Chapter 1

Results

In this section, we evaluate five convolutional neural networks: the model proposed by Chavez-Garcia et al. **?** and the four MicroResnet presented in the last chapter. We select the best performing one based on the classification score on the test set. We show the results of the same architecture trained on regression instead of classification and prove the latter yields better results. Finally, we qualitatively evaluate the best model by predicting traversability in real-world terrains.

1.1 Quantitative results

1.1.1 Model selection

We compare the four different MicroRenset architectures **??** and the Original CNN **??** propose in the original work **?** on classification. The dataset is labeled as described in sec $sec: label - patches.Werunfiveexperimen$
$SEwithaROCAUCof0.896onthetestset.Wesuggestthatthebigkernelsizeallowsthenetworktoextrapolatemoregou$
$SEgainsahugeboostof0.021comparedtoMicroResNet7x7.$

1.2 Classification results

Table **??** table shows in deep the MicroResNet7x7-SE's performance on different datasets. We obstain an 95% accuracy and 0.96 AUC on the validation set and 88% accuracy and 0.89 AUC on the test set.

1.3 Regression Results

Using MicroResNet7x7-SE as regressor also yield decent results. The model scores a MSE of 0.006 on the validation and 0.020 on the test set. Table **??** shows in detail the score for each dataset. Figure **??** plots the real advancement against the regressor output for the test set. In most cases the model did not correctly predict the advancement. The regressor has several advantages compared to the classifier, First, since we can use the same model to classify patches with a different threshold. This is done by seeing if the predicted advancement is greater or lower than a threshold. Regression is a usually feasible solution. For example, if we need to consider multiple thresholds. Classification

Dataset		Samples	MicroResNet7x7-SE		Size(m)	Resolution(cm/px)
Type	Name		ACC	AUC		
Synthetic	Training	429312	-	-	10×10	2
	Validation	44032	95.2 %	0.961	10×10	2
	Arc Rocks	37273	85.5 %	0.888	10×10	2
Real evaluation	Quarry	36224	88.2 %	0.896	32×32	2

Table 1.2. Classification results of MicroResNet7x7-SE on different datasets.

Dataset		Samples	MicroResNet7x7-SE	Size	Resolution(cm/px)
Type	Name		MSE		
Synthetic	Training	-	-	10×10	2
	Validation	44032	0.006	10×10	2
	Arc Rocks	37273	0.020	10×10	2
Real evaluation	Quarry	36224	0.022	32×32	2

Table 1.3. Regression results of MicroResNet7x7-SE on different datasets.

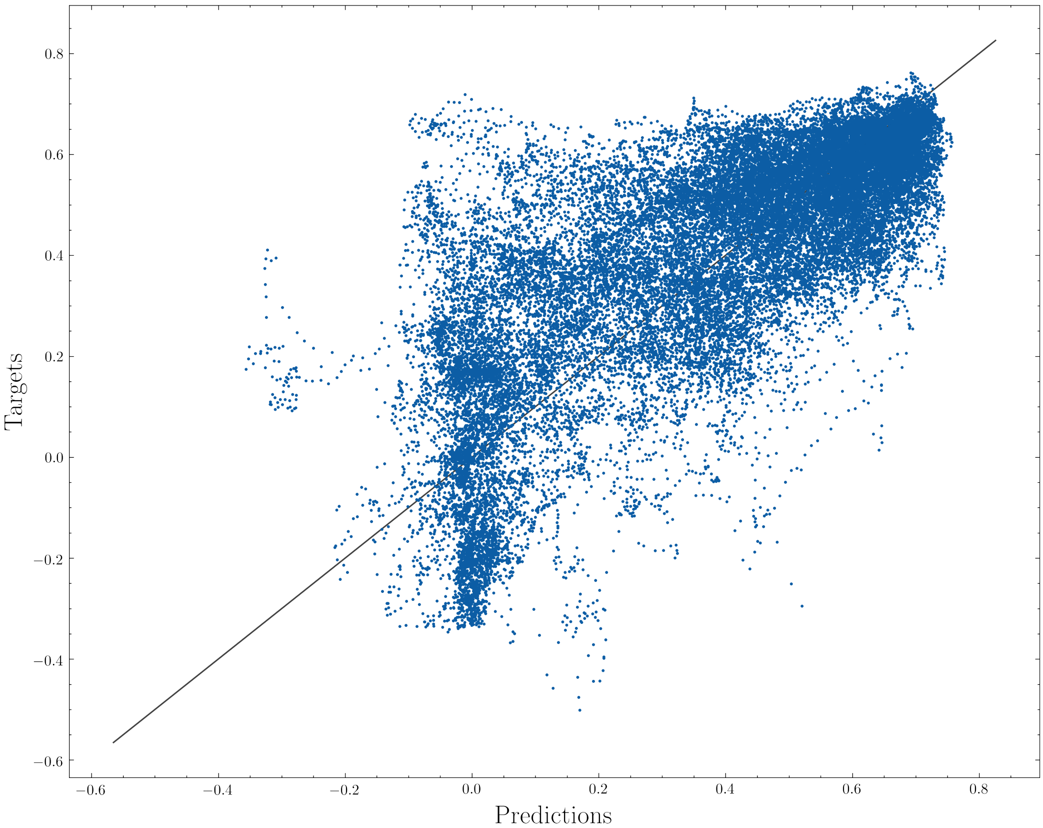


Figure 1.1. RReal advancement against the regressor output for the test set. Most of the points are far away to the diagonal line showing that the regressor predicts the wrong advancement in many cases.

requires the training of one classifier per threshold while regression can evaluate always the same model.

However, if we fixed a threshold, the same exactly architecture trained directly to predict traversability outperforms the regressor. Table **??** shows the accuracy of two MicroResNet7x7-SE on classification and regression after a threshold of twenty centimeters is selected. The classifier outperformed the regressor. For this reason, we decided to evaluate the MicroResNet7x7-SE classifier. We evaluate the classifier predictions by plotting the traversability probability on different

	Regression	Classification
	ACC	
Top	72.8%	**88.2%**
Mean	73.6%	**87.8%**

Table 1.4. Regression and classification accuracy for the same model, MicroResNet7x7-SE, on a threshold of 20cm. The regression accuracy is computed using its output labeled with the selected threshold and the binary targets from the classification dataset. This mimic the situaion when the fixed a miniuim advancement for the robot.In this case, the classifier outperforms the regressor.

maps in 3D. We used a sliding window to extract the patches to generate the patches. Then, we created a texture based on the traversable probability. For each map, we evaluated four headings. subsectionQuarry We first evaluate the test set composed by the quarry map 32×32m and with a maximum height of 10m. We expect the trail on the slopes to be traversable at almost any headings, especially when Krock move from left to right and vice-versa. The top part should be hard to traverse in almost any case. Figure **??** shows the traversability probability directly on the map.

some where place the colormap bar

Correctly, the lower part of the map, composed by flat regions, is labeled with high confidence as traversable in all rotations. On the other hand, the traversability of the slopes and the bumps on the top region depends on the robot orientation.

1.3.1 Bars

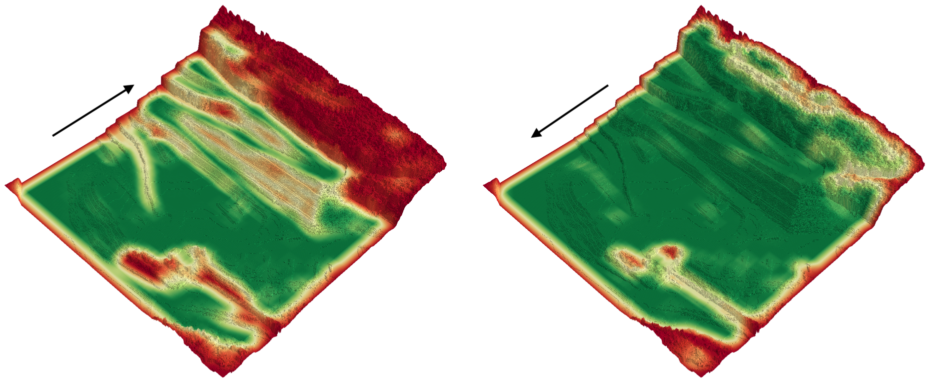
Bars is a map composed of walls with different heights, thus we expected to be mostly not traversable at different headings. Figure **??** shows the traversability probabilities on the terrain.

Due to the high number of not traversable walls, this is a hard map to traverse the robot . Interesting, we can identify a corridor near the bottom center of the maps. This region shows how the model correctly label those patches depending on the orientation. Figure **??** highlights this detail.

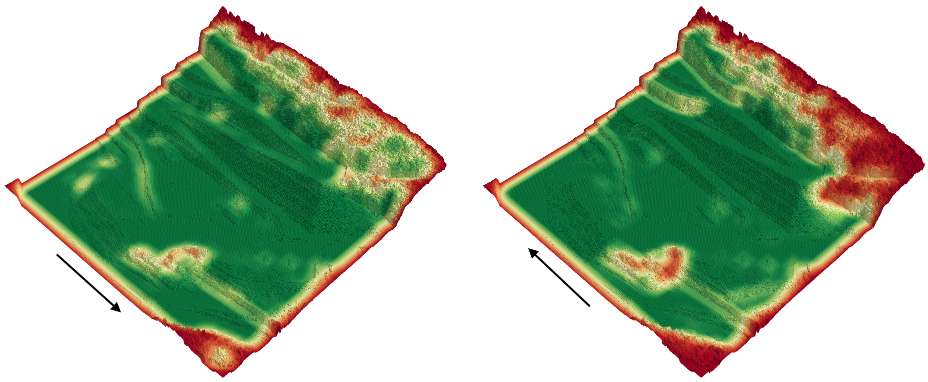
1.3.2 Small village

We apply the same procedure to evaluate the network on a 10×10m small village map. Figure **??** describes its traversability for four different rotations.

All the streets are labeled as traversable, while some buildings' roofs (e.g. the church's one) are traversable depending on the Krock orinetation. For example, most steep roofs are not traversable

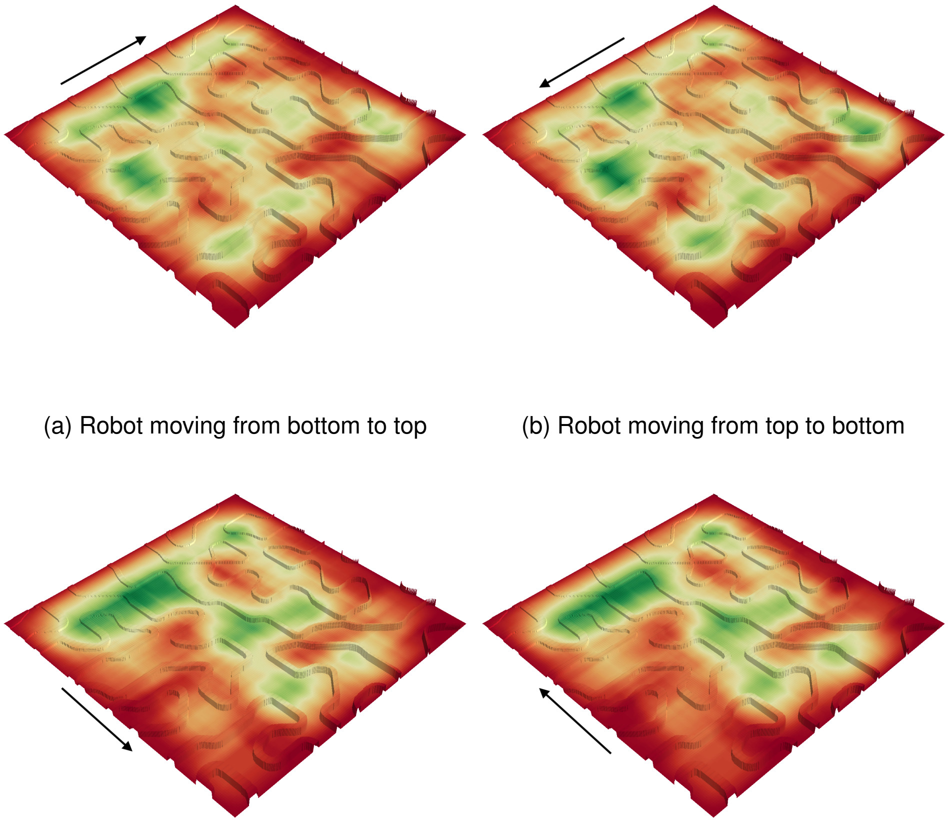


(a) Robot moving from bottom to top (b) Robot moving from top to bottom



(c) Robot moving from left to right (d) Robot moving from right to left

Figure 1.2. Traversability probability on the test set, a quarry 32×32m and a maximum height of 10m, for different Krock's orientations.



(a) Robot moving from bottom to top

(b) Robot moving from top to bottom

(c) Robot moving from left to right

(d) Robot moving from right to left

Figure 1.3. Traversability probability on the bars map, a 10×10m , for different Krock's orientations.

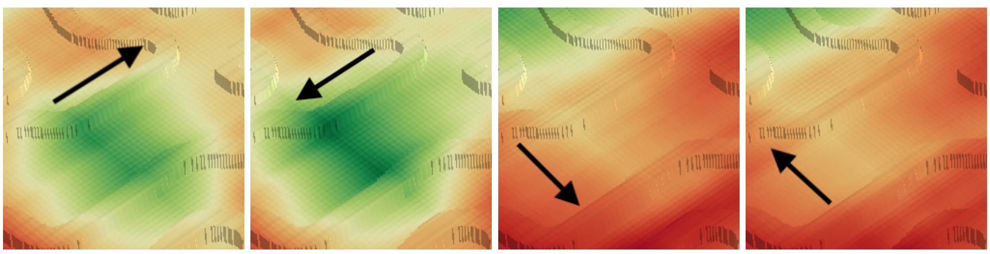
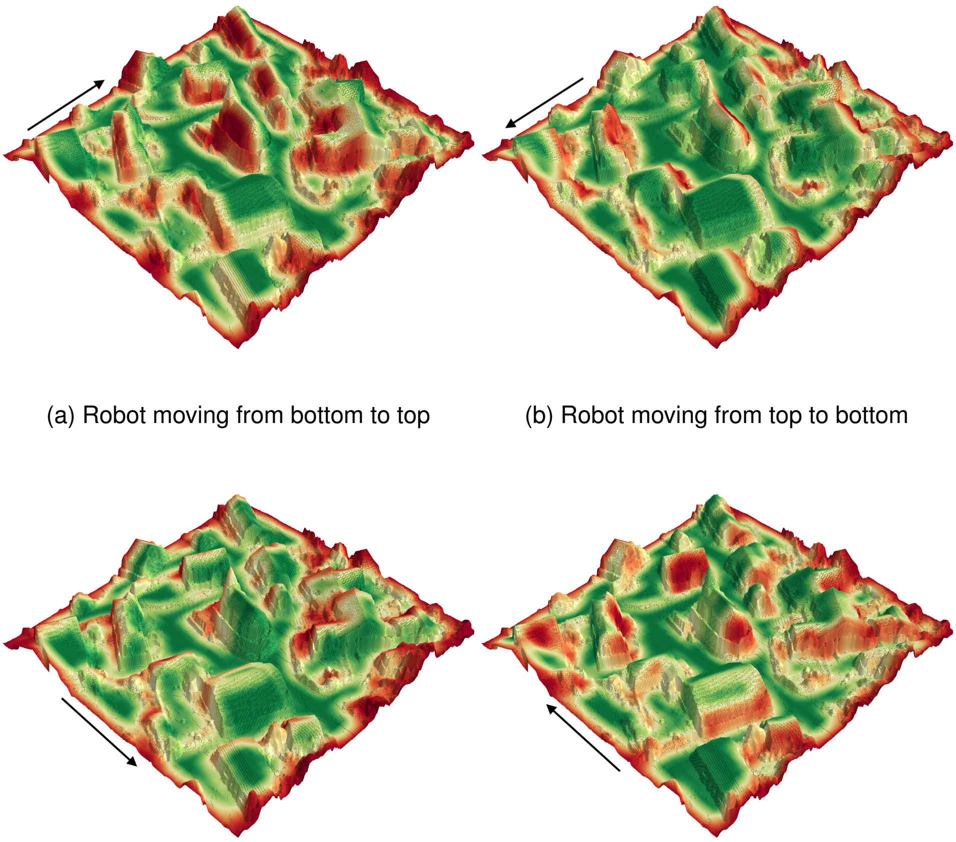


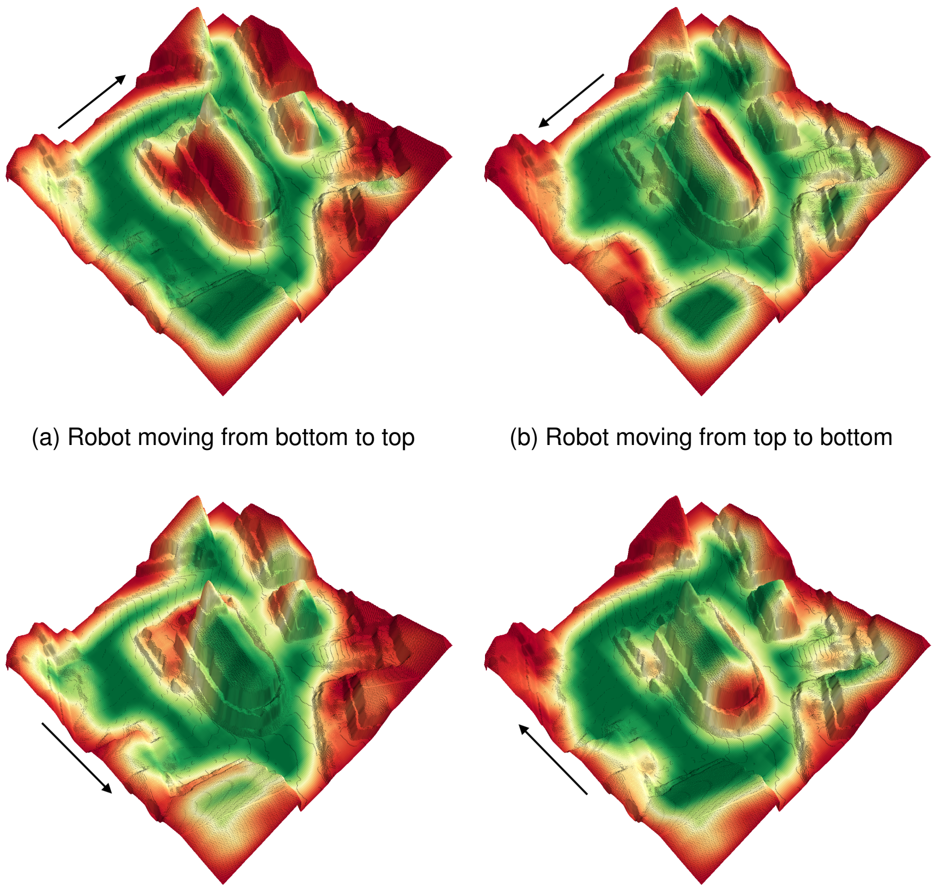
Figure 1.4. Detail of a region in the bars map where there are two walls forming a corridor. When the robot is following the trail the region is label as traversable.



(a) Robot moving from bottom to top
(b) Robot moving from top to bottom
(c) Robot moving from left to right
(d) Robot moving from right to left

Figure 1.5. Traversability probability on the map of a small village for different Krock's rotation. The surface covers 30×30m and has a maximum height of 10m.

when walking uphill. On the other hand, if krock walks side by side they can be traversed. Figure **??** shows this behavior on the church's roof.



(a) Robot moving from bottom to top

(b) Robot moving from top to bottom

(c) Robot moving from left to right

(d) Robot moving from right to left

Figure 1.6. Detail of the church in the small village map for different Krock's rotation. All the fours images are correctly labeled accordingly to the robot orientation. In the first two images only downhill part of the roof is traversable. While in the last two the robot is able to travel till the end.