Seminar WT 2020/21: Terrain Segmentation and Roughness Estimation using RGB Data: Path Planning Application on the CENTAURO Robot

Cüneyt EREM

Supervisor: Prof. Dr. Maren Bennewitz

January 19, 2021

Abstract

In the real world environment, robots are used in different areas like security, rescue operation, etc. For this reason, they need to be sensitive towards the environmental changes not to lose robot control and successfully draw a path if necessary and give fast responses they needed. This report delivers a summary of the study and critical point of view that how they give their approach. The terrain segmentation and roughness estimation are two major factors to calculate the current situation by considering the images and creating a valid path for navigation. They deliver the solutions to pixel-wise labels of terrain and uses an RGB-based convolutional neural network. It also shows how to adjust the robot's center of the mass polygon with its legs and position as creatures do naturally.

1 INTRODUCTION

Over the past years, the number of robots increases highly because of their benefits to the people. One of the areas that the autonomous wheeled robots are used in rescue operations that can travel in the surroundings. Especially many different techniques are used as safe zone inside of the internal places [1]. However, on the outside, navigation, path planning, and avoiding obstacles are hard problems for the robots. To find out better solutions and handle the outdoor environment problems, robots can use the natural creature features like human perceptual visual and control actions that recognition of the materials like roughness and slippage. So, terrain segmentation and estimation will be stimulated as planetary and material finding examples.

The Convolutional Neural Network has a big impact on the state-of-the-art for terrain identification in terms of performance [2] but categorizing the visual objects is not very easy to handle. The robot predicts the terrain roughness and its features at the same time by using physical rules like reflection and illumination as the human brain evaluates these principles. Then the robot can identify the surroundings that make terrain segmentation and roughness estimation that yields to create path planning with the algorithm and can travel securely. While it moves, it can change its speed and come up with different path planning to keep pace with the environment.

This study mostly focuses on how to handle terrain segmentation and roughness estimation, then create path planning for the robot that adjusts its legs and body as well as its wheels. The training has a single network as Fig.1 illustrated, it applies CNN for the terrain segmentation and roughness estimation with low-level RGB properties which mainly based on pixel-wise label estimation instead of sharing some layers and parameters together and terrain estimation that continues values are calculated.

Applying the traveling and updating the path planning at the same time provides the vision to solve kind of problems, the training uses RGB data and count layer depth with current state-of-the-art terrain segmentation [3]. The purpose of using CNN instead of traditional encoder-decoder is about performance issues that can decrease the inference time and RGB cameras are very useful compared to Lidar sensor. Then it can give instant or very fast responses against the changes or obstacles for the terrain. It also changes any network module to work with different properties that are prioritized and not necessary for depth.

These properties are adjusted for the CENTAURO wheeled and legged robot for the terrain surface. The dataset can be found here: //sites.google.com/site/tsrenet This study examines the summary with the sections of relevant studies, procedure, test, and implementation. Then discussion gives a critical evaluation from different aspects, then a conclusion is given.

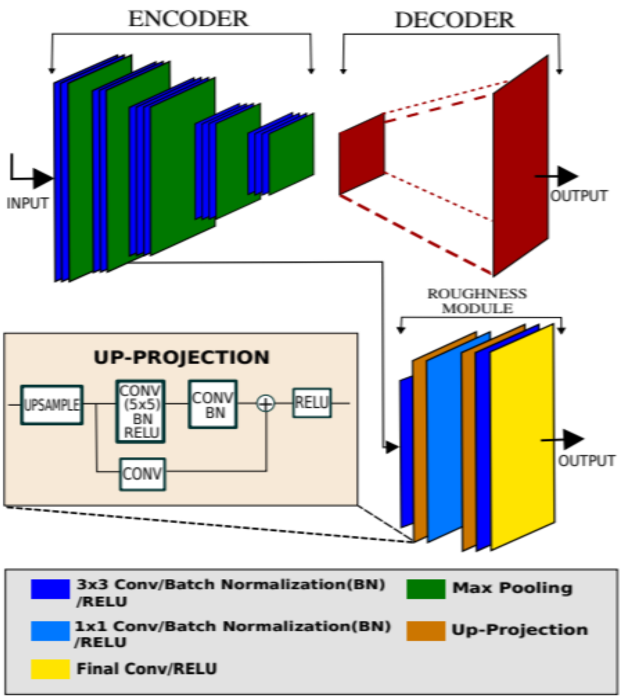

Figure 1

2 RELEVANT STUDIES

Terrain segmentation classification has the real time segmentation process for the image exhibition by considering the color, texture and features. The state-of-the-art uses deep learning algorithms like CNN instead of multi-class detectors. Bell et al. [4] uses the Conditional Random Field and CNN for the terrain segmentation in which it uses the sliding window approach that tags the particles of the materials. On the other hand, Wang et al. [5] uses Fully Convolutional Network (FCN) architecture for the 4D light field data whereas Schwartz et al. [6] depends on the global context. The FCN models are mostly based on the cityscapes dataset but this model of the study is mostly based on F. Schilling's study model and also specific terrain dataset are used for the training of the fine-tuned images.

The roughness estimation is performed in several studies such as D. B. Gennery [7] and S. B. Goldberg [8] for the navigation. Howard and Seraji [9] evaluates the roughness by undercover the rock particles using image-based approach. However, J. Kim et al [10] used RGBD camera and CNN architecture for the surface roughness estimation by reflection and S. Ram [11] uses data auto labeled with the metric that with SegNet. In this study, it estimates as the continuous terrain roughness. Some properties like friction that particles are detected by SegNet CNN or slippage that are detected by geometric and image features are used as a similar way. The appearance learning uses non-linear map on tilt and sliding but this study uses different way because it uses combinations of the approaches. The joint learning can share the layers in the single network like using semantic division and depth estimation or true features estimation or 3D rebuilding. For this study, it segments and estimates the terrain as end-to-end which is not proposed before so that diverse datasets are used for each module.

3 PROCEDURE

Terrain Segmentation: The architecture for the terrain segmentation is deep convolution encoder-decoder where encoder features are extracted into pooling, then decoder up-samples. The robot actions can be considered as pixel-wise and predictions need better performances, less memory usage and lower inference time so that SegNet and ERFNet architectures are used instead of FCN-based architectures. In addition after the training step, if dropout layers and batch normalization are combined, it can be useful.

For the sake of computational time efficiency and ease of use, the pre-trained models instead of from the rough with open terrain segmentation datasets in which it can include some biases for unseen terrains are used. The bias can be handled by applying random flips and contrast with 0.25 probability.

For the fine-tuning, RGB images are reshaped with 256×512 dimensions and loss function weights are computed by Median frequency balancing. Whereas, class imbalance minimizes the cross-entropy loss (lr= 10-3, momentum= 0.9, weight decay = 0.0005). The Softmax output confidence is 0.4 for labeling the data. The further steps like CRF are not used not to increase memory usage.

Roughness Estimation: The CNN depth boosts the quality of being exact for the feature spaces so that training network focuses on bottom layers for the estimation such that the probability P(F/I) with input image and features with roughness estimation R can count up the probability P(R/F, I). Then the formula is defined as;

$$Pr(R, F|I) = Pr(R|F, I)Pr(F|I) \tag{1}$$

SegNet bottom layers can provide enough feature map division and it can encode them in the roughness module by using up-projection to boost spatial decomposition that delivers the non-excessive memory used layers and so that in-osculate faster. The roughness module contains 3×3 kernel convolution layer, up-projection blocks with 1x1 convolution that applies the batch normalization and ReLU activation of the convolution layers. The training step with 25 batch size, 10-7 learning rate and loss of Euclidean and BerHu [12], it adds the roughness module with second pooling layer so that segmentation weights are not in use but it dos not enhance the regression performance. The threshold c changes the norms between L1 and L2 that x is error for prediction, output gradient descent is calculated for c and k in every step. The loss of the BerHu is better that Euclidean in performance issues. the formula for the c is defined as;

$$f(x) = \begin{cases} |x|, & \text{if } |x| \leq c \\ \frac{x^2+c^2}{2c}, & \text{otherwise} \end{cases} \tag{2}$$

$$c = \frac{1}{5} max_k(|x_k|) \tag{3}$$

4 TEST

Datasets: The terrain dataset is created and ejected terrain particles like sand, stone, wood, metal, road and grass from the pixel-wise images of ADE20K and equilibrate with closed images of OpenSurfaces. The 65% of the data are used for training, 35% are used for evaluating in 1380 images, and RGB images from the camera are used for validation. The roughness dataset is estimated from RGB images to make the computation faster which is different approach than the previously proposed approaches. The roughness value for the x, y, z points are computed by plane values divided by distance like the formula:

$$r_i = \frac{|-d - ax_i - by_i - cz_i|}{\sqrt{a^2 + b^2 + c^2}} \tag{4}$$

The GeoMat dataset is used for the training. The terrain image patches with 3D depth have 400×400 size and 800×800 pixels and converted into points.

Test result: The validation part consists of roughness estimation output that is implemented in Caffe. The terrain segmentation uses three networks that are ENet, SegNet, and ERFNet. For the 256 ×512 pixels of the images, the inference times are 10.25ms, 8.29ms and 3.86ms per frame. The average accuracies for the evaluation are 54%, 64% and 65% whereas mean Intersection over Union are 27%, 43% and 49% respectively. As a result, the state-of-the-art method ERFNet has the best performance for terrain segmentation. For the 50 epoch, an estimation can be improved if the first layer of the CNN is stopped. As it can be seen in Fig. 2, Grand truth and Segnet can identify the particles like sand, stone, etc. but still, they can be identified wrongly. In Fig.3, the RGM image by using brightness and contrast effect, the estimation can be done better.

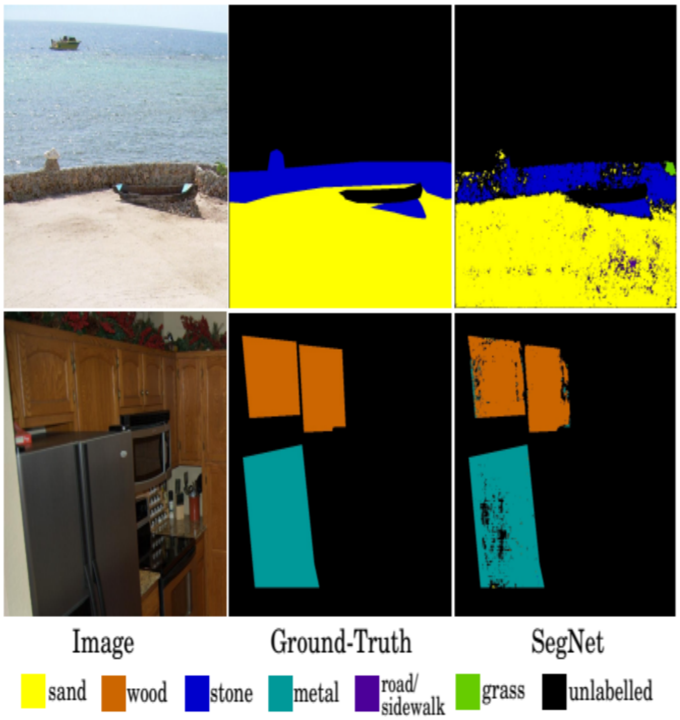

Figure 2

Terrain Roughness: If the error assessments are pixel-wise and uses loss functions Euclidean and BerHu, the Root Mean Square Error and the Root Mean Square Log Error are quite proximate, yet BerHu is better in the validation dataset as seen in the table.

TABLE I

ERROR ASSESSMENT FOR EUCLIDEAN AND BERHU LOSS FUNCTIONS

Our model	RMSE	RMSLE
Euclidean	0.4350	0.1123
berHu	0.4335	0.1119

Figure 3

In Fig. 4, the model predicts roughness as a 0.05 - 1.4cm range instead of 0-3.7cm of grand truth model because of biases in which the real-world also consists of. The smooth plane is predicted as very low roughness whereas borders are high. The training process computes moderate for the lower values. Also, the appearance-based features has lower biases as Fig. 5 shows.

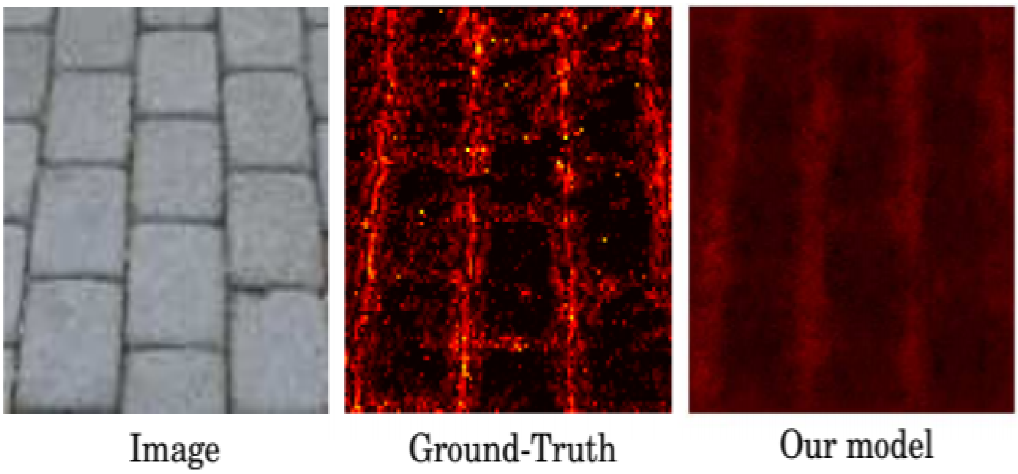

Figure 4

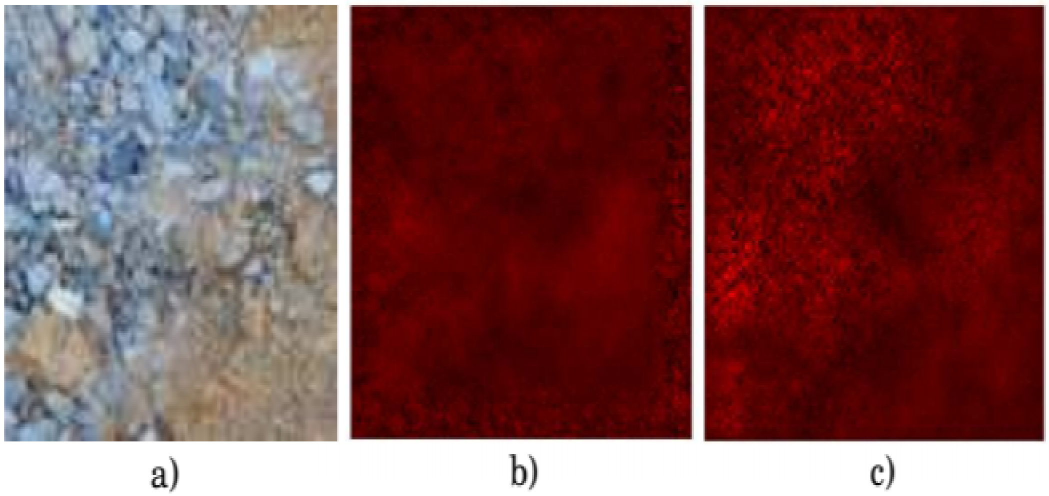

Figure 5

5 Model application on the robot

The wheeled-legged robot CENTAURO has 1.5 m height and 93kg weight having an RGB head camera, four 7-DoF wheeled legs, real-time control over ROS, and CartesI/O interface. It can behave 6D pose control actions with these features. In the first look, it can be seen as a big spider that can attack but it will not. This study measured risks for the terrain segmentation that can manage to detect rough surfaces that lower the body height by changing the center of mass, and roughness estimation to draw a plan to estimate the lowest risk actions.

Figure 6

As it can be seen in Fig. 7, terrain segmentation and roughness estimation results can be different and have some minor mistakes that some particles are not detected correctly. The head camera uses 30 FPS, 192×384 images with minor blurring effect and applies ERFNet architecture to yield 15.37ms inference time. Still, modeling outcomes yields good results for the lower distances and different image sizes.

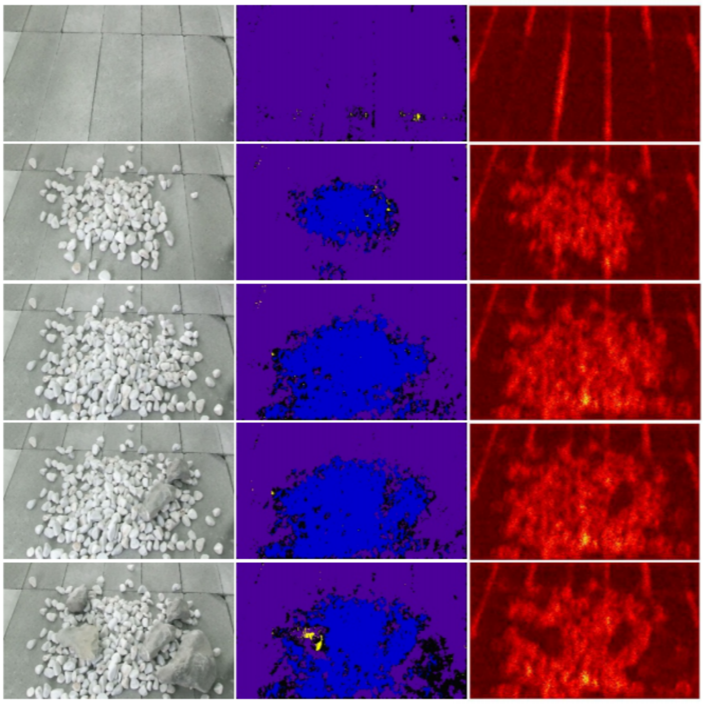

Figure 7

When the robot moves across the uneven surface, it can lose its balance and may collapse or cannot control itself properly. In natural life, creatures can balance themselves by decreasing the center of mass height so that they can have higher control of their bodies. Therefore same principle is applied here that the robot decreases its height by expanding its legs and increase its area on its feet. In Fig. 8, as it can be seen that stretched legs position is not very stable that has 0.36m2 support polygon and 0.95m pelvis height while moving, however, bending legs position that has 0.63m2 support polygon and 0.75m pelvis height can provide the robot with much stable travel not to lose the balance. It is the reason that torques of the joint motors are increased, yet it consumes more energy.

Before the robot moves through the roughness, it needs some initial preparation time to approximate the magnitude and apply the change if it satisfies the average roughness value for the desired action. It can act as stretched, bending, or both balanced center of mass positions for its control. As it can be seen in Fig. 9, the stretched legs robot position has very low torque value until it reaches the terrain roughness particles that cause high undulations of the torque. The bending legs position has more torque value and energy consumption until the roughness terrain, then it has smaller ripples than before. The stretched-bend position is the proper choice for the minimum torque and maximum balance robot control.

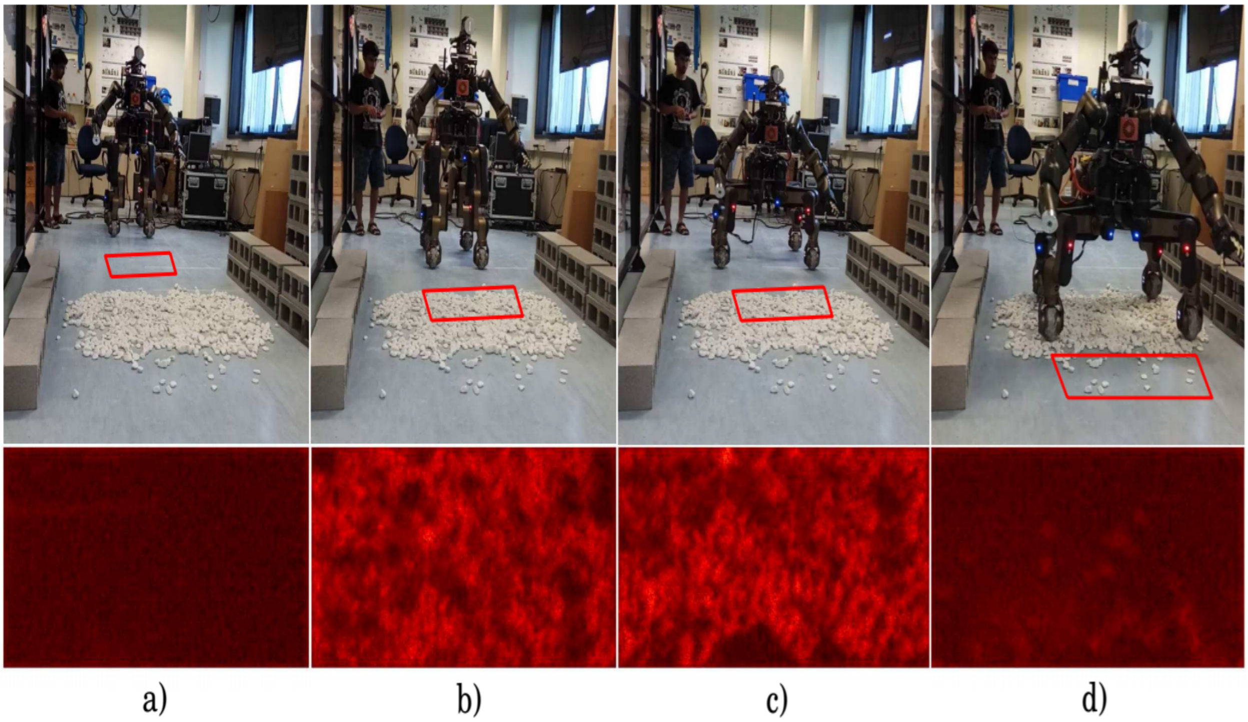

Figure 8

Before the robot has entered the roughness area, it also has the path planning with the algorithm to estimate the route safely. The robot has a Lidar sensor with 0.3Hz, using a Heatmap, 2rad/s rotating sensor along its head camera to measure its polygon area and height. It can change its behavior according to the obstacle shapes and places such that it can make itself more strait for the narrow road or pass by the obstacle if the road has available space. There are three main steps to evaluate and make an estimation of the algorithm.

The first step is the segmentation of the terrain by using the Lidar sensor that it can make classification as shorter height than 0.4m points, greater height than 0.4m points, and obstacle-free points. It can manage go through the shorter height points but cannot handle the higher height points. The second step uses A* algorithm [13] to find the path that if the robot encounters an obstacle, it can keep its distance from the obstacle, it can expand its legs so that can pass over the obstacle or narrow the legs not to hit the obstacles and pass over the objects. It calculate the A* algorithm with computing the cost function that (xp, yp) is the parent node, (xc, yc) is the current target node, and calculate the g(xc, yc) and heuristic h(xc, yc) with their theta. Wc is the weight change, —delta w— is the polygon change, Wg is the cost of goal change.

$$g(x_c, y_c) = g(x_p, y_p) + W_t * \frac{|\theta_{\mathbf{ent}} - \theta_{\mathbf{ext}}|}{2\pi} + W_c * (|\delta w|) \qquad (5)$$

$$\theta_{\mathbf{ent}} = \arccos\left(\frac{x_p - x_c}{\sqrt{(x_p - x_c)^2 + (y_p - y_c)^2}}\right) \qquad (6)$$

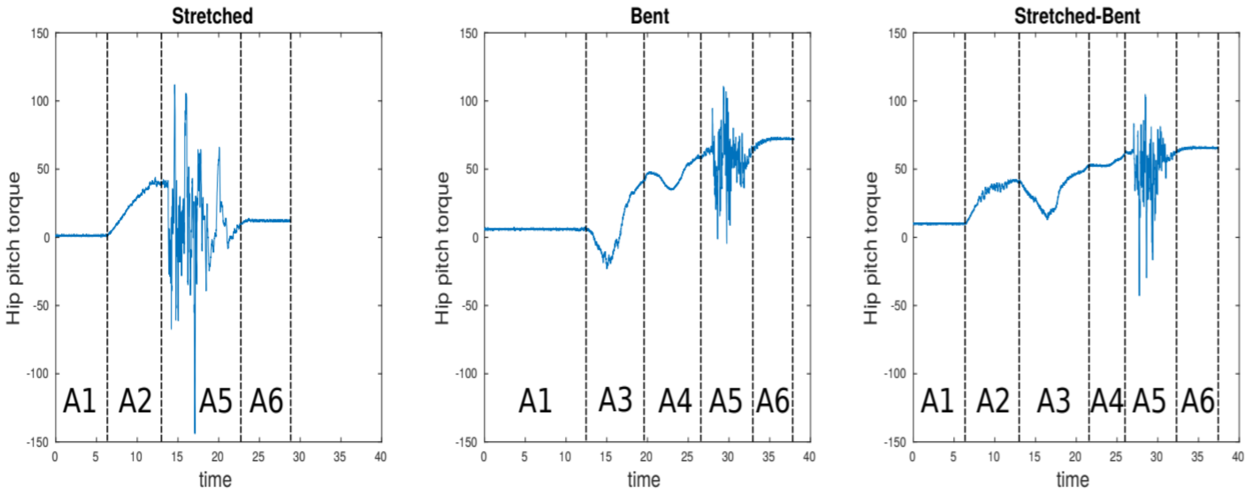

Figure 9

$$h(x_c, y_c) = \sqrt{(x_g - x_c)^2 + (y_g - y_c)^2} + W_g * \frac{|\theta_{\mathbf{goal}} - \theta_{\mathbf{ent}}|}{2\pi} \tag{7}$$

$$\theta_{\mathbf{goal}} = \arccos(\frac{x_g - x_c}{\sqrt{(x_g - x_c)^2 + (y_g - y_c)^2}}) \tag{8}$$

The third step estimates the roughness values by the defined model if A* cannot find proper way. For example, if there are some path consists of small particles less than 0.3 cm, then it can clear the path and make new road.

The Fig. 10 shows that the robot can apply the A* algorithm and make a path planning within 20ms if possible and roughness estimate within 15.35ms. The robot can travel through the available space and keep its distances from the obstacle safely. The robot also can narrow its legs and it can go along the available middle way by not hitting the right and left object. The last example shows that it can pass over the obstacles with lowering its speed safely.

6 DISCUSSION

This part mainly focuses on critical evaluation whether this study [14] mainly focuses on true components and make improvement on the robot performance on terrain segmentation and terrain estimation for the path planning on changing real world environment.

In the related work, the study mentioned about the previous works in different aspects. Some of the project topics for the terrain segmentation were based on Convolutional Neural Network , Conditional Random Field, Fully Convolutional Network on cityscapes dataset, but this study based on less used specific terrain dataset. For the roughness estimation, previous studies are based on image-based approach, CNN, SegNet but this study based on continuous terrain roughness. Also, for the segmentation and estimation, end-to-end solution for the

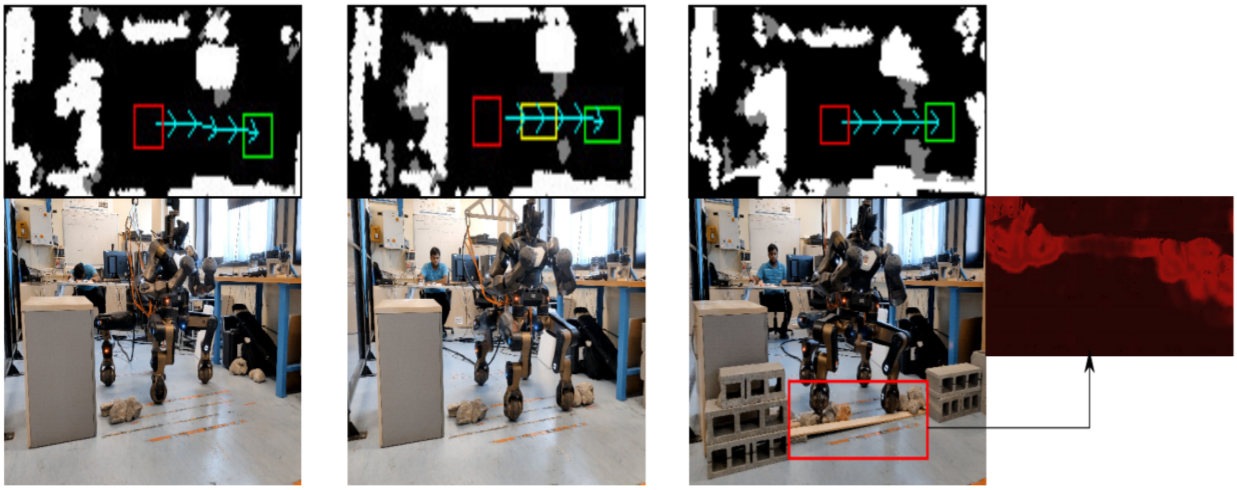

Figure 10

different datasets is never stated on previous studies so that it is very important as a new approach.

The author uses an easy words and readable sentences not to cause fatigue. The organization of the paper is presented well and clear that it mentions how to form the idea and what will be the output and how could be appeared. It gives and example for how to handle bias error problem for the real world environment of robot actions. For example it mentions the properties of the layers, weights and other important measures such that it refers to use of CNN depth, SegNet configurations by layers, activation functions and normalization, and training configurations with losses. Then we can compare the architecture with previous studies on similar models and datasets so that we can see difference between them and how the performance differ.

This paper is mostly related to previous studies to put on new ideas to apply them successfully. Therefore, if other researches have no idea about this topic, first they need to read previous related papers and understand the concept well. The dataset and resource code are available but implementation is not specified well especially there are not enough implementation steps. Therefore it may take more time for non-familiar researches unless they are familiar with these kind of studies. Still source code is understandable by examining the paper.

The experiment indicates the dataset configuration first, then mentions about the three networks results like ENet, SegNet, and ERFNet for the segmentation. The best performance is state-of-the-art method ERFNet among them and it can be improved a bit by freezing the first layer of the CNN. Then for the roughness, Euclidean and BerHu losses are applied and BerHu is better one. Even if the applied changes can affect the performance positively but lower value, it is satisfactory. As it can be seen in S. Ram's study, training the images with the state-of-the-art method with the SegNet, this way seems the good performance output but modifying the number of the layers and bias values may improve the performance slightly.

This method will most probably fail in the places where it contains the different types and having very similar colors of the particles. It can even misinterpret and label the cluttered spaces wrongly because it cannot find a valid path if the space contains so many obstacles in different positions. The A* with these measurements cannot find the instant results or even best path. In their later research [15], they tried to solve this issue by including Trapezium Obstacle Negotiating method that can act more complex behaviors against the cluttered spaces. Therefore, its algorithm should be improved in more difficult and cluttered places.

The author makes some comments on their future works such that they can improve the cameras usage for being the minimum blurring and also instant response for changing the planning of the robot estimation time. Also they plan to do speed improvement and add autonomous path planning by combining the vision-based terrain and different terrain estimation components like torque. So we can say they can apply different future plans apart from their applications and improve the performance.

In addition, this study can consider adding new features for the CENTAURO robot. It uses its legs and center of mass to decrease the loss of the control while traveling but it does not use its hands that much. In the future studies, it can use its hands in a more efficient way such that it can hold the wall and use force support from the wall in the narrow corridors and increase its legs if the robot hands are powerful enough. Then the robot control better and performance can be improved effectively.

7 CONCLUSION

In conclusion, this report gives a summary and critical discussion about the study on the CENTAURO robot that provides end-to-end CNN solution for terrain segmentation and roughness estimation.

The dataset is modified by images of terrain with their roughness values. The procedure and types of measurements are explained for the CNN configuration and added new aspects with different networks. Then test and implementation of the robot are discussed widely The robot alters its position, leg, and body configurations, the difference between a predetermined module and a newly designed module. Then the center of the mass polygon and speed changes of the robot is explained how it applies the A* algorithm with different measurements and reacts in a fast way.

The discussion part makes an assessment that how the authors explained their ideas and what are the strong points and which points should be stronger. For future work, what kind of aspects are put into life after this study.

References

[1] 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.

[2] F. Schilling, X. Chen, J. Folkesson, and P. Jensfelt, "Geometric and Visual Terrain Classification for Autonomous Mobile Navigation," in IEEE/RSJ IROS, 2017, pp. 2678–2684.

[3] R. Garg, B. V. Kumar, G. Carneiro, and I. Reid, "Unsupervised CNN for Single View Depth Estimation: Geometry to the Rescue," in ECCV. Springer, 2016, pp. 740–756.

[4] S. Bell et al., "Material Recognition in the Wild with the Materials in Context Database," in IEEE CVPR, 2015, pp. 3479–3487.

[5] T.-C. Wang et al., "A 4D Light-Field Dataset and CNN Architectures for Material Recognition," in ECCV, 2016, pp. 121–138.

[6] G. Schwartz and K. Nishino, "Material Recognition from Local Appearance in Global Context," CoRR, vol. abs/1611.09394, 2016.

[7] D. B. Gennery, "Traversability Analysis and Path Planning for a Planetary Rover," Aut. Robots, vol. 6, no. 2, pp. 131–146, 1999.

[8] S. B. Goldberg, M. W. Maimone, and L. Matthies, "Stereo Vision and Rover Navigation Software for Planetary Exploration," in IEEE Aerospace Conference, vol. 5, 2002, pp. 2025–2036.

[9] A. Howard and H. Seraji, "Vision-based Terrain Characterization and Traversability Assessment," Journal of Robotic Systems, vol. 18, no. 10, pp. 577–587, 2001.

[10] J. Kim et al., "RGBD Camera Based Material Recognition via Surface Roughness Estimation," in IEEE WACV, 2018, pp. 1963–1971.

[11] S. Ram, “Semantic Segmentation for Terrain Roughness Estimation Using Data Autolabeled with a Custom Roughness Metric,” Master’s thesis, 2018.

[12] A. B. Owen, “A Robust Hybrid of Lasso and Ridge Regression,” Tech. Rep., 2006.

[13] S. M. LaValle, Planning Algorithms. New York, NY, USA: Cambridge University Press, 2006.

[14] V. Suryamurthy, V. Sushrutha Raghavan, A. Laurenzi, N. Tsagarakis, and D. Kanoulas, “Terrain segmentation and roughness estimation using rgb data: Path planning application on the centauro robot,” in IEEE-RAS International Conference on Humanoid Robots. IEEERAS, 2019.

[15] V. Suryamurthy, D. Kanoulas, D.G.Caldwell and N.G. Tsagarakis, ”Agile Legged-Wheeled Reconfigurable Navigation Planner Applied on the CENTAURO Robot”, 2020