Terrain Classification from an Aerial Perspective

Sivert Frang Lunsaeter†[1], Yumi Iwashita*[2], Adrian Stoica[2], Jim Torresen[1]
[1]*Department of Informatics*, *University of Oslo*, Oslo, Norway
[2]*Jet Propulsion Laboratory*, *California Institute of Technology*, Pasadena, CA, USA

***Abstract*—Terrain knowledge around unmanned ground vehicles (UGVs) is vital for autonomous navigation. Having global understanding of the surroundings of UGVs is important, although the field of view from UGVs is very limited. Thus, we utilize an aerial vehicle to provide a large terrain map from sequential aerial images. In this paper, we present multiple techniques to accelerate the process of terrain classification so that it can run onboard on the aerial platform. The main techniques used to accelerate the process is a "knowledge distillation" of a deep neural net to a shallower one, and a super pixel implementation. We evaluated our system on Jetson TX1 with actual images collected from a weather balloon which confirmed the effectiveness of the proposed system.**

***Index Terms*—Terrain classification, super-pixel, deep learning**

I. Introduction

For an autonomous unmanned ground vehicle (UGV), information about the terrain around it is vital for autonomous navigation. Imagine an UGV that is able to do exploration and perform work that is not viable to humans because of the uninhabitable and unsustainable environment. In a dynamic or unexplored terrain, ensuring that the UGV has information about the current terrain around it would significantly help to optimize the autonomous navigation capabilities. Especially an UGV driving with high speed may not have enough time to avoid obstacles which are occluded until recently.

To solve this issue, one of the solutions is to support UGVs from aerial platforms [7]. An important function to be achieved by aerial platforms is terrain classification from aerial images, which will be used for several processes including path planning for UGVs and exploration planning of aerial platforms. Softman et al. proposed a laser-based terrain classification of urban environment from an aerial platform [16]. In [4], Delmerico et al. introduced a collaborative search system which consists of an aerial robot and an UGV. A terrain map of the environment is provided by the aerial robot. Jafri et al. mainly focused on a path planning of an UGV, whose cost map is generated from terrain classification results [11]. The above works targeted a relatively narrow environment, where a large terrain map is not generated. Getting a larger terrain map is important to have global understanding of an area where UGVs are and to plan global and efficient path for these.

† This research was conducted while he was an intern student at JPL.
*Corresponding author. Yumi.Iwashita@jpl.nasa.gov

Generation of a large terrain map consists of several steps as follows: (1) terrain classification after taking an aerial image, (2) registration of sequential aerial images, and (3) fusion of terrain classification results of multiple images. Each step, especially steps 1 and 3 are generally computationally expensive to run onboard on an aerial platform, as we explain in the following paragraphs. Therefore, in this paper, we focus on validating candidate techniques to accelerate these two steps. To the best of our knowledge, this is the first study which verifies multiple techniques to speed up the steps for terrain classification.

High-accuracy terrain classification is achieved by Deep Learning approaches [3] [18] [12]. However, in general these approaches are computationally demanding due to its large number of layers in their neural network (NN) architecture. To reduce the computational cost, one idea is to decrease the number of layers in the NN architecture, but still keep as much as possible of the performance of the full NN architecture. This can be achieved by "distilling the knowledge" which is proposed by Hinton et al. [9] [10]. In the distillation process one tries to distill the knowledge trained into a deep model into a more shallow one.

After the 2nd step (image registration), there are some overlapping areas among images. Therefore, in the 3rd step, terrain classification results of multiple images are fused. A naive approach to fuse multiple images is voting of terrain classification results at each pixel, which results in high calculation cost. To solve this issue, we apply a super-pixel approach (e.g. SLIC, Simple Linear Iterative Clustering [1]) to an aligned large image, which categorize multiple neighboring pixels with similar value in an image into super pixels. This dramatically reduces the computational cost. There are several existing works [5] which use the super-pixel approach to accelerate the terrain classification process, but there is none we are aware of which applies a super-pixel approach to fuse terrain classification results.

Our paper is organized as follows. Section II describes the system overview, acceleration of terrain classification, and fusion of multiple terrain classification results of sequential images. Section III explains a collected data set, implementation details, and experimental evaluations of the approaches. Finally, section IV presents the conclusions and discusses future works.

II. Generating large terrain map from sequential aerial images

A. System overview

First, we explain the system overview for generating a large terrain map from sequential aerial images. As we explained in the introduction, there are three steps ((1) terrain classification, (2) image registration, and (3) fusion of terrain classification results of multiple images). While time-series images are taken from an aerial platform, these three steps are applied sequentially to each image (1st and 2nd steps can be applied at the same time if there are multiple computing resources). We will explain more details about the 1st and 3rd steps and how we accelerate these in the following sections. When the 2nd step is applied, GPS information of the aerial platform is used as an initial position if it is available. The registration process is done by estimating a ***homography*** matrix between images. This process creates a larger terrain map with some overlap regions between images.

B. Acceleration of terrain classification

There are multiple DL approaches for terrain classification. In this study, we use DeepLab [2], and techniques which we explain in this section can be applied to any DL approaches. The Deeplab architecture is large, with over a hundred convolutional operations in it. It is a requirement that an onboard device that is to run the algorithm has enough memory. Unfortunately, some of devices such as the Jetson TX1 does not. Besides, as we explained in the introduction, the computational cost depends on the number of layers in DL approaches.

To accelerate the process of terrain classification, there are multiple approaches, such as distilling the knowledge of the DL model and applying a super-resolution approach to images before terrain classification. In this study we focus on the distillation and the super-resolution approach is left as a future work.

The architecture of the DeepLab has 34 residual blocks spread over 4 main parts, where all blocks in the same part has the same dimensions in the NN layers. The first, second, third, and last parts have 3, 4, 23, and 3 blocks, respectively. We modified the DeepLab architecture so that the total number of residual blocks becomes 13, while still maintaining 4 main parts. Each of the first and second parts now has 2 residual blocks, while the third part has 8. The last part has only one. To distill the knowledge from the full DeepLab, first we train the full DeepLab with a training data set. We then train the distilled DeepLab model by calculating a loss function against both annotation information in the training data and predicted values by the full DeepLab. A weight is set to calculate loss values as $loss = \lambda loss_a + (1-\lambda) loss_f$, where $loss_a$ and $loss_f$ are defined as loss functions based on annotation information in the training data and predicted values by the full DeepLab, respectively. In experiments, we set λ = 0.75.

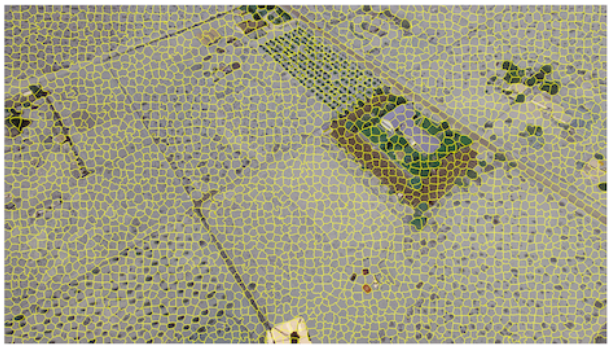

Fig. 1. An example result of SLIC [1]. Yellow boundaries show different patches separated by SLIC.

Fig. 2. An example of captured images.

C. Acceleration of fusion of terrain classification results

In the overlapping area between images, each pixel is classified at least twice. A straight forward approach to fuse overlapping area is to average scores / get max score of terrain classification results at each pixel. There are, however, a costly disadvantage in the cost in time, since the cost increases linearly with the number of pixels in the overlapping area.

To reduce the computational cost, we apply a super-pixel approach, which creates a continuous patch that ideally shares similar features. Specifically we utilized SLIC (Simple Linear Iterative Clustering) [1], and its example result is shown in Fig. 1, whose original image is Fig. 2. Yellow boundaries show different patches separated by SLIC. Shuurmans et al. proposed an approach of a super-pixel-based max-pooling operation [15], and we take advantage of this max pooling operation as follows. After a large map is generated by image registration process in the 2nd step, we apply SLIC to it to generate multiple patches. At each patch, we apply the super-pixel-based max-pooling operation, which results in terrain classification at each SLIC patch.

III. Experiments

In this section we describe the data set, details of implementation, and experimental results.

A. Data set

This work is a part of a weather balloon project at Jet Propulsion Laboratory. Thus, we generated our data set from

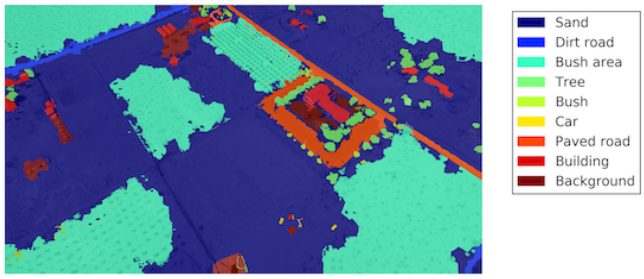

Fig. 3. Annotated image of Fig. 2.

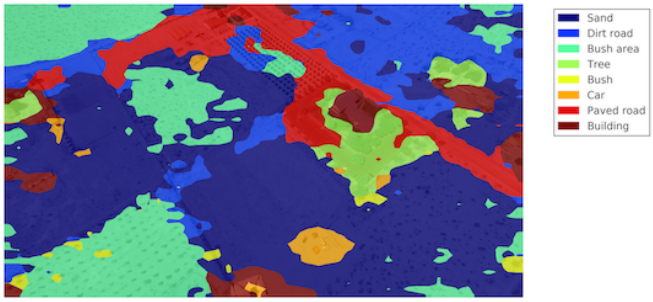

Fig. 4. Terrain classification after reducing the image resolution of full size image. Ground truth is shown in Fig. 3.

images taken during a balloon flight. (Zephyrus VII test flight in July 2018 [6]). An example captured image is shown in Fig. 2. With an image annotator tool developed by Tangseng et al. [17], annotations containing terrain type were created for each image to serve as the ground truth in the training process. The data set differentiates between twelve classes (sand, soil, dirt road, rock, bedrock, tree, bush, bush area, car, person, paved Road, building). Annotated image of Fig. 2 is shown in Fig. 3. Here, image size is 1920 $\times$ 1080 and the number of annotated images is 240. In experiments, 50 % of images are used for training and testing, respectively. Image shown in Fig. 2 is not used in training of the DL model, but used as part of the test data.

B. Implementation details

Both the 1st and 3rd steps for terrain classification and fusion are implemented in PyTorch and run on the GPU. The 2nd step for image registration is implemented in C++, which is called from Python using the Boost framework. All experiments are done on Jetson TX1, which we mounted on a quadrotor drone. (DJI Phantom III).

Calculation cost, accuracy, and memory usage of terrain classification depends on input image resolution into distilled / full DeepLab. The image resolution we use is too large (1920 $\times$ 1080) to load on Jetson TX1. One can simply reduce the image resolution, but the accuracy is very low, as shown in Fig. 4, whose ground truth is shown in Fig. 3. Therefore we split the full image into multiple grids with overlapping areas between grids. Here we set overlapping areas to smooth the gap between grids in the step 3.

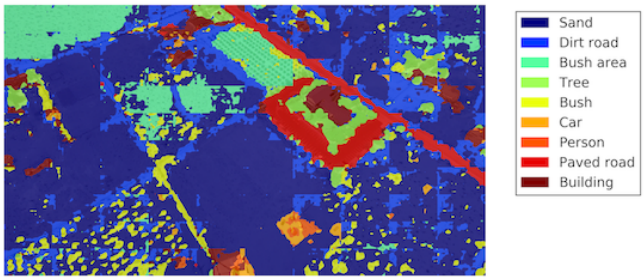

Fig. 5. Grid-based terrain classification result of Fig. 2 without any overlap (setting (i)).

In the following experiments, we used two settings: (1) evaluation of 1st and 3rd steps with a single image and (2) evaluation of the whole steps with sequential images.

C. Evaluation with a single image

In this section we evaluate the acceleration of terrain classification and fusion processes. All experiments are done with images which are divided into 80 grids (8 $\times$ 10 regions). We tested 5 different settings: (i) grid-based approach without overlap, (ii) grid-based approach without overlap but with CRF (conditional random field) [13], [14], (iii) grid-based approach with overlap, (iv) grid-based approach with overlap and CRF, and (v) grid-based approach with super-pixel (our technique). Here, CRF is a popular technique to improve the accuracy of terrain classification. In settings (iii) and (iv), average score of the terrain classification results in overlapping area is calculated in each pixel.

Figures 5 $\sim$ 9 show visualization of terrain classification of Fig. 2 by 5 different settings, respectively. In Fig. 5 (setting (i)), more detailed terrains are classified compared with Fig. 4 (simple rescaling the whole image), but we can clearly see gaps between grids and small noisy results. The CRF approach in Fig. 6 (setting (ii)) removed small noise, but we still see gaps. The overlap-based approach in Figs. 7 (setting (iii)) and 8 (setting (vi)) improved results visually. Finally, super-resolution results in Fig. 9 (setting (v)) show smooth results between grids.

Table I and Fig. 10 show quantitative evaluations of the 5 settings and calculation costs, respectively. From these results, the grid-based approach with averaging score in each pixel and CRF-based approaches (settings (ii) $\sim$ (vi)) have a high calculation cost in time. Super-pixel approach improved the calculation cost a lot and also the terrain classification accuracy compared with setting (i). CRF-based approach shows the best performance, but takes too much time. Overall, setting (v) (grid-based approach with super-pixel) is a reasonable choice to be used by considering the balance between calculation time and accuracy.

D. Evaluation of the system with sequential images

In the final experiment, we used 6 sequential images as shown in Fig. 11. Actual scale in each image is 80 [m] $\times$ 150 [m]. We run all steps with these images. Figure 12

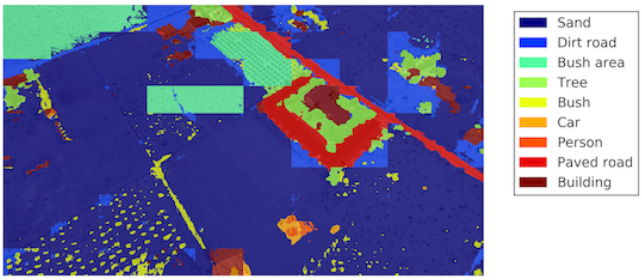

Fig. 6. Grid-based terrain classification result of Fig. 2 without any overlap but with CRF (setting (ii)).

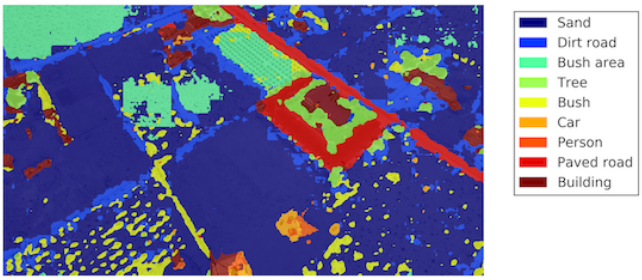

Fig. 7. Grid-based terrain classification result of Fig. 2 with overlap (setting (iii)).

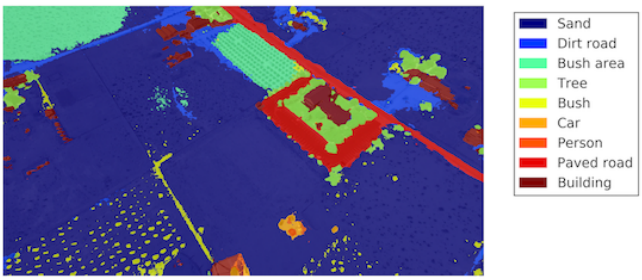

Fig. 8. Grid-based terrain classification result of Fig. 2 with overlap and CRF (setting (vi)).

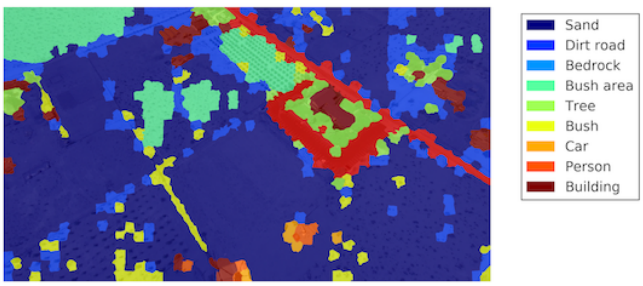

Fig. 9. Grid-based terrain classification result of Fig. 2 with overlap and super-pixel (setting (v)).

TABLE I
EVALUATION RESULTS OF THE DIFFERENT VARIANTS.

Setting	Overall acc	Freq. weight acc
(i) Non-overlapping grids	0.62	0.49
(ii) Non-overlapping grids w. CRF	0.68	0.53
(iii) Overlapping grids	0.64	0.50
(vi) Overlapping grids w. CRF	0.70	0.55
(v) Overlapping grids w. super-pixel	0.64	0.50

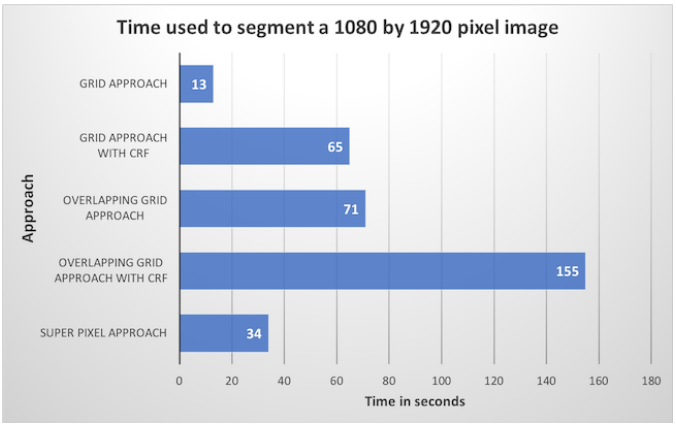

Fig. 10. Calculation cost of the different settings.

shows the registration results, which produces a well aligned image. Figure 13 shows terrain classification results. Although there is some misclassification between bush class and bush area class, the basic information of these two classes are the same. The vast majority of the terrain classification is good. The overall calculation time to process 6 images is 55 [sec] by the overlapping-based super-pixel approach, although overlapping-based CRF approach took 185 [sec], which is too long.

IV. CONCLUSIONS AND FUTURE WORK

In this paper, we proposed a system to accelerate the process of generating a large terrain map from sequential aerial images. In our system we integrated two techniques: (1) "knowledge distillation" to generate much shallower neural network architecture to accelerate the terrain classification process, and (2) integration of super-pixel approach to fuse terrain classification results between overlapping images. The effectiveness of the proposed system is confirmed by the undertaken experiments. The implemented techniques seem to work as well as can be expected, given the limited resources available for computation. Most of the misclassification is between similar classes, such as "sand" and "dirt road", and there is little misclassification between very different classes, such as "building" and "sand". In a large terrain map the distinction between the most different classes is most important, and our experiments show that our system achieve that.

Future work includes applying a super-resolution approach to images before terrain classification. Future work also include comparing the current Deeplab ResNet architecture used to other ResNet architectures, both shallow and deep.

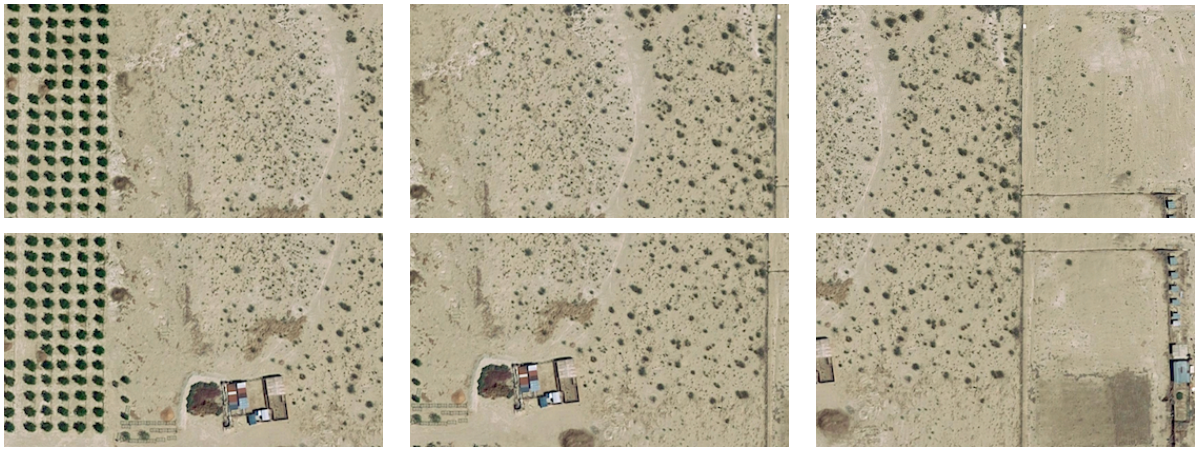

Fig. 11. The six images used in the 2nd experiment.

Fig. 12. Image registration results from 6 images (Fig. 11).

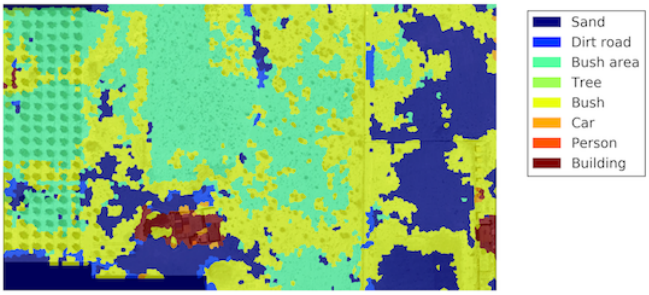

Fig. 13. Terrain classification and fusion results by grid-based super pixel.

ACKNOWLEDGMENT

The research described in this paper was carried out at the Jet Propulsion Laboratory, California Institute of Technology, under a contract with the National Aeronautics and Space Administration. Government sponsorship acknowledged. This work was partly supported by the Research Council of Norway and Diku project Collaboration on Intelligent Machines (COINMAC) project, under grant agreement 261645.

REFERENCES

[1] R. Achanta et al, 'SLIC Superpixels Compared to State-of-the-Art Superpixel Methods', IEEE Transactions on Pattern Analysis and Machine Intelligence 34.11, pp. 2274–2282, 2012.

[2] L. Chen et al, 'DeepLab: Semantic Image Segmentation with Deep Convolutional Nets, Atrous Convolution, and Fully Connected CRFs', IEEE Transactions on Pattern Analysis and Machine Intelligence, pp.834-848, 2017.

[3] L. Chen and et al., 'Semantic image segmentation with deep convolutional nets and fully connected crfs', PAMI, 2017.

[4] J. Delmerico, E. Mueggler, J. Nitsch, and D. Scaramuzza, 'Active Autonomous Aerial Exploration for Ground Robot Path Planning', IEEE Robotics and Automation Letters, 2017.

[5] M. Ghiasi and R. Amirfattahi, 'Fast semantic segmentation of aerial images based on color and texture', Iranian Conference on Machine Vision and Image Processing (MVIP), 2013.

[6] H. Hall et al, 'Project Zephyrus: Developing a rapidly reusable high-altitude flight test platform', IEEE Aerospace Conference, Mar. 2018, pp. 1–17, DOI: 10.1109/AERO.2018.8396809.

[7] E.Harik, F. Guérin, F. Guinand1, J. Brethé, H. Pelvillain, 'UAV-UGV cooperation for objects transportation in an industrial area', IEEE Int. Conf. on Industrial Technology (ICIT), 2015

[8] K. He et al, 'Deep Residual Learning for Image Recognition', IEEE Conf. on Computer Vision and Pattern Recognition (CVPR), 2016.

[9] G. Hinton, O. Vinyals and J. Dean, 'Distilling the Knowledge in a Neural Network', NIPS Deep Learning and Representation Learning Workshop, 2015.

[10] Z. Hu et al, 'Harnessing Deep Neural Networks with Logic Rules', ACL 2016.

[11] M. Haider Jafri, Rahul Kala, 'Path Planning of a Mobile Robot in Outdoor Terrain', S, Intelligent Systems Technologies and Applications pp 187-195, 2015.

[12] Y. Iwashita, K. Nakashima, A. Stoica, and R. Kurazume, 'TU-Net and TDeepLab: Deep learning-based terrain classification robust to illumination changes, combining visible and thermal imagery', IEEE Conf. on Multimedia Information Processing and Retrieval, 2019.

[13] P. Krähenbühl and V. Koltun, 'Efficient Inference in Fully Connected CRFs with Gaussian Edge Potentials', Proc. of the 24th International Conference on Neural Information Processing Systems, pp.109–117, 2011.

[14] J. Lafferty, A. McCallum and F. C N Pereira. 'Conditional Random Fields: Probabilistic Models for Segmenting and Labeling Sequence Data', Int. Conf. on Machine Learning, 2001.

[15] M. Schuurmans, M. Berman and M. B. Blaschko, 'Efficient semantic image segmentation with superpixel pooling', http://arxiv.org/abs/1806.02705.

[16] B. Sofman, J. Bagnell, A. Stentz, and N. Vandapel, 'Terrain Classification from Aerial Data to Support Ground Vehicle Navigation', Carnegie Mellon University, CMU-RI-TR-05-39, 2006.

[17] P. Tangseng, Z. Wu and K. Yamaguchi, 'Looking at Outfit to Parse Clothing', arXiv:1703.01386.

[18] A. Valada, R. Mohan, and W. Burgard, 'Self-supervised model adaptation for multimodal semantic segmentation', International Journal of Computer Vision (IJCV), 2019.