	7.34	0.10	0.00	0.06	0.00	0.06	1.11	0.00	0.02	0.00	90.89	0.03	0.13	0.09	0.03	0.02	0.00	0.03	0.09
axis	7.28	0.52	0.02	0.98	0.00	0.00	0.01	0.02	0.00	0.05	0.14	86.14	0.37	3.78	0.13	0.17	0.25	0.09	0.04
distance_tube	41.18	0.02	0.00	0.07	0.10	0.00	0.10	0.01	0.00	0.06	0.00	0.15	57.36	0.00	0.00	0.00	0.93	0.00	0.02
motor	1.16	0.67	0.23	0.00	0.01	0.68	0.17	0.15	1.27	0.00	0.00	0.09	0.01	95.17	0.04	0.04	0.00	0.27	0.05
container_box_blue	2.25	0.03	0.02	0.04	0.01	0.03	0.04	0.02	0.01	0.02	0.01	0.03	0.00	0.07	97.21	0.03	0.00	0.13	0.04
container_box_red	3.78	0.04	0.00	0.01	0.03	0.03	0.01	0.00	0.04	0.02	0.01	0.03	0.00	0.17	0.33	95.33	0.05	0.06	0.07
bearing_box_ax16	3.51	0.01	0.00	0.05	0.08	0.12	0.02	0.01	0.01	9.97	0.00	0.05	1.52	0.02	0.06	0.07	84.25	0.25	0.02
em_01	4.11	0.17	0.11	0.02	0.01	0.66	0.12	0.07	0.43	0.05	0.00	0.01	0.00	0.61	0.11	0.02	0.01	93.43	0.06
em_02	2.53	0.07	0.04	0.04	0.00	0.04	0.04	0.03	0.03	0.07	0.01	0.01	0.09	0.14	0.10	0.17	0.00	0.05	96.53

Figure 14: Confusion matrix on atWork_full dataset variant by DeepLabv3+ with MobileNetv2 encoder.

Comparing individual classes

- Class IOU shows an increasing trend with increase in Percentage of pixels.
- Percentage of pixels is shown to increase with surface area [Slide 13].
- DeepLabv3+ tends to learn larger objects first.

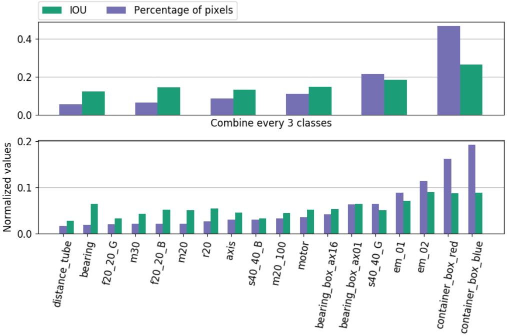


Figure 15: Individual class IOUs achieved by DeepLabv3+ with MobileNetv2 encoder is plotted with the percentage of pixels.

Quantizing the inference graph

- Common operations such as convolution and pooling are replaced by quantized equivalents.
- The inputs to quantized operations are converted to 8 bit.
- With MobileNetv2 encoder, 67 % drop in occupied disk memory is achieved. Drop in mIOU is around 9 %.
- With Xception encoder, 73 % drop in occupied disk memory is achieved. Drop in mIOU is around 2 %.

Quantizing the inference graph

Encoder	mIOU (%)	Number of parameters	FLOPS	Disk memory (MB)
MobileNetv2	84.66	2.11M	6.41B	8.7
MobileNetv2-8	75.17	2.11M	328.87M	2.8
Xception	92.42	41.05M	126.27B	165.6
Xception-8	90.4	41.05M	1.94B	44.7

Table 4: This table summarizes the average mIOU across all four dataset variants, number of parameters, and floating point operations (FLOPS) of both the quantized and full precision encoders of DeepLabv3+. "M" denotes million and "B" denotes billion.

Table of Contents

Conclusion and future work

Contributions

- Artificial image generation algorithm.
- Segmentation dataset with 18 atWork objects.
- Evaluation of DeepLabv3+ with resource efficient encoders MobileNetv2 and Xception.

Future work

- Model interpretability.
- Architecture search.
- Fusion of 2D image data with point cloud information.

Acknowledgements

- MATLAB ImageLabeler. URL: https://de.mathworks.com/help/vision/ref/imagelabeler-app.html
- 20 different colors. URL: https://sashat.me/2017/01/11/list-of-20-simple-distinct-colors/
- Google image downloader. URL: https://github.com/hardikvasa/google-images-download/
- Surface area from 3D CAD models. URL: https://github.com/robocup-at-work/models and https://github.com/rockin-robot-challenge/at_work_models
- HBRS latex beamer template. URL: https://git.fslab.de/mmklab/latex-templates/tree/master/presentation

References

[1] C. dataset. Examples of fine annotations. Online accessed: 2018-08-03. URL: https://www.cityscapes-dataset.com/examples/#fine-annotations.

[2] N. Silberman. NYU Depth Dataset V2. Online accessed: 2018-08-03. URL: https://cs.nyu.edu/~silberman/datasets/nyu_depth_v2.html.

[3] J. Carstensen et al. RoboCup@Work Rulebook Version 2016. Online accessed: 2018-08-03. URL: http://www.robocupatwork.org/download/rulebook-2016-01-15.pdf.

[4] A. Garcia-Garcia et al. "A Review on Deep Learning Techniques Applied to Semantic Segmentation". In: CoRR abs/1704.06857 (2017). arXiv: 1704.06857. URL: http://arxiv.org/abs/1704.06857.

[5] L. Chen et al. "DeepLab: Semantic Image Segmentation with Deep Convolutional Nets, Atrous Convolution, and Fully Connected CRFs". In: CoRR abs/1606.00915 (2016). arXiv: 1606.00915. URL: http://arxiv.org/abs/1606.00915.

[6] L. Chen et al. "Encoder-Decoder with Atrous Separable Convolution for Semantic Image Segmentation". In: CoRR abs/1802.02611 (2018). arXiv: 1802.02611. URL: http://arxiv.org/abs/1802.02611.

[7] E. Bendersky. Depthwise separable convolutions for machine learning. Online accessed: 2018-08-03. URL: https://eli.thegreenplace.net/2018/depthwise-separable-convolutions-for-machine-learning/.

Thank you!

Questions?