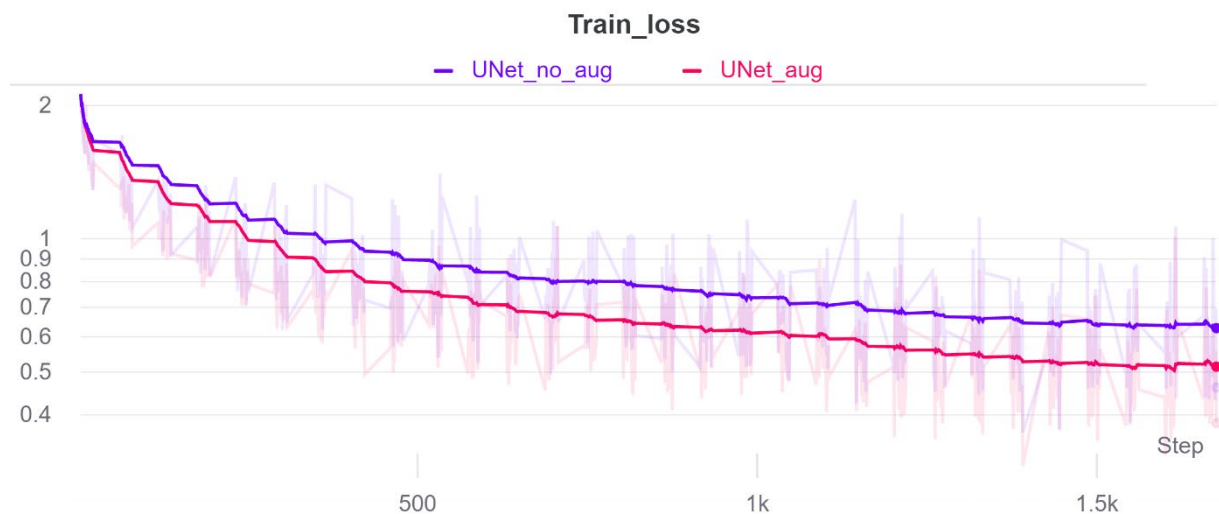
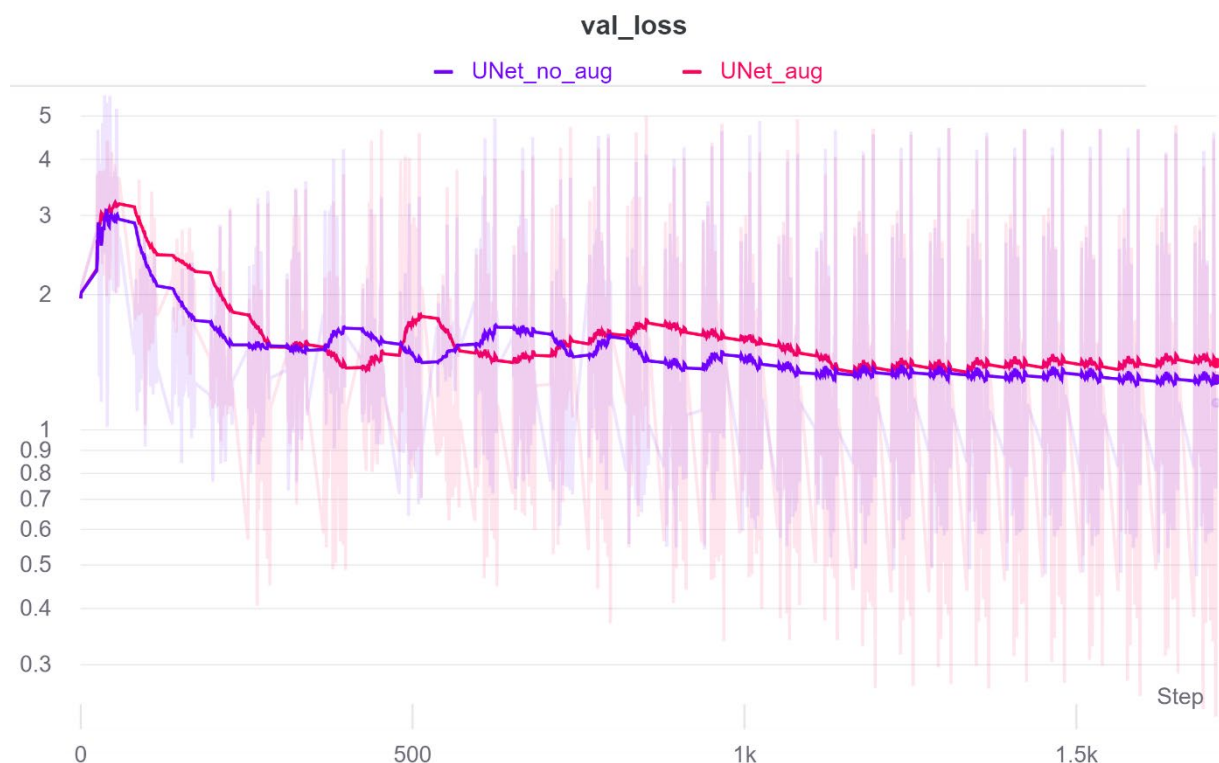


Task 1

In this task, we have to visualize training and validation losses for UNet [1], which was trained with and without data augmentation [2]. According to the result, augmentation improves the quality of training and worsens it for validation. However, it was expected to get a vice versa result because usually data augmentation is used for the training dataset, and it penalizes our model, while on the validation set, we have improving of the results. Nevertheless, this result was achieved with smoothed loss curves by EMA. We can notice that we have huge oscillations in case of validation loss, and the loss curve for the network with augmentation tends to achieve much fewer loss values.



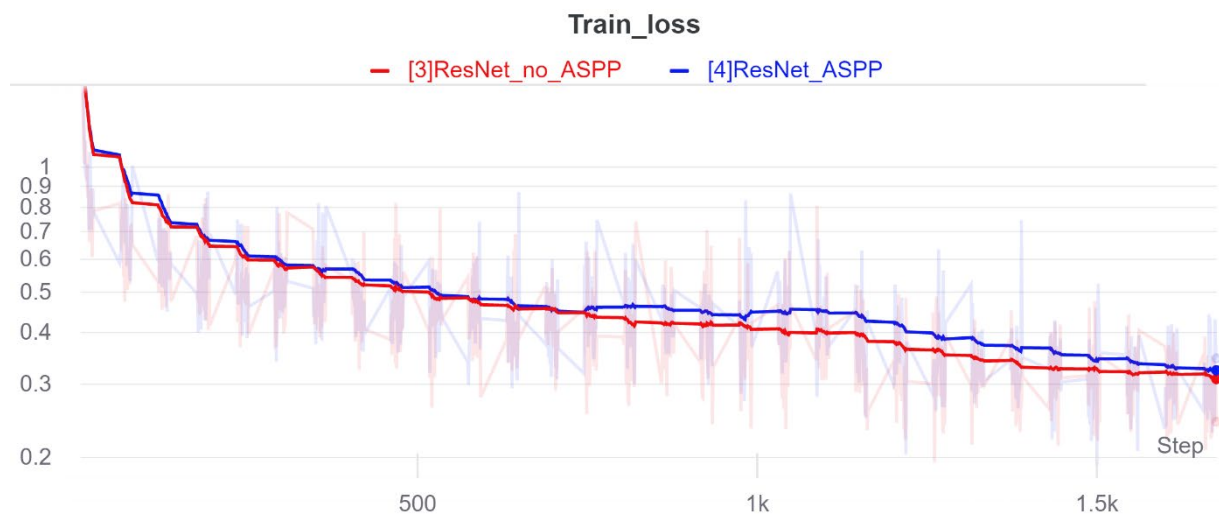
Picture 1. Loss during the training procedure for networks with and without data augmentation



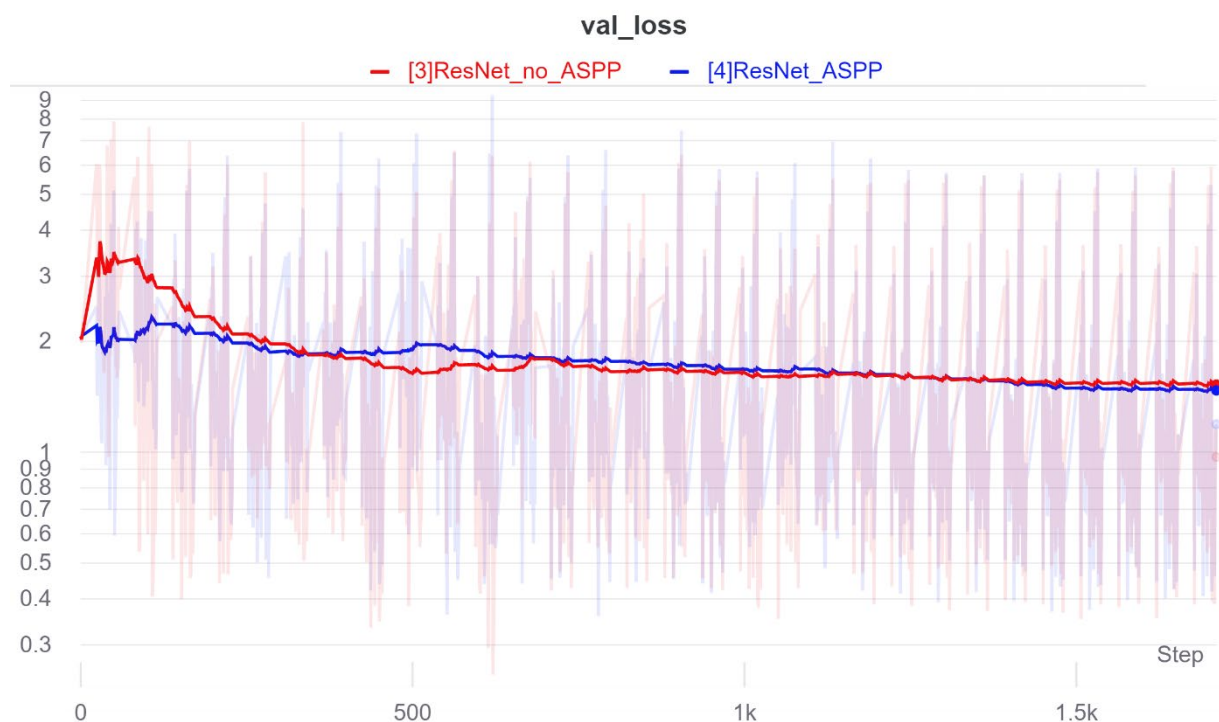
Picture 2. Loss during the validation procedure for networks with and without data augmentation

Task 2

We visualized and compared training and validation losses for DeepLab [3] network with ResNet18 backbone, which was used with and without the ASPP module. According to the result, we can see that ASPP tends to improve the result a bit for the validation set. I think the difference between curves becomes higher after some additional epochs. Also, I noticed that image predictions have a significant difference. Shapes of predictions after ASPP block was smoothed, while network without ASPP had sharp and noisy visualization. The reason of this effect is the fact that in ASPP we use dilated convolutions and upsampling, and therefore shapes become smoother.



Picture 3. Loss during the training procedure for networks with and without the ASPP block



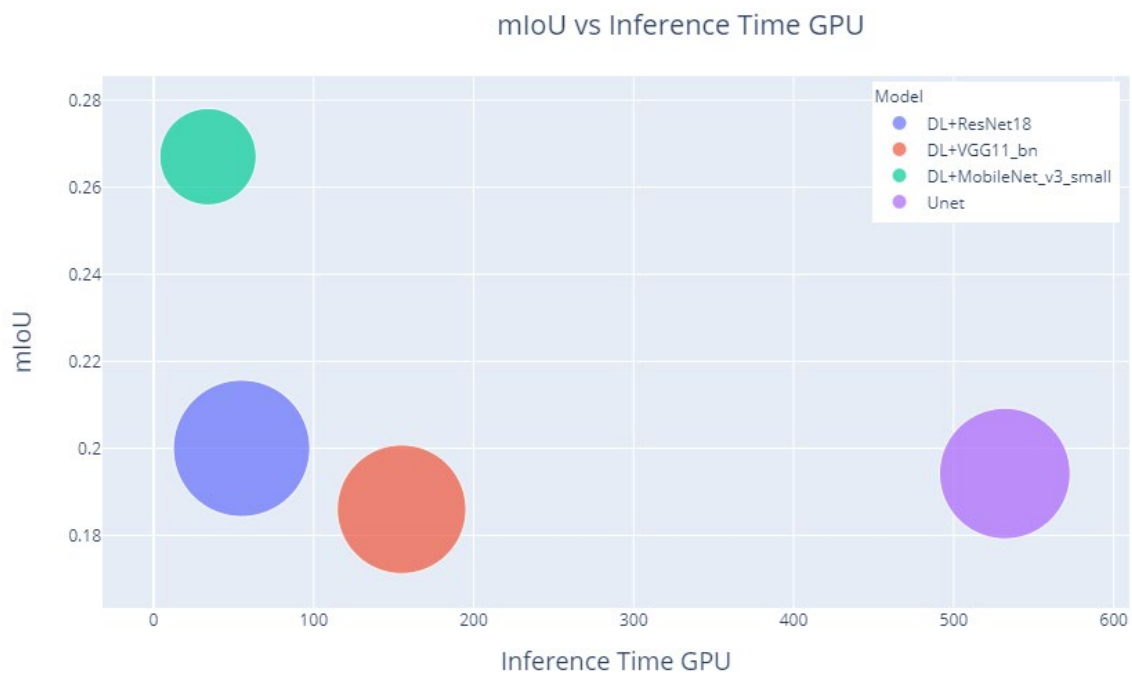
Picture 4. Loss during the validation procedure for networks with and without the ASPP block

Task 3

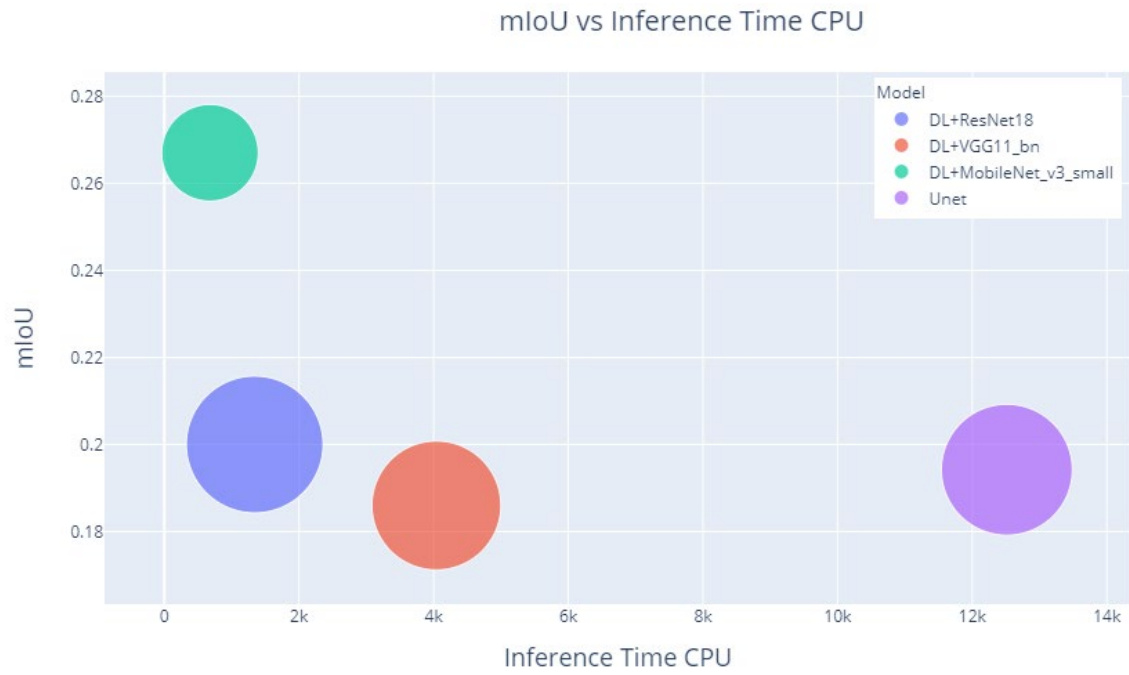
We compared UNet with augmentations and DeepLab with different backbones (ResNet18, VGG11_bn, MobileNet_v3_small). ASPP block also was used for each DeepLab network. The following parameters were compared: training time (in minutes), inference time (in milliseconds), model size with respect to maximal mIoU. We can see that the MobileNet is the best architecture for all criterions: it's the fastest one during the training and inference, it has the best mIoU parameter, and also it has the least size.



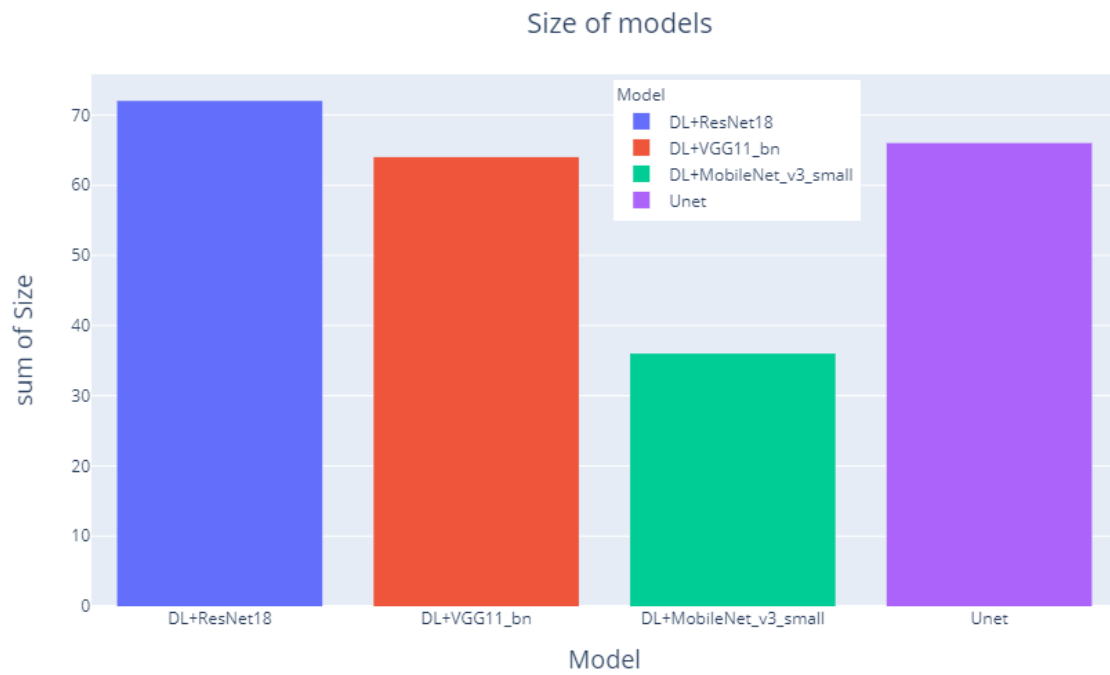
Picture 5. mIoU vs Train time in minutes for different models. Sizes of circles describes size of architecture.



Picture 6. mIoU vs Inference time GPU in milliseconds for different models.



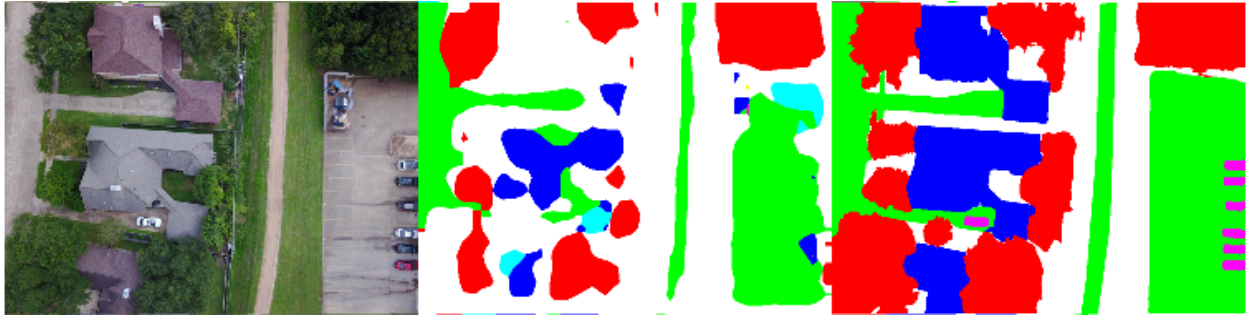
Picture 7. mIoU vs Inference time CPU in milliseconds for different models.



Picture 8. Sizes in MB for different models.

Task 4

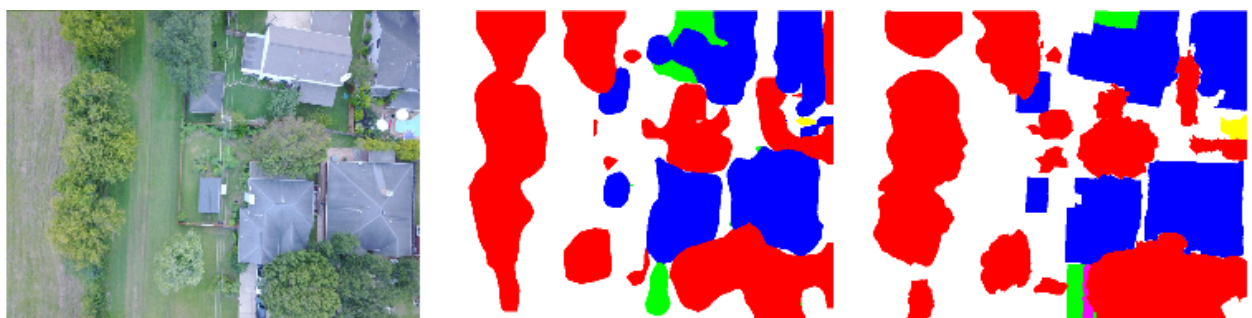
We chose the best model according to the mIoU and compared visualized predictions for good and failure cases. According to the achieved result, MobileNet [5] has the best mIoU metric (0.27). This result was expected [6], using depthwise separable convolutions in residual blocks is very efficient. Let's look at some images.



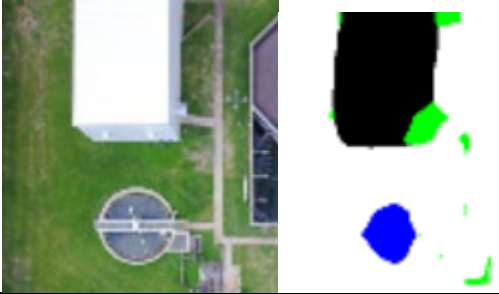

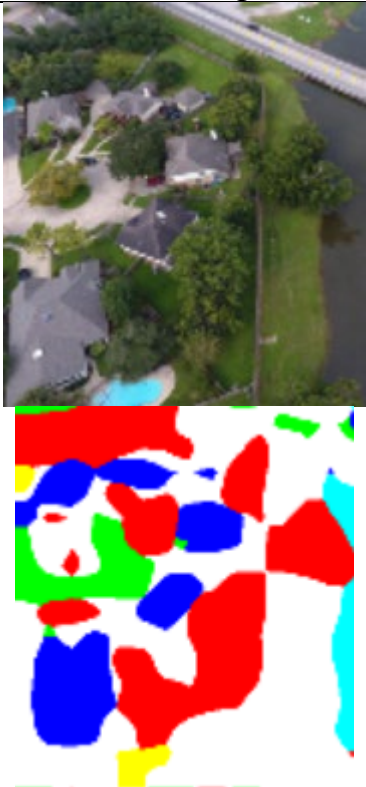
Picture 9. Poorly detection case. Left - original image. Middle - Prediction, Right - Ground truth.





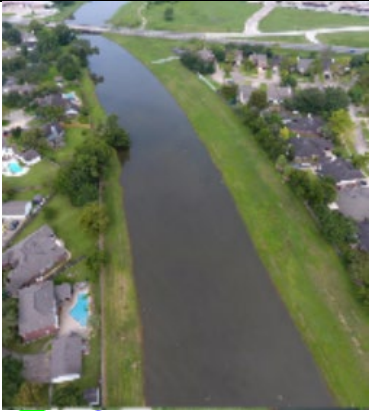



We can notice that our network poorly detects boundaries of objects, especially if they have an unusual shape and blind to small things like cars. The reason for this effect to my mind is using results only from the ASPP module, which has highly dilated convolutions. Although dilated convolutions allow increasing the receptive field of the network significantly, they can hardly distinguish the boundaries of objects. Moreover, the second reason is very strong upsampling in ASPP blocks, which also affects the result this way. In original DeepLab architecture, output data from ASPP block and backbone are concatenated and used in the following convolutions. In our case, we use the only output from ASPP. So I think concatenating with the output of the backbone can improve the result.







However, if objects have usual shapes and big sizes, we have pretty good results of detection. Moreover, we know that the main advantage of MobileNet is speed and small size, that allows to use them on the smartphones and microcomputers.







Picture 10. Good detection case. Left - original image. Middle - Prediction, Right - Ground truth

Background (Black)	
Good example	Bad example
None	
Usually, net predicts background for very unusual objects, that has for example pure white color	
Building (Blue)	
Good example	Bad example
	
Big shapes usually are good detected. Boundaries of objects or tiny objects are detected badly	

Road (Green)	
Good example	Bad example
 	 
Usually good predictions, except boundaries, that can merge	
Water (Cyan)	
Good example	Bad example
 	 
Usually detected very good, but cases with reflection in water classified in wrong way. Need to increase number of pictures with such reflections to fix this.	

Tree (Red)	
Good example	Bad example
 	<p>None</p>
Always detected very good except boundaries	
Pool (Yellow)	
Good example	Bad example
 	 
Usually, well-detected. Interesting case with car that was detected as a pool.	

Vehicle (Purple)	
Good example	Bad example
<p>None</p>	 
Net is blind for small objects such as cars	
Grass (White)	
Good example	Bad example
 	
Perfectly detected. Because we have it on every picture.	

References

- [1] O. Ronneberger, P. Fischer. U-Net: Convolutional Networks for Biomedical Image Segmentation. Medical Image Computing and Computer-Assisted Intervention (MICCAI), Springer, LNCS, Vol.9351: 234--241, 2015
- [2] A. Buslaev, V. Iglovikov. Albumentations: Fast and Flexible Image Augmentations. 2020
- [3] L. Chen, G. Papandreou. DeepLab: Semantic Image Segmentation with Deep Convolutional Nets, Atrous Convolution, and Fully Connected CRFs. Computer Vision and Pattern Recognition. 2017
- [5] A. Howard, M. Zhu. MobileNets: Efficient Convolutional Neural Networks for Mobile Vision Applications. Computer Vision and Pattern Recognition. 2017
- [6] S. Bianco, R. Cadene. Benchmark Analysis of Representative Deep Neural Network Architectures. Conference on Computer Vision and Pattern Recognition (CVPR). 2018