Deep Transfer Learning of Traversability Assessment for Heterogeneous Robots

Josef Zelinka, Miloš Prágr, Rudolf Szadkowski, Jan Bayer, and Jan Faigl

Czech Technical University, Faculty of Electrical Engineering, Technická 2, 166 27, Prague, Czech Republic,
{zelinjo1|pragrmi1|szadkrud|bayerja1|faiglj}@fel.cvut.cz,
WWW home page: https://comrob.fel.cvut.cz

Abstract: For autonomous robots operating in an unknown environment, it is important to assess the traversability of the surrounding terrain to improve path planning and decision-making on where to navigate next in a cost-efficient way. Specifically, in mobile robot exploration, terrains and their traversability are unknown prior to the deployment. The robot needs to use its limited resources to learn its terrain traversability model on the go; however, reusing a provided model is still a desirable option. In a team of heterogeneous robots, the models assessing traversability cannot be reused directly since robots might possess different morphology or sensory equipment and thus experience the terrain differently. In this paper, we propose a transfer learning approach for convolutional neural networks assessing the traversability between heterogeneous robots, where the transferred network is retrained using data available for the target robot to accommodate itself to the robot's traversability. The proposed method is verified in real-world experiments, where the proposed approach provides faster learning convergence and better traversal cost predictions than the baseline.

1 Introduction

Our work is motivated by autonomous tasks such as mobile robot exploration, where robots encounter terrains that might impede their movement but have unknown properties due to the nature of the mission. In such deployments, the robots can improve the efficiency of their navigation by learning the terrain properties incrementally during the mission. Further, we can reason about distributing the exploration to multiple robots to finish the mission faster. For a heterogeneous team of robots, each robot can be assigned to a suitable part of the environment, such as a small crawler exploring tight spaces, while larger and faster robots can be assigned to open areas. However, robotic platforms with varying builds and sensory equipment have different terrain perceptions; hence, each platform needs to learn standalone terrain assessment models. The knowledge transfer approach can reduce the complexity of training and maintaining multiple standalone models.

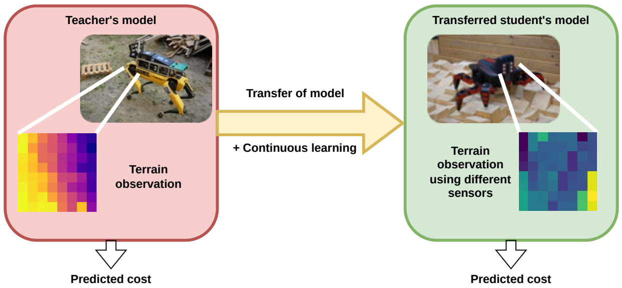


Figure 1: Assuming that the cost assessment model for the teacher is available, the teacher's model is transferred to the student and modified to accept the student's observation format indicated by the different colormaps. Then, the transferred model continues learning using the student's observations to be informed about the student's experiences.

Transfer learning is a part of the machine learning principles that aim to improve the performance in the target domain by the experience in the source domain [1][1]. In this paper, we propose to utilize transfer learning to share terrain traversability assessment models between heterogeneous robotic platforms, as illustrated in Figure 1. The individual models are neural networks that predict a continuous score describing the difficulty of the terrain traversal. The student's neural network is initialized by weights transferred from the teacher, and then the neural network is tuned using the data obtained in the student's target domain. If the dimensions of observations are unequal due to different sensors being used by the teacher and student, the input dimensionality is reduced (or increased) by additional neural network layers. The proposed approach is compared to the baseline, learned only in the target domain, using a robot with heterogeneous terrain experience simulated by various traversability assessment methods. Besides, the feasibility of the approach is validated in the experimental scenario with real heterogeneous robots.

The rest of the paper is organized as follows. The

[1]The source domain (teacher) denotes the entity providing knowledge to the individual in the target domain (student).

related work on traversability assessment and transfer learning is briefly reviewed in Section 2. In Section 3, the problem of traversability transfer between heterogenous robots is presented. Section 4 describes the proposed approach for the transfer of traversability assessment experience. The evaluation results of the proposed approach are presented in Section 5. Section 6 concludes the paper.

2 Related work

The traversability assessment is to support path planning and decisions, such as avoiding impassable terrain or optimizing the path for the specific needs of the robot [2]. Correct traversability estimation is essential for applications where the robot encounters different terrains, including dangerous environments. Such fields are represented by extra-terrestrial exploration [3–5], search and rescue missions [6], and agriculture or off-road driving [7]. In [2] and [8], the authors provide a thorough overview of traversability assessment methods suitable for mobile robots. Therefore, we focus our brief review on recent neural network approaches, which provide appearance-based traversability predictions and utilize image-processing and classification methods.

In [5], a fully convolutional neural network is utilized to locate the best possible place for the rover to land by classifying multiple terrain types and has been used for the Mars rover mission. An alternative architecture is proposed in [9], where traversability predictions on future paths are achieved using Generative Adversarial Networks (GANs) [10] that create virtual images from the already traversed path. However, a vast dataset is needed to train a neural network from scratch, fully highlighting the need for an approach capable of enhancing performance with a smaller dataset.

For such cases, transfer learning can be employed, which is an approach capable of improving knowledge in the target domain by the transfer from the source domain [1]. In [11], the authors utilize transfer learning in the form of the weight transfer from remotely similar tasks to reduce the dataset size in the target domain necessary to train the convolutional neural network and thus shorten the training time. After transferring weights between tasks, it is desirable to fine-tune them to suit the task's needs in the target domain. The classification feature extractor output layers are reinitialized in [12] to comply with the possibly different classes in the target domain. If the source and target tasks vary greatly, an entire redesign of the classifier's architecture may be employed as in [13], where it is advocated that the learning rate of the neural network is set lower when applying only slight corrections to weights during the fine-tunning. While fine-tuning, it can be beneficial to refrain from updating the weights in some layers (denoted as layers freezing) [14].

The costly data collection and various tasks and robots' bodies make robotics an interesting field for deploying transfer learning techniques. Although testing only in simulators, the authors of [15] utilize transfer learning to propagate experience in different scenarios of robotic soccer, where they propose a solution for transferring neural networks between tasks with different inputs and action spaces. In [16], humanoid robots observe human gestures and motions to replicate them later. Another method to transfer human experience to neural networks utilized by robots is presented in [17], where humans provide knowledge directly to the network. A robotic arm is trained to reach a destination of a colored block in [18]. The transfer is carried out between robotic arms with a different number of joints.

However, since the traversability over a single terrain may greatly vary between robotic platforms, such as wheeled and legged ground vehicles [19], the obtained knowledge cannot be shared directly between different robots. Therefore, we aim to utilize transfer learning to distribute a traversability assessment model consisting of a neural network.

3 Problem Statement

We examine various robots R^i perceiving diverse terrains T during operational usage. Let the robots be deployed in an environment modeled as the 2.5D grid, where each cell υ can be labeled by a number, and thus $\upsilon \in \mathbb{N}$, and the cell size d_υ corresponds to the footprint of the smallest robot in the team. The center of each robot's footprint is discretized as the cell $\upsilon^{\text{robot}} \in \mathbb{N}$. Robots move through the environment along paths ψ that are represented as sequences of neighboring cells $\upsilon_1, \ldots, \upsilon_n$ corresponding to the robot's discretized positions.

The robot R^i's path-planning decisions are made with respect to (w.r.t.) the particular robot's cost c^i by finding a path with the minimal expected cost

$$\psi^{*,i} = \operatorname{argmin}_{\psi \in \Psi(\upsilon,\upsilon')} \sum_{\upsilon_j \in \psi} c^i(\upsilon_j), \tag{1}$$

where $\Psi(\upsilon, \upsilon')$ is a set of all paths from υ to υ'. However, the cost function c^i is not known a priori; thus, the robot has to learn it to estimate the cost by $\hat{c}^i$. Hence, a traversability assessment model r^i is needed that assigns the predicted cost $\hat{c}^i$ for a terrain T observed using exteroceptive sensors as

$$r^i : A \to \hat{c}^i, \tag{2}$$

where A is the observed terrain appearance.

Since the mobile robot's traversability is considered too complex to be assessed using a handcrafted function, a model r^i is trained from the robot's experience to predict its future cost $\hat{c}^i$. The costs utilized for training are computed using proprioceptive sensors because of their ability to measure how the environment influences the robot's body. The training of each traversability model r^i aims to minimize the *Root Mean Square Error* (RMSE) between the model's cost assessments and the cost C^i measured using proprioceptive sensors as

$$\mathrm{RMSE}^i = \sqrt{\frac{1}{n}\sum_{j=1}^{n}(r^i(A(T_j)) - c_j^i)^2}. \quad (3)$$

We address the traversability assessment for heterogeneous robotic platforms $R^1 \neq R^2$, which possess different capabilities. Therefore, we assume that differences between the platforms can result in unequal cost measurements $c_j^1 \neq c_j^2$ over some terrain T_j. On the other hand, we assume that there are underlying similarities between how the robots interact with the terrain. The problem being addressed is to improve the performance of traversability assessment by transferring the cost assessment model r^1 from the robot R^1 before learning on the robot R^2, and thus learning its cost assessments relatively sooner while achieving similar or better predictions than the regressor r^2 trained using only the R^2's data.

4 Method

In the proposed approach for transferring mobile robot terrain traversal experience between the heterogeneous robots, the traversability experience is denoted as the traversal cost c. Then, for each robot, the costs are predicted using the robot's regressor r. Each regressor is a neural network trained using the robot's prior traversal costs associated with the description of the particular terrain where the cost was experienced. The teacher's terrain experience, represented as the teacher's learned network, is transferred to the student who has no prior terrain experience by using the teacher's weights to initialize the student's network. After the transfer, the student's network is further trained to adapt to the student's domain fully.

In the rest of this section, we describe in detail the traversal costs used by the individual robots, the regressor, its learning process, and the knowledge transfer.

4.1 Robots' Traversal Costs

The traversability assessment regressor is trained on trails that include observations of the traversed terrain paired with the cost perceived over the terrain,

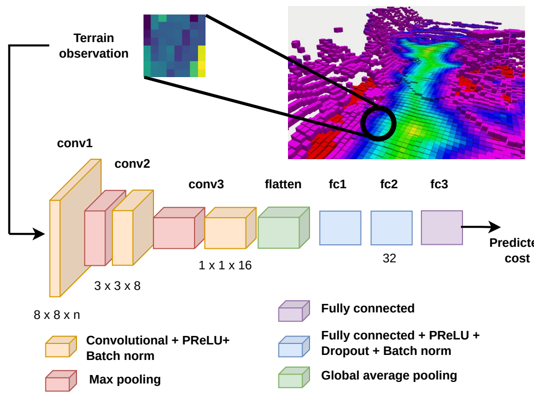


Figure 2: Architecture of the regressor, where n is the number of terrain observation's channels entering the neural network and the observation window width is $w = 8$.

which each robot can compute using its cost computation method. All the cost computation methods return strictly positive values because zero and negative values would incentivize infinite paths, preventing the robot from reaching its goal. We consider the following cost computation methods.

Velocity c_v - monitors relation of the achieved speed v and commanded velocity v using equation $c_v = \frac{v}{v_{\mathrm{cmd}}}$.

Slope c_s - computes the cost $c_s = 1 + \theta$ as an angular distance in the degrees from the straight pose θ. The offset by 1 is to accent the energy expenditure even on flat terrains.

Difference c_d - is defined similarly to c_s as $c_d = 1 + \gamma$, where γ expresses the maximal angular distance in degrees of the subsequent robot's positions.

Further, the costs are adjusted as

$$c_{\mathrm{adjusted}} = c_{\max} \cdot \tanh \frac{c}{c_{\max}}, \quad (4)$$

where c stands for c_v, c_s, and c_d. The adjustment is to lower the high-cost values for cases the achieved velocity is negligible when compared with commanded velocity because the robot gets stuck.

Tilt c_a - is computed as $c_a = \alpha = |\tan \frac{d_z}{d_{xy}}|$, where α is the absolute angle of the two opposing footholds (for the legged robots) from the flat surface. For example, the left front and right rear legs are considered opposing. The values d_z and d_{xy} measure a difference in elevation and flat-plane distance of the footholds, respectively.

4.2 Traversal Cost Regressor

The cost regressor r, based on the regressor proposed in [20], uses the terrain appearance and geometry to assess the robot's traversal cost. The regressor is a neural network that uses the image processing-like architecture shown in Figure 2 and operates as follows.

During the deployment, the robot uses its range measurements to build a height map $\mathbb{N}$ of the mission environment in the form of an elevation grid map with the squared cell size of 7.5 cm [21]. Depending on the carried sensory equipment, the grid map may also include the terrain color in addition to the elevation information, which can be further utilized in regression and extrapolation of the learned traversal experience.

The regressor is learned from the cost measurements localized in $\mathbb{N}$ paired with the terrain observations at the respective locations. Each terrain observation is in the form of a $w \times w \times n$ segment centered at the location, where w is the observation window size selected so that the entire robot's footprint is covered. The dimensionality n is either 1 when only range measurements are available, or 3 in the case range measurements are accompanied by `a` and `b` channels of the `lab` color space. The network is learned using `Adam` optimizer w.r.t. the mean absolute percentage error

$$\Lambda = 100 \left| (y_t - y_p) \frac{1}{y_t} \right|, \tag{5}$$

where y_t is the expected output of the neural network and y_p the prediction.

4.3 Knowledge transfer

During the knowledge transfer, the weights from the source domain are utilized in the target domain. Besides, when needed, the transferred model is adjusted as shown in Figure 3. The width w of the input window is prepared separately since it is selected to fit the robot receiving the model. However, the observation dimensionality can differ between teachers' and students' exteroceptive sensors. In such a case, an additional convolutional layer is used to reshape the input and accommodate the transferred regressor to the student's perceived data. The layer comprises 1×1 convolutional kernel with the input and output channels corresponding to the perceived number of channels and the number of transferred regressor's input channels, respectively.

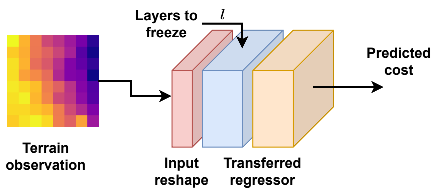


Figure 3: Setup used during the transfer of knowledge, where l denotes the frozen layers during the training.

The new model is retrained using the student's dataset collected by the student. It is because the teacher's and student's costs are heterogeneous, albeit it is assumed that the network captures underlying terrain properties that can be transferred. Additionally, during the retraining of the regressor, l layers can be frozen by fixing their weights since it is assumed that the initial layers extract general features that are primarily similar between various data.

5 Results

The proposed knowledge transfer method has been examined in several experimental scenarios. First, we simulate the heterogeneity of the robots using a small hexapod crawler with varying cost perception, which provides an easy way to verify the feasibility of the proposed approach. Then, we display the proposed knowledge transfer using two different real robots.

(a) (b)

Figure 4: (a) SCARAB II hexapod crawler and (b) Spot with sensor payload.

The proposed method can be used with any set of ground vehicles. However, we focus on multi-legged robots since their traversal capabilities permit deployment in a wide range of terrains. The utilized robots are the small hexapod crawling robot SCARAB II [22] and the four-legged Spot that are depicted in Figure 4. Besides their morphology, the robots also differ in size. Spot's footprint is larger than SCARAB II that occupies a disk with a 25 cm radii, while Spot's footprint fits into 1.1 m × 0.5 m rectangle. Furthermore, SCARAB II carried the Intel RealSense Tracking Camera T265 and the RGB-D Intel RealSense Depth Camera D435 providing depth and color appearance exteroceptive data. Spot perceived only range measurements using the Ouster OS0-128 LiDAR and does not perceive color.

5.1 Cost Assessment Methods Examination

The feasibility of the proposed method is firstly verified in a scenario where the difference in perception of heterogeneous robots is simulated using various cost assessment methods of SCARAB II. The robot collected the datasets in the Bull Rock Cave near Brno, Czech Republic, as described in [20]. The datasets were collected in various parts of the cave system, and each set is a result of the robot walking over one of the particular terrains, whose selection is shown in Figure 5. Each dataset collection included approximately 5 minutes of the navigation, enabling the robot to observe 6×6 m of the environment.

(a) (b) (c)

Figure 5: Test terrains in the Bull Rock Cave in the (a) *Chiffon*, (b) *Hall*, and (c) *Room* areas.

The transfer learning approach is examined using five scenarios for each pair of the cost assessment methods. The scenarios are prepared by randomly choosing five datasets from the collected dataset pool as testing data for the direct and transferred model. The testing datasets are removed from the datasets available to train the teacher's and the student's cost assessment models. From the remaining datasets, twelve are randomly drawn for the teacher and five for the student to create training datasets for their cost assessment models. All regressors are trained for 300 epochs with a training-validation split of 9-to-1, and the width of the observation window is $w = 8$.

Table 1: Mean (std) of the RMSE for 5 randomly generated scenarios for each pair of cost assessment methods. For the transfer regressors, the number denotes the frozen layers l during the retraining of the transferred model.

Scenario	Direct	Transfer 0	Transfer 4	Transfer 8
$c_d \to c_s$	2.10 (0.27)	2.08 (0.27)	2.27 (0.58)	2.00 (0.19)
$c_d \to c_v$	3.25 (1.37)	1.68 (0.57)	1.43 (0.32)	1.95 (1.32)
$c_s \to c_d$	2.49 (1.33)	1.43 (0.16)	1.34 (0.22)	1.40 (0.16)
$c_s \to c_v$	2.83 (0.95)	2.19 (0.11)	1.98 (0.40)	2.01 (0.45)
$c_v \to c_d$	3.41 (1.41)	1.60 (0.15)	1.52 (0.23)	1.34 (0.08)
$c_v \to c_s$	2.58 (0.57)	2.36 (0.65)	2.27 (0.28)	1.98 (0.09)
Overall	**2.78 (0.98)**	1.89 (0.32)	1.80 (0.34)	**1.78 (0.38)**

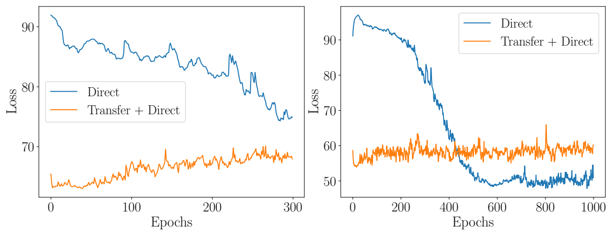
The results presented in Table 1 show that the model transfer lowered the RMSE of predictions on the testing datasets. The best results are achieved with $l = 8$ frozen layers.

Table 2: RMSE's mean (std) values between 300 and 600 epochs of $c_v \to c_s$ transfer.

Epochs	Direct	Transfer 0	Transfer 4	Transfer 8
300	2.58 (0.57)	2.36 (0.65)	2.27 (0.28)	1.98 (0.09)
600	2.13 (0.28)	2.31 (0.38)	2.25 (0.30)	2.25 (0.21)

In Table 2, the transfer from c_v to c_s is chosen to examine the influence of training for increased number of the training epochs. The same randomly generated transfer scenarios are utilized as in Table 1; however, the models are trained for 600 epochs instead of 300. The results show that the direct student's model has improved with the increased number of epochs, while the RMSE of the transferred model slightly increased during the prolonged training, likely because of overfitting the training data. The transfer model learned with 300 epochs has achieved sufficient performance, and the results suggest that the transfer helps reduce the necessary number of training epochs.

Transfer between Slope c_s and Velocity c_v Cost Assessment Method - We further examine the transfer between the student's slope c_s and teacher's velocity c_v cost assessment method in detail as those methods compute cost using dissimilar approaches. The dataset is split so that the student's dataset is overall a third size of the teacher's, hence suitable to showcase the knowledge transfer as there is much information to be received by the student. Teachers'



(a) (b)

Figure 6: Progress of the cost assessment model's neural network training for (a) 300 and (b) 1000 epochs.

and students' direct baseline cost assessment models were trained for 300 epochs. After the teacher's model was transferred to the student, it was tweaked using 300 epochs and $l = 0$ frozen layers to create the transferred model. Figure 6a summarizes the first 300 epochs in the student's domain, where an initial boost can be observed for the transferred model. The prolonged training for 1000 epochs, shown in Figure 6b, results in the improved validation loss achieved by the student's model. After training for 300 epochs, the RMSE of the regressors' predictions against the collected ground truth is 3.69 and 1.84 for the direct and transferred models, respectively. Hence, we can conclude that the transferred model improved performance faster.

Moreover, we examined the traversal of a single cave trail. Figure 7 shows the predicted costs by the student's direct and transferred models compared to the collected ground truth. We can observe that the transferred model follows the ground truth better than the direct model.

Figure 8 shows the cost prediction for the entire environment observed from the trail. Note that the ground truth for the whole view is unavailable as the robot traversed just a single path (the trail). Thus, we manually evaluate the cost assessments' feasibil-

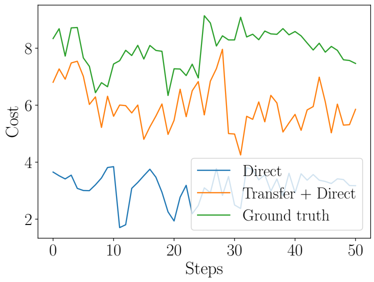
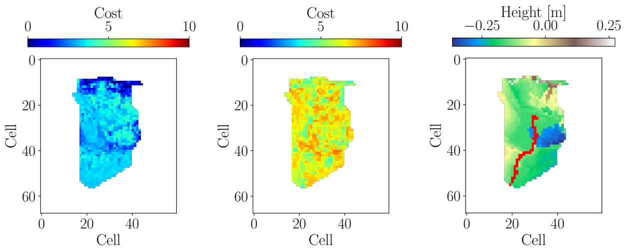


Figure 7: Predicted costs by the student's and transferred neural network compared to the ground truth after training for 300 epochs.



(a) Direct model (b) Transfer model (c) Height Map

Figure 8: (a) Direct and (b) transferred model's cost assessments of the perceived environment after training for 300 epochs; (c) and the environment's height map, where the path of the robot is in red. The shown maps have squared cells with the size 7.5 cm.

ity for path planning. Compared to the direct model, the transferred model returns higher costs in locations where the elevation of the height map changes, suggesting that the transferred model produces improved assessments.

5.2 Transfer between SCARAB II and Spot

The dataset comprising heterogeneous robots is created by adding the Spot's datasets to the SCARAB II's data. The datasets were collected in indoor and outdoor locations of the Czech Technical University in Prague campus at Charles Square, Prague, Czech Republic. Indoors, see Figure 4b, Spot moved over surfaces partially covered with artificial grass and spikes in the form of soundproofing material. Outdoors, the robot traversed various surfaces such as hard-packed soil, cobbles, and sloped grass. Spot can move faster than SCARAB II; thus, the Spot's datasets are longer, as Spot traverses more terrain in similar 5-minute long deployment, where Spot is capable of traveling through 15 × 15 m environment. Therefore, using fewer datasets to train the cost assessment model is sufficient. In the following paragraphs, we examine the performance when transferring both from Spot to SCARAB II, and in the opposite direction.

Spot Knowledge Transfer to SCARAB II - The transfer from Spot to SCARAB II is achieved using observation windows with the width of $w = 8$ cells, which is suitable for the smaller hexapod crawler. Each transfer scenario, comprising transfer from Spot to one of the hexapod's cost models, is evaluated in 5 setups. For each setup, 5 datasets are randomly chosen to train the Spot's teacher model, while SCARAB II receives 6 randomly chosen datasets. The trained models are examined on 5 randomly chosen datasets, which differ from the training sets. The regressors are trained for 300 epochs with a 9-to-1 training-validation split.

SCARAB II models the environment as a colored height map, while Spot uses only a height map. Thus, we consider the student's model input with both $n = \{1, 3\}$ channels in each transfer scenario. When using the three-channel version, which perceives both the elevation and the `a` and `b` channels of the `lab` color space, a convolutional layer reshaping the input, is added to accommodate teacher's (Spot's) model that has only one input channel.

Table 3: Mean (std) of the RMSE for knowledge transfer from Spot to SCARAB II. Transfer x denotes x frozen layers during retraining the regressor, $x \in \{0, 4\}$. **Depth** denotes $n = 1$ input channel for the resulting model perceiving just height map, **Depth** + **ab** denotes $n = 3$ where SCARAB II utilizes colored observations.

Observ.	Scenario	Direct	Transfer 0	Transfer 4
Depth	Spot → c_d	2.13 (0.59)	1.45 (0.26)	1.45 (0.27)
	Spot → c_s	2.50 (0.72)	2.29 (0.35)	2.28 (0.53)
	Spot → c_v	3.80 (1.52)	2.51 (1.04)	2.31 (0.85)
	Overall	**2.81 (0.94)**	2.08 (0.55)	**2.01 (0.55)**
Depth+ ab	Spot → c_d	2.44 (1.19)	1.60 (0.30)	1.53 (0.60)
	Spot → c_s	2.80 (1.49)	2.15 (0.21)	2.05 (0.13)
	Spot → c_v	2.37 (0.86)	2.34 (1.06)	1.87 (0.87)
	Overall	**2.54 (1.18)**	2.03 (0.52)	**1.81 (0.54)**

Table 3 shows the performance of the trained models. All transferred models perform overall better than the direct model. However, the performance of the transferred model has not improved when modifying the teacher's model to accept the colored height map collected by SCARAB II, although the direct model has improved when using $n = 3$ input channels. In the authors' opinion, the added convolutional layer could not sufficiently modify the input observation to achieve good performance in combination with the underlying transferred model.

SCARAB II Knowledge Transfer to Spot - For the transfer from SCARAB II to Spot, the scenarios are adjusted by using $w = 16$ to match Spot's body size, and the regressors are trained for 100 epochs in Spot's target domain. Besides, the reshaping convolutional layer is added during the transfer to Spot to utilize the SCARAB II's model with the three input channels.

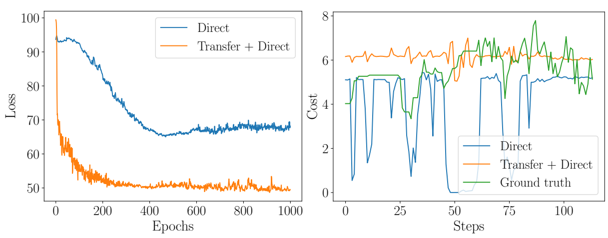
Table 4: Mean (std) of the RMSE for knowledge transfer from SCARAB II to Spot. Transfer x denotes x frozen layers during retraining the regressor, $x \in \{0, 4\}$. **Depth** denotes $n = 1$ input channel for the resulting model perceiving just height map, **Depth + ab** denotes $n = 3$ where SCARAB II utilizes colored observations. Note that only one direct model is created for both the **Depth** and **Depth + ab** observation setup, since for the student, both scenarios possess the same number of input channels n.

Observ.	**Scenario**	**Direct**	**Transfer 0**	**Transfer 4**
Depth	$c_d \rightarrow$ Spot	0.37 (0.14)	0.28 (0.14)	0.52 (0.32)
	$c_s \rightarrow$ Spot	0.32 (0.14)	1.72 (2.12)	0.29 (0.23)
	$c_v \rightarrow$ Spot	0.92 (0.82)	0.31 (0.19)	0.24 (0.19)
	Overall	**0.54 (0.37)**	0.77 (0.82)	**0.35 (0.25)**
Depth+ ab	$c_d \rightarrow$ Spot	/	0.11 (0.01)	0.11 (0.01)
	$c_s \rightarrow$ Spot	/	0.12 (0.00)	0.11 (0.01)
	$c_v \rightarrow$ Spot	/	0.11 (0.01)	0.11 (0.00)
	Overall	/	**0.11 (0.01)**	**0.11 (0.01)**

The results in Table 4 indicate improvements when utilizing transfer learning. The transfer of the SCARAB II's model perceiving color has achieved the best performance on the testing dataset. The authors suppose that the added convolutional layer and increased number of channels help the model better grasp the underlying terra-mechanical properties.

5.3 Individual Transfer between Spot and SCARAB II

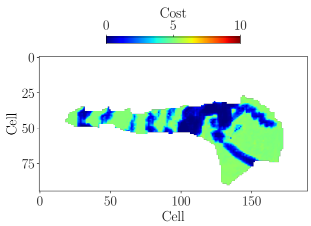
We further present a detailed overview of the knowledge transfer between Spot and SCARAB II. SCARAB II utilizes the difference cost computation method c_d, and the width of the observation window is set to $w = 8$. After 300 training epochs, the student's direct transferred and fine-tuned models achieved RMSE of 1.79 and 1.09, respectively.



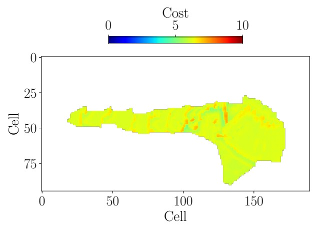
(a) (b)

Figure 9: (a) Progress of the cost assessment model's neural network training for 1000 epochs; (b) and student's and teacher's predicted costs, and the ground truth after training for 300 epochs.

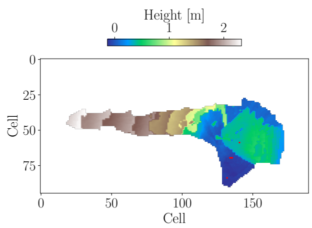
The progress of the regressors' training after a prolonged training for 1000 epochs is depicted in Figure 9a. The boost caused by the transfer can be observed in the initial epochs. After training for more than 400 epochs, the improvement in loss stops; particular losses do not change significantly until the end of the training. The transferred model produces lower validation losses, and the RMSE after training for 1000 epochs is 1.64 and 1.38 for the direct and transferred model, respectively. The observation suggests that both models are overfitted after such prolonged training. Figure 9b shows the measured and predicted costs on a particular trail where we can observe that the transferred model follows the ground truth closely, avoiding spikes in the assessments observed in the direct model. However, even the transferred model cannot closely follow the oscillations of the ground truth.



(a) Direct model



(b) Transfer model



(c) Height map

Figure 10: (a) Direct and (b) transferred model's cost assessments of the perceived environment after training for 300 epochs; (c) and the environment's height map with the squared cell of the size 7.5 cm.

Figure 10 illustrates the cost assessments and the height map with the marked robot trail in the *Room* part of the cave. Since the robot traversed only a tiny part of the observed environment, we present a manual evaluation of the cost assessments as the cost measurements are not presented in the observed area. The transferred model assigns a higher cost to the terrain edge, while the student's model underestimates the difficulty. Additionally, the transferred model suggests a more challenging cost in all areas of the environment, which resembles the actual perceived cost more closely. Thus, we conclude that the transfer can correct the student's direct model predictions.

6 Conclusion

In this paper, we present an approach for sharing knowledge about traversability between heterogeneous robots. Traversal cost predictors are created using neural networks processing observations from exteroceptive sensors. The knowledge transfer is implemented as the transfer of neural network weights, and the transferred networks are fine-tuned to adapt to the receiving robot's terrain perception. The proposed method is verified using a small hexapod crawler and a large quadruped walker, with the proposed method lowering the traversability prediction error. Next, we aim to deploy the proposed method in path planning tasks, with the final goal of simultaneous online learning on multi-robots.

Acknowledgement

This work was supported by the Czech Science Foundation (GAČR) under research project No. 21-33041J.

References

[1] S. J. Pan and Q. Yang, "A survey on transfer learning," *IEEE Transactions on Knowledge and Data Engineering*, vol. 22, no. 10, pp. 1345–1359, 2010.

[2] P. Papadakis, "Terrain traversability analysis methods for unmanned ground vehicles: A survey," *Engineering Applications of Artificial Intelligence*, vol. 26, no. 4, pp. 1373–1385, 2013.

[3] S. Singh, R. Simmons, T. Smith, A. Stentz, V. Verma, A. Yahja, and K. Schwehr, "Recent progress in local and global traversability for planetary rovers," in *IEEE International Conference on Robotics and Automation (ICRA)*, 2000, pp. 1194–1200.

[4] D. B. Gennery, "Traversability analysis and path planning for a planetary rover," *Autonomous Robots*, vol. 6, no. 2, pp. 131–146, 1999.

[5] B. Rothrock, R. Kennedy, C. Cunningham, J. Papon, M. Heverly, and M. Ono, "SPOC: Deep Learning-based Terrain Classification for Mars Rover Missions," in *AIAA SPACE*, 2016, pp. 1–12.

[6] B. Cafaro, M. Gianni, F. Pirri, M. Ruiz, and A. Sinha, "Terrain traversability in rescue environments," in *IEEE Safety Security and Rescue Robotics (SSRR)*, 2013, pp. 1–8.

[7] A. Huertas, L. Matthies, and A. Rankin, "Stereo-based tree traversability analysis for autonomous off-road navigation," in *IEEE Workshops on Applications of Computer Vision (WACV/MOTION)*, 2005, pp. 210–217.

[8] M. A. Bekhti, "Traversability Cost Prediction of Outdoor Terrains for Mobile Robot Using Image Features," Ph.D. dissertation, Shizuoka University, 2020.

[9] N. Hirose, A. Sadeghian, F. Xia, R. Martín-Martín, and S. Savarese, "VUNet: Dynamic scene view synthesis for traversability estimation using an rgb camera," *Robotics and Automation Letters*, vol. 4, no. 2, pp. 2062–2069, 2019.

[10] A. Creswell, T. White, V. Dumoulin, K. Arulkumaran, B. Sengupta, and A. A. Bharath, "Generative adversarial networks: An overview," *IEEE Signal Processing Magazine*, vol. 35, no. 1, pp. 53–65, 2018.

[11] K. Gopalakrishnan, S. K. Khaitan, A. Choudhary, and A. Agrawal, "Deep convolutional neural networks with transfer learning for computer vision-based data-driven pavement distress detection," *Construction and Building Materials*, vol. 157, pp. 322–330, 2017.

[12] H.-C. Shin, H. R. Roth, M. Gao, L. Lu, Z. Xu, I. Nogues, J. Yao, D. Mollura, and R. M. Summers, "Deep convolutional neural networks for computer-aided detection: CNN architectures, dataset characteristics and transfer learning," *IEEE Transactions on Medical Imaging*, vol. 35, no. 5, pp. 1285–1298, 2016.

[13] R. Girshick, J. Donahue, T. Darrell, and J. Malik, "Region-based convolutional networks for accurate object detection and segmentation," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 38, no. 1, pp. 142–158, 2016.

[14] R. Ribani and M. Marengoni, "A survey of transfer learning for convolutional neural networks," in *Conference on Graphics, Patterns and Images Tutorials (SIBGRAPI-T)*, 2019, pp. 47–57.

[15] M. E. Taylor, S. Whiteson, and P. Stone, "Transfer via inter-task mappings in policy search reinforcement learning," in *International Conference on Autonomous Agents and Multiagent Systems (AAMAS)*, 2007, pp. 156–163.

[16] N. Makondo, M. Hiratsuka, B. Rosman, , and O. Hasegawa, "A non-linear manifold alignment approach to robot learning from demonstrations," *Journal of Robotics and Mechatronics*, vol. 30, no. 2, pp. 265–281, 2018.

[17] K. Ogawa and P. Hartono, "Infusing common-sensical prior knowledge into topological representations of learning robots," *Artificial Life and Robotics*, pp. 1–10, 2022.

[18] C. Devin, A. Gupta, T. Darrell, P. Abbeel, and S. Levine, "Learning modular neural network policies for multi-task and multi-robot transfer," in *IEEE International Conference on Robotics and Automation (ICRA)*, 2017, pp. 2169–2176.

[19] P. Arena, C. F. Blanco, A. Li Noce, S. Taffara, and L. Patanè, "Learning traversability map of different robotic platforms for unstructured terrains path planning," in *International Joint Conference on Neural Networks (IJCNN)*, 2020, pp. 1–8.

[20] J. Zelinka, M. Prágr, R. Szadkowski, J. Bayer, and J. Faigl, "Traversability transfer learning between robots with different cost assessment policies," in *2021 Modelling and Simulation for Autonomous Systems (MESAS)*, 2022, pp. 333–344.

[21] J. Bayer and J. Faigl, "Decentralized topological mapping for multi-robot autonomous exploration under low-bandwidth communication," in *European Conference on Mobile Robots (ECMR)*, 2021, pp. 1–7.

[22] M. Forouhar, P. Čížek, and J. Faigl, "SCARAB II: A small versatile six-legged walking robot," in *5th Full-Day Workshop on Legged Robots at IEEE International Conference on Robotics and Automation (ICRA)*, 2021, pp. 1–2.