applied sciences

MDPI

Article

A Vision-Based Two-Stage Framework for Inferring Physical Properties of the Terrain

Yunlong Dong [1], Wei Guo [1], Fusheng Zha [1,2,*], Yizhou Liu [1], Chen Chen [1] and Lining Sun [1,*]

[1] State Key Laboratory of Robotics and System, Harbin Institute of Technology, Harbin 150001, China; yunlongdong@hit.edu.cn (Y.D.); wguo01@hit.edu.cn (W.G.); liuyizhou@hit.edu.cn (Y.L.); danny_cc@hit.edu.cn (C.C.)

[2] Shenzhen Academy of Aerospace Technology, Shenzhen 518000, China

* Correspondence: zhafusheng@hit.edu.cn (F.Z.); lnsun@hit.edu.cn (L.S.)

Received: 21 August 2020; Accepted: 11 September 2020; Published: 17 September 2020

Abstract: The friction and stiffness properties of the terrain are very important pieces of information for mobile robots in motion control, dynamics parameter adjustment, trajectory planning, etc. Inferring the friction and stiffness properties in advance can improve the safety, adaptability and reliability, and reduce the energy consumption of the robot. This paper proposes a vision-based two-stage framework for pre-estimating physical properties of the terrain. We established a field terrain image dataset with weak annotations. A semantic segmentation network that can segment terrains at the pixel level was designed. Given that the same terrain also has different physical properties, we designed two kinds of image features, and we use a decision-making model to realize the mapping from terrain to physical properties. We trained and tested the network comprehensively, and experimented with the complete framework for estimating physical properties. The experimental results show that our framework has good performance.

Keywords: visual estimation; physical properties; two-stage; field terrain; dataset

1. Introduction

The physical properties of the terrain, including friction properties and stiffness properties, are very important information for autonomous mobile robots. For example, legged robots can use more favorable leg stiffness control to adapt to different terrain after obtaining the terrain stiffness information [1,2]. They can achieve better body balance to walk safely on the terrain with the friction information [3–5]. Wheeled robots can thus avoid excessive wheel slip and sinking. This will save energy, improve safety and reduce odometer error for wheeled robots [6,7]. Overly strong vibrations will seriously affect the accuracy of the robot's camera, lidar, IMU and other sensors [8]. Tracked robots can take measures to effectively reduce the vibrations caused by the contact between the track and the hard terrain after obtaining the stiffness information [9]. Physical properties combined with three-dimensional information can help build a better model of the surroundings, which can be used to generate cost maps and guide mobile robots in motion planning [10,11]. Without understanding the terrain's physical properties, any application based on the principle of terramechanics can not be reliably applied.

Considering the importance of physical properties of the terrain to mobile robots, researchers are committed to estimating terrain physical properties through various methods. Ding et al. [12] established a foot-terrain interaction model and estimated terrain physical parameters through foot–soil interaction experiments. Tsaprounis et al. [13] designed a new friction model including the exponential decay part, and the Coulomb and viscous friction to estimate the friction coefficient. Mohamed Fnadi et al. [14] proposed a new nonlinear observer designed to estimate the contact cornering stiffnesses in real-time. In the above research, only after the robot contacted the terrain could the physical properties of the terrain be obtained.

However, in many robot tasks, such as motion planning, gait adjustment, dynamic model parameter control and so on, the physical properties of untraveled terrain need to be pre-estimated. That is to say, the methods to obtain the physical properties of terrain through the contact between robot and terrain lag behind the task requirements.

Therefore, we should consider an algorithm that can estimate the physical properties of the terrain in advance. The performance of human beings in such tasks gives us some inspiration. Humans can pre-infer some tactile properties (including friction properties, stiffness properties, temperature properties, dry and wet properties, etc.) of the material to some extent simply through visual observation before touching it [15–17]. This reveals that vision is an effective way to acquire the physical properties of objects.

To our excitement, computer vision has grown rapidly with the vigorous development of GPU computing. Some GPU packages, such as Nvidia Jetson TX1 [18], are ideal for onboard processing. Algorithms based on deep learning have achieved excellent results in many fields of vision because of its powerful feature extraction ability [19–21].

Moreover, visual algorithms based on deep learning always needs a large number of data. Scholars have established many image datasets, such as CityScapes [22], Microsoft COCO [23] and so on [24,25], which involve many application fields. However, no image dataset is specifically used to help mobile robots infer the physical properties of terrain. Therefore, this paper establishes a new dataset to deal with such tasks. Perhaps the most similar dataset to ours is COCO-Stuff [26]. Although COCO-Stuff contains many different terrains, there were several reasons to build our new dataset. Firstly, the final purpose of our study was to estimate the friction and stiffness for different terrains, so the terrain was what we were concerned with. However, almost every image in COCO-Stuff contains too many things (i.e., salient objects)—trains, cars, people, etc.—whose friction and stiffness are not what we want to study. Secondly, the existence of irrelevant things is not conducive to the network learning the essential features of the terrain. That is why Schilling et al. [27] clipped the images and focused their attention on the lower image regions. Thirdly, compared with COCO-Stuff, our dataset has more specifications for camera height and field of view, so it is more suitable for most field mobile robots.

Inference tasks of computer vision can be divided into image classification, object recognition, semantic segmentation and instance segmentation from coarse to fine [28]. It is worth mentioning that getting the boundary between different terrains in the task of perceiving the physical properties is very important for mobile robots. Boundary information can provide an accurate workspace for the mobile robot in motion planning, so it is helpful for the robot to choose the optimal path and achieve the optimal state. For example, clear boundaries can help the legged robot accurately select the discrete footholds which are most conducive to its performance. Therefore, this paper considers the problem from the perspective of semantic segmentation. Researchers have designed many excellent networks for semantic segmentation, including FCN [29], SegNet [30], DeepLabv3 [31], ResNet [32], etc. These networks replace the full connection layers which cause spatial information loss with the transposed convolution layers to upsample the feature map, so as to predict at the pixel level. However, current networks are aimed at segmenting targets with specific shapes and specific topologies, such as cars, pedestrians, cups and cats. There is no network designed to segment different terrains, which is one of the motivations for our research.

Additionally, the way of inferring physical properties in this paper draws on the characteristics of human perception. Therefore, the internal mechanism of biology should be considered. Some scholars study the internal mechanism of human perception of the tactile properties through vision. Adelson's research points out that the optical and mechanical properties of material appearance are important clues for humans to perceive the tactile characteristics through vision [33]. Komatsu's research found that an important stage in perceiving material properties is the inverse extraction of image information, which is interpreted as the process of inferring optical properties and surface mesostructure based on retinal images in the visual system [34]. The study also shows that the information retrieved from these inverse extractions is analyzed in the advanced visual stage, and then the brain makes reasonable inferences using this information. These studies enhance the feasibility of using computer vision to perceive the physical properties of the terrain. Their way of studying the problem from a biological perspective has inspired us to extract the features of digital images and establish the reasoning mechanism in a bionic way.

This paper proposes a vision-based two-stage framework to estimate the physical properties (including friction properties and stiffness properties) of the terrain. We expended much effort building a field terrain image dataset(FITI). In our framework's first stage, to take advantage of deep CNNs by extracting deep features of images; we designed a semantic segmentation network named TerrainNet to segment different terrains at the pixel level. In the second stage, from the perspective of bionics, we take the corresponding digital image features and use the decision-making mechanism to realize the mapping from terrain type to friction property and stiffness property. The experimental results show that our framework has good performance.

This article is organized as follows. We introduce the details of our dataset and the vision-based two-stage framework in Section 2. In Section 3, the comprehensive experiments are described; they were done to verify the proposed framework. In Section 4, we summarize the whole article and give the conclusion.

2. Proposed Method

We surveyed several places to determine the typical terrains that need to be photographed and established an image dataset accordingly. The dataset is weakly annotated, which means it is only annotated according to terrain type, rather than having more complex and explicit strong annotations according to physical properties. We introduce the process of the vision-based two-stage framework which is used to estimate the physical properties. In the first stage, a ResNet-based semantic segmentation network named TerrainNet is designed to infer different terrains of the image densely. In the second stage, from the perspective of bionics, we design, extract and analyze image features, and finally realize the mapping from terrain types to physical properties using a decision-making mechanism.

2.1. FITI Dataset

There are many types of terrain in the wild environment, including land, rock, water, etc. Therefore, we need to determine the typical terrain types to build the dataset. We randomly selected 8 common field sites from the coast of Songhua River, several parks and several hills in northern China. After choosing an area of about one hectare in each location (the ratio calculation method used in this paper does not need accurate area), we calculated the ratio of the area of each type of terrain to the total area of 8 sites. Finally, we determined the typical terrain types by sorting the ratios. The details are as follows.

During the investigation, a remote-control drone was used to take pictures at a constant height over an area of about one hectare at each site. We gridded the images captured and counted the number of grid elements of different terrains. Then, we aggregated the number of grids for each terrain in all photographs of all survey sites accordingly and calculated the ratios compared to the total number of grids respectively. The statistical results are shown in Figure 1. It was found that although there are many types of terrain in the wild environment, the typical terrain types (ignoring types that account for less than 5%) are grassland, land, rock, ice, water and asphalt road. Therefore, we built the image dataset with the above 6 types of terrain as the main types. Additionally, the method of determining typical types in this paper can get more accurate statistics with the same labor and time costs.

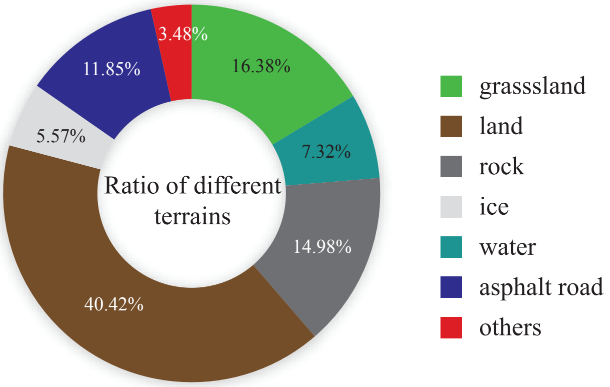
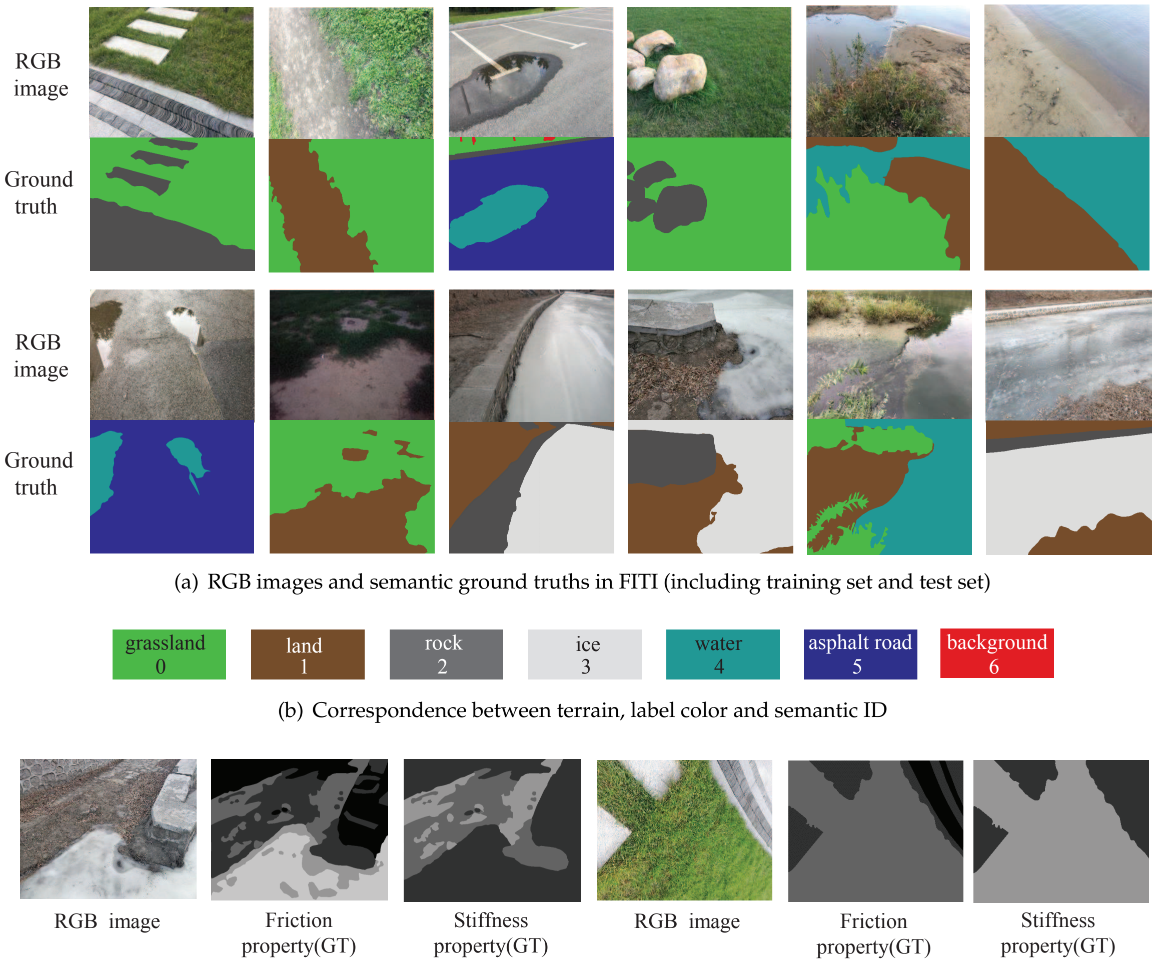


Figure 1. General statistics on the proportions of different terrains in 8 sites.

It is worth noting that the terrain types in this paper are super categories [26]. The super category contains several subcategories, such as land, which contains muddy land, leaf-covered land, etc., as shown in Figures 2 and 3. Therefore, we may call the subcategory of a super category different states of the same terrain type. The Section 2.3 will show more details of different states of terrain and how to deal with the terrain and different states in the process of physical property inference.

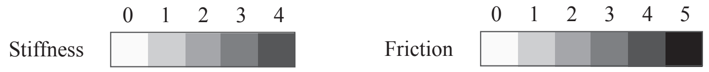


(a) RGB images and semantic ground truths in FITI (including training set and test set)

(b) Correspondence between terrain, label color and semantic ID

(c) RGB images and physical property ground truths in test set

Figure 2. *Cont.*



(d) Correspondence between grayscale and physical property level

Figure 2. The first and third rows of (**a**) are RGB images, and the second and fourth rows are their corresponding ground truths. Ground truths are weakly labeled, that is, they are only labeled according to the terrain types. (**b**) The corresponding relationship between the terrains and the annotation colors. (**c**) The physical property ground truths of the test set. (**d**) The correspondence between grayscale and physical property level. The values of friction coefficient and stiffness corresponding to the friction levels and stiffness levels are presented in Section 2.3.

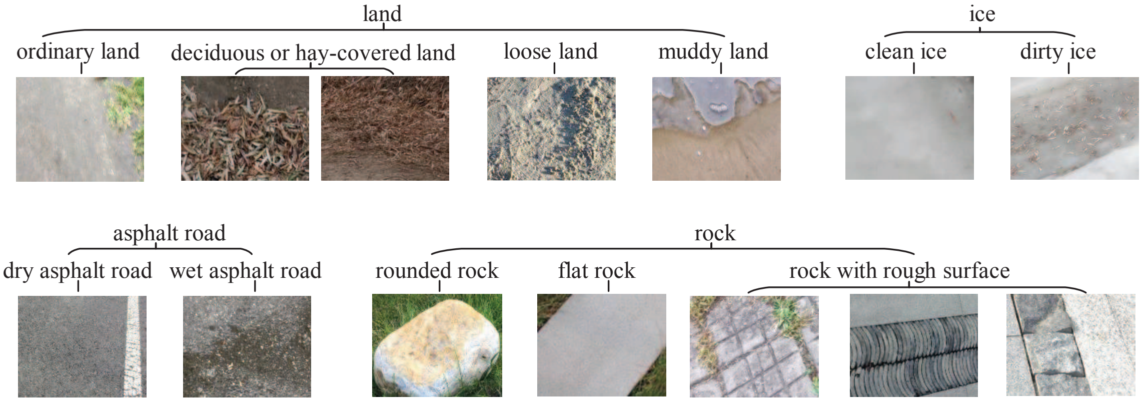


Figure 3. The same terrain may have different states.

Then, we collected RGB images according to certain specifications to build the dataset. It is necessary to pay attention to the camera shooting perspective when collecting the RGB images since the mobile robot has its own special perspective determined by the position of its camera and the posture of its body. In principle, the direction of the camera's optical axis is acceptable as long as it is not perpendicular to the ground.

Visual perception elements of most field mobile robots such as LS3 [35], BigDog [36] and some wheeled robots [37,38] are located between 0.4 and 1.5 m from the horizontal plane when robots maintain upright postures. Considering that the terrain may be uneven or inclined, and the robot is undulating while walking, we extend the above range with a scale factor of 1.2. That is, when collecting RGB images, it is appropriate to control the vertical distance between the lens center of the camera and the ground between 0.33 and 1.80 m. In addition, we intended to control the angle between the optical axis of the camera and the horizontal plane so that the visual field coverage was within 20 m because the texture of the terrain at a long distance is not clear in the image.

In order to better capture the textural information of different types of terrain, it is necessary to ensure that the collected images have high resolution. In our dataset, the resolution of raw RGB images is above 2048 × 1024. Meanwhile, we collected images in different seasons and weather conditions.

We used LabelMe [39] to weakly annotate raw RGB images, which means we just annotated images according to the terrain types, instead of friction properties and stiffness properties. The latter labeling strategy is more complex and labor-consuming. From the training point of view, the latter is more suitable for end-to-end training, but it has serious drawbacks.

For example, unlike some deterministic properties such as classification, size and so on, the friction property of an individual material cannot be determined because friction is produced by the interaction of two materials. Thus, if we want to label the friction properties of the image, we must determine the material that the robot is in contact with in the terrain, such as iron, plastic or rubber. We even need to determine the shape of the sole or the pattern of the wheel. However, labeling in this way would greatly reduce the versatility of the dataset; that is, once we change the material or shape of the sole of a robot, we would need to relabel the whole dataset. In other words, the dataset with strong annotations is narrow in application and more difficult to modify. Moreover, compared to weak annotations, strong annotations make it more difficult to add other physical properties, such as damping, bearing properties and shearing properties, because the entire dataset has to be relabeled. From the perspective of supervision, the high cost of strong annotations also led us to consider a way to reduce supervision. The weak-annotation method in this paper makes the dataset more versatile in the field of mobile robots, although it greatly increases the difficulty of the algorithm. Furthermore, in later parts, we will show that our framework is easy to adjust for different robots or other attribute inferences.

Although we did not build a training set labeled with physical properties to train our model, we had to build the test set with physical property labels to quantitatively evaluate the accuracy of our inferential results. Under the assumption that the sole material of the robot is rubber, the test set is labeled densely according to physical property and stiffness property. That is to say, the training set is composed of RGB images and semantic ground truths, while the test set is composed of RGB images, semantic ground truths, stiffness property ground truths and friction property ground truths. We used the Likert scale to get the discrete representations of physical properties of different terrains and their different states, and calibrate the corresponding friction coefficient and stiffness values. The process is consistent with the process of building labels for the dataset used to train the decision mechanism in Section 2.3. To avoid redundancy, we put the details in the Section 2.3.

In general, we built a field terrain image dataset named FITI that can be used by most mobile robots to sense the physical properties of the terrain in the wild. FITI consists of 3370 high-resolution images. We labeled 7 super categories including land, grassland, rock, asphalt road, water, ice and background. The RGB image and semantic ground truth in the dataset are shown in Figure 2a. We show the correspondence between terrain, label color and semantic ID in Figure 2b. To evaluate the algorithm quantitatively, a test set with semantic and physical properties as labels was established under the assumption that the sole of the robot was rubber. The physical property ground truth of the test set and the correspondence between grayscale and the physical property level are presented in Figure 2c,d. We balanced the number of images for each type to reduce data bias. Moreover, based on some experience [26,40], we divided the training set and testing set according to the ratio of 4:1.

2.2. First Stage: From RGB Image to the Terrain Type

Compared with other image datasets used for semantic segmentation tasks, FITI brings special difficulties to the network. The main reason is that different types of terrain do not have their specific shapes and topologies, unlike many typical segmented objects. Therefore, the neural network we designed had to have a strong ability to learn the texture features of different terrains, which also explains why our dataset requires images to be high-resolution. Moreover, the coexistence and relative positions of different terrains in the same field of vision also have certain constraints.

We chose ResNet-101 as our baseline network due to its superior performance in semantic segmentation. ResNet-101 builds a network of 101 convolutional layers by stacking residual blocks. Its network structure can be divided into two parts according to its functions. The first part is used for downsampling, which occupies the vast majority of the hidden layers, so it has strong feature extraction capabilities. The second part is used for upsampling, which restores the feature map to the same size as the input image through a transposed convolutional layer.

Although ResNet-101 shows good segmentation of objects with specific shapes or specific topologies, such as cars, pedestrians, cups and so on, it cannot be directly used to segment different terrains which have no specific shapes or specific topologies.

First of all, texture features are key information for the network to infer the type of terrain. ResNet is worthy of further improvement to adapt to texture learning in the terrain segmentation tasks of this paper. FITI is built for mobile robots, so the distances between terrain in the field of vision and the camera are different; that is, some terrain is close to the camera, and some terrain is far away from the camera. In this case, the features of the distant terrain may be less than 7 × 7 pixels.

However, ResNet uses a filter with a size of 7 × 7 in the first convolutional layer and this operation will cause the rich information in the input high-resolution image to be truncated at the beginning of the network, as can be seen from Equation (1).

$$(n'_H \times n'_W) = \left(\left\lfloor \frac{n_H + 2p - f}{s} + 1 \right\rfloor \times \left\lfloor \frac{n_W + 2p - f}{s} + 1 \right\rfloor\right) \quad (1)$$

where n_H, n_W, n'_H and n'_W are the height and width of the feature map before and after convolution; f is the size of the convolution kernel; p is the padding number; s is the stride; and $\lfloor . \rfloor$ is the floor function. Thus, it cannot guarantee that subsequent layers can effectively learn the details of the image. What is more, repeated downsampling operations such as convolution and pooling can cause the initial image resolution to decrease significantly, which results in loss of texture details. However, there is no corresponding remedy in ResNet.

Moreover, ResNet does not make good use of the global contextual information of the image, nor does it have sufficient ability to parse the scene in the entire field of view. Therefore, it cannot effectively learn the coexistence and relative positions of different types of terrain. As shown in Figure 4, without a global context clue, an overexposed rock may be mistakenly identified as ice due to the similarity in appearance. These disadvantages need to be addressed.

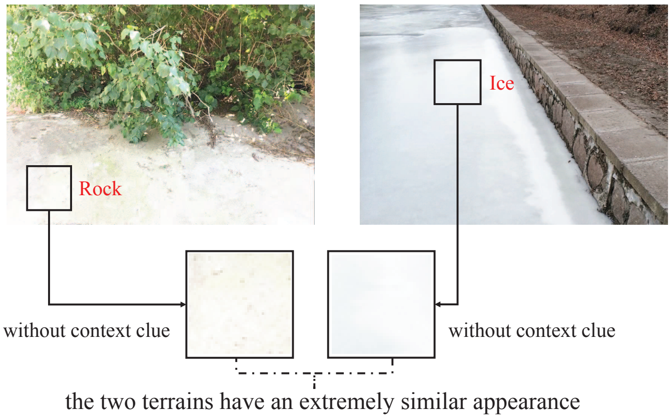


Figure 4. In the local image, without context clues, an overexposed rock and ice have are extremely similar in appearance.

In this paper, we replace the first layer of ResNet with our designed parallel feature extraction structure, so that as much information as possible in the initial high-resolution image enters the network instead of being truncated from the beginning. As shown in Figure 5, there are three parallel operations in our structure, which are convolution, maximum pooling and average pooling, instead of just one convolutional layer. The convolution operation learns the low-level features by updating the convolution kernels. Average pooling and maximum pooling are used to obtain the common features and the most significant features of the corresponding region, respectively. The information obtained by the three parallel pipelines is more abundant than the original single pipeline.

Differently from the filter parameters in the first layer of ResNet, we use a smaller filter size which is 3 and a larger number of filters which is 68 in the downsampling operation. As shown in Equation (1), we obtain three sets of feature maps with larger height and width and more channels after the parallel operation. Finally, we concatenate them in the channel direction to fuse features and output this large feature map to the second layer of the network. Our parallel feature extraction structure is used to improve the network's ability to absorb rich information from the initial high-resolution image and provide the possibility for subsequent layers of effectively learning the details of the terrain image.

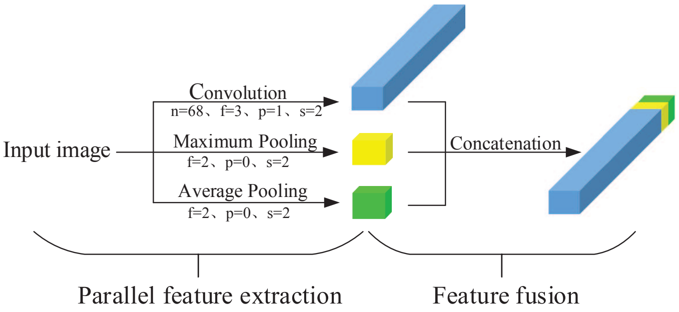


Figure 5. Parallel feature extraction structure.

Besides, we add long skip connections between blocks of ResNet to enhance the network's ability to learn fine texture features. With the repeated downsampling of the feature map layer by layer, tiny spatial details and fine structures are gradually lost in the deep layers. Therefore, we should pay attention to the important role of low-level fine details retained in the shallow layer. At the same time, considering that skip connection can enhance feature propagation in deep networks, we use skip connections to propagate fine features in shallow layers into deep layers. The specific modification of ResNet is shown in Figure 6. We divide the 33 basic residual blocks in the original ResNet into 4 groups according to conv2_x, conv3_x, conv4_x and conv5_x and rename them block 1, block 2, block 3 and block 4. We use long skip connections to pass the feature maps output from block 1 to each block behind it as input, and perform the same operation on other blocks. As a result, the subsequent layers, especially the deep layers inside and behind block 4, can obtain the fine features in the shallow layers by receiving the feature maps from the shallow blocks.

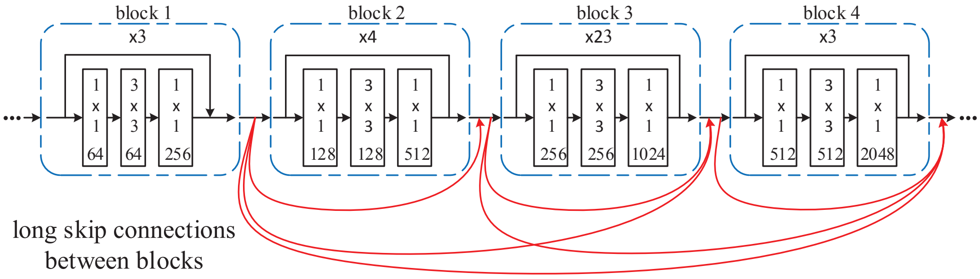


Figure 6. Long skip connections between blocks (red curves).

To enable the network to obtain more contextual information, we insert a pyramid pooling module. Contextual information refers to the information regarding the surrounding environment—a certain type of terrain in an image, including the coexistence of this type and other types in the same image, spatial location relationship, etc. The network with context prior has a strong ability to parse scenes. Sometimes scene information is the key to distinguishing targets with similar appearances. As shown in Figure 7, the images of two kinds of terrain have similar appearances in a small area. If the network only infers the terrain type based on local small area, it will inevitably have poor accuracy. However, if the network can take into account the contextual information of the area to be inferred, it will make a more reasonable inference based on the local and global information. The example shows the superiority of using contextual information for inference.

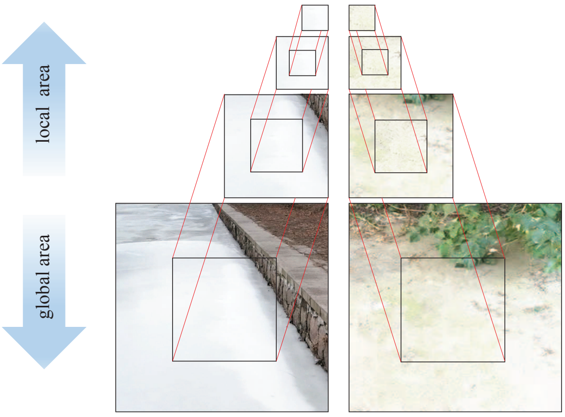


Figure 7. The images at the top show that ice and rock have similar appearances in the local image due to the influence of illumination or other factors. The images at the bottom show that a large perception area has more contextual information. As can be seen from the figure, contextual information is very beneficial for distinguishing types correctly. The red lines indicate the position of the small area in the large area.

We insert the pyramid pooling module behind the downsampling part of the network. As shown in Figure 8, in the pooling operation of each layer in the pyramid, the filter sizes are 64, 32, 16 and 8, respectively. As shown in Equation (2),

$$\begin{aligned} n_i &= \left\lfloor \frac{n_{i-1}+2p-f}{s}+1 \right\rfloor \\ j_i &= j_{i-1}\times s \\ r_i &= r_{i-1}+(f-1)\times j_{i-1} \\ start_i &= start_{i-1}+\left(\frac{f-1}{2}-p\right)\times j_{i-1} \end{aligned} \tag{2}$$

where j is the distance between two adjacent features, $start$ is the center coordinate of the upper left feature and r is the receptive field size. The larger the filter size, the larger the receptive field, which means that the network tends to perceive the global area and more contextual information. Filters of different sizes make the network extract the information of areas of different sizes. Moreover, the pyramid fuses feature maps from different pooling operations. It aggregates contextual information from different regions, thereby giving the networks a comprehensive perception of local and global information.

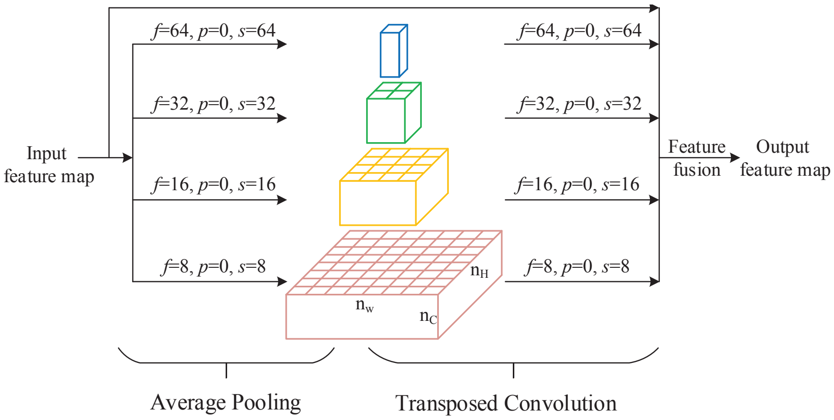


Figure 8. Pyramid pooling module.

Figure 9 shows the complete network architecture called TerrainNet. We use TerrainNet to realize the inference from RGB image to terrain type.

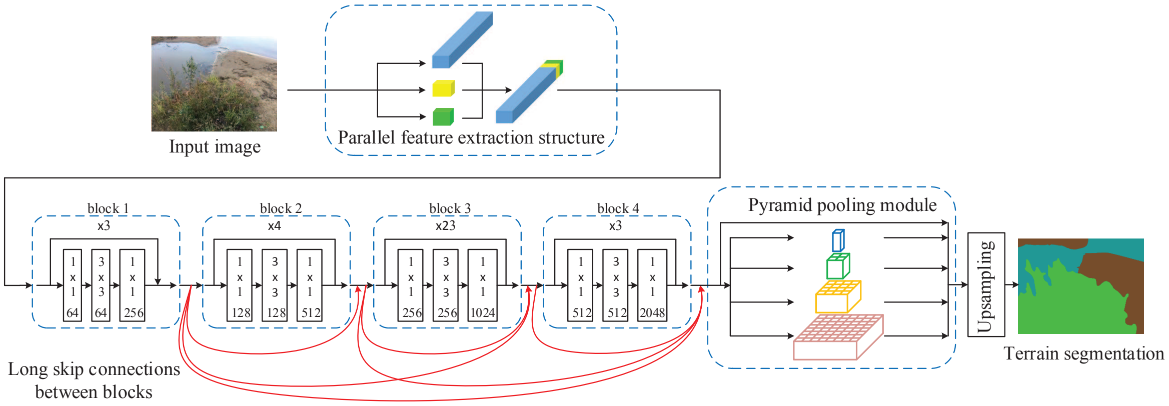


Figure 9. Complete structure of TerrainNet.

Additionally, we use the Adam [41] optimization algorithm to update the parameters, as shown in Equation (3).

$$
\begin{aligned}
m_t &= \beta_1 m_{t-1} + (1-\beta_1) g_t \\
v_t &= \beta_2 v_{t-1} + (1-\beta_2) g_t^2 \\
\hat{m}_t &= \frac{m_t}{1-\beta_1^t} \\
\hat{v}_t &= \frac{v_t}{1-\beta_2^t} \\
\theta_t &:= \theta_{t-1} - \alpha \frac{\hat{m}_t}{\sqrt{\hat{v}_t} + \epsilon}
\end{aligned}
\tag{3}
$$

where g_t is the gradient, θ_t is the parameter to update, m_t is the biased first moment estimate, v_t is the biased second raw moment estimate, $\hat{m}_t$ is the bias-corrected first moment estimate, $\hat{v}_t$ is the bias-corrected second raw moment estimate, α is the learning rate, β_1, $\beta_2 \in [0, 1]$ are the exponential decay rates for the moment estimates and ϵ is used to prevent the denominator from being zero. The details of the network's hyperparameter settings will be presented in section "Experiment".

2.3. Second Stage: From the Terrain Type to Physical Properties

The friction and stiffness properties of the same terrain are also related to its state, such as the surface flatness, compactness, whether the surface has a covering, etc. That means the same type of terrain has different physical properties in different states, as shown in Figure 3, so we need to explore the mapping from terrain types to physical properties. Related biological studies show that optical characteristics and surface structure characteristics are important reference factors for humans to perceive the physical properties of materials through vision. Therefore, we designed two corresponding digital image features to imitate those two characteristics.

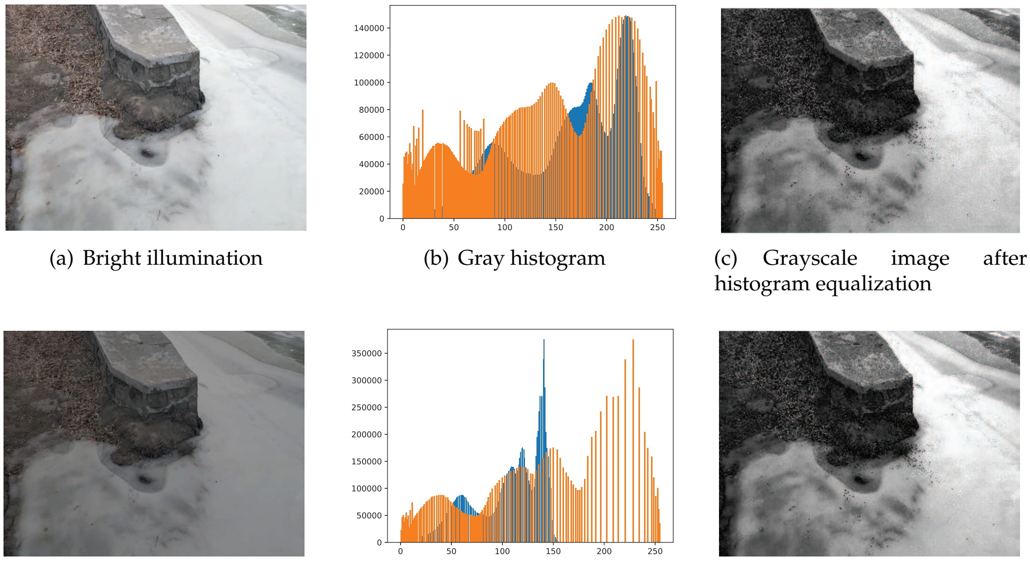
The optical characteristics of an image are a complex set of many attributes, such as brightness, transparency, color and so on. We inferred the type of terrain in the previous section. Consequently, with this strong prior, it was not necessary to design corresponding feature representations for all attributes in optical characteristics. We chose brightness to represent the optical characteristics and designed a method to extract the brightness of the image.

The gray value of each pixel in the image can represent the brightness of it, which ranges from 0 to 255. In addition, Gaussian blur can approximately mimic certain perceptual characteristics of the human eyes; that is, the brightness evaluation of a point is affected by the brightness around the point. Therefore, as shown in Equation (4), the gray-scale image corresponding to the RGB image of the terrain is Gaussian blurred, so as to extract the brightness features of the image.

$$L = I * F \tag{4}$$

where L is a Gaussian blurred image, I is the gray image, F is the Gaussian kernel and * represents convolution.

It should be noted that in order to improve the robustness of brightness features to the changes of ambient illumination, we perform histogram equalization on the image before extracting the brightness features, as shown in Figure 10.



(a) Bright illumination

(b) Gray histogram

(c) Grayscale image after histogram equalization

(d) Dark illumination

(e) Gray histogram

(f) Grayscale image after histogram equalization

Figure 10. Histogram equalization reduces the sensitivity to the changes of the overall environment illumination. The illumination of (**d**) is much darker than that of (**a**), but histogram equalization greatly reduces the illumination difference between the two images. (**c**,**f**) are the processed image. In (**b**,**e**), the blue bars represent the original gray distribution, and the orange bars represent the processed gray distribution.

The surface structure characteristic refers to the changes in depth, direction and arrangement of the material surface. Significant changes in the pixel values of material images usually reflect important events and changes in material properties, generally including discontinuities in depth, discontinuities in surface orientation and changes in physical properties. Therefore, we can describe surface structure characteristics by the variation of the pixel value. Moreover, the gradient of a pixel in an image is the variation of the pixel value in its neighborhood. Thus, this paper uses the Sobel [42] operator to calculate the image gradient to represent the surface structure characteristics.

Sobel is a first-order gradient calculation operator, and its calculation formula is shown in Equation (5).

$$\begin{aligned} G_x &= I * \begin{bmatrix} -1 & 0 & +1 \\ -2 & 0 & +2 \\ -1 & 0 & +1 \end{bmatrix} \\ G_y &= I * \begin{bmatrix} +1 & +2 & +1 \\ 0 & 0 & 0 \\ -1 & -2 & -1 \end{bmatrix} \\ G &= \sqrt{G_x^2 + G_y^2} \end{aligned} \tag{5}$$

where G_x and G_y are gradients in X and Y directions respectively, and G is the gradient of the pixel.

To verify the effectiveness of the two feature extraction methods, we used them to extract the corresponding features of the terrain image, and performed statistical analysis.

As shown in Figure 3, each terrain of land, rock, ice and asphalt road has different states. Taking land as an example, we can see that it has four different states in the dataset, namely, ordinary land, deciduous or hay-covered land, loose land and muddy land. It is these different states that lead to different friction and stiffness properties of the same type of terrain. For the sake of description, we name the land in different states land 1, land 2, land 3 and land 4 in order. We extracted the brightness features of the land 1 images, and then gathered statistics. The details are as follows.

First, we selected 50 RGB images containing land 1 from the FITI dataset and cropped the corresponding parts of land 1 into many image patches with a resolution of 100 × 100. Then, a large number of image patches (ours is 300) were randomly selected to extract the brightness features, and the same number of brightness patches were generated. After that, we calculated the average brightness in each brightness patch. Finally, by fitting all the averages using Equation (6), we obtained a normal distribution that can describe the optical characteristics of land 1. In addition, land 2, land 3 and land 4 were treated the same.

$$p = \frac{1}{\sqrt{2\pi\sigma}} e^{-\frac{(v-\mu)^2}{2\sigma^2}} \tag{6}$$

where v is the average value, μ is the mean and σ^2 the variance.

In the same way, we extracted the brightness features of ices, rocks and asphalt roads in different states and gathered statistics. Of course, we treated the surface structure characteristics equally.

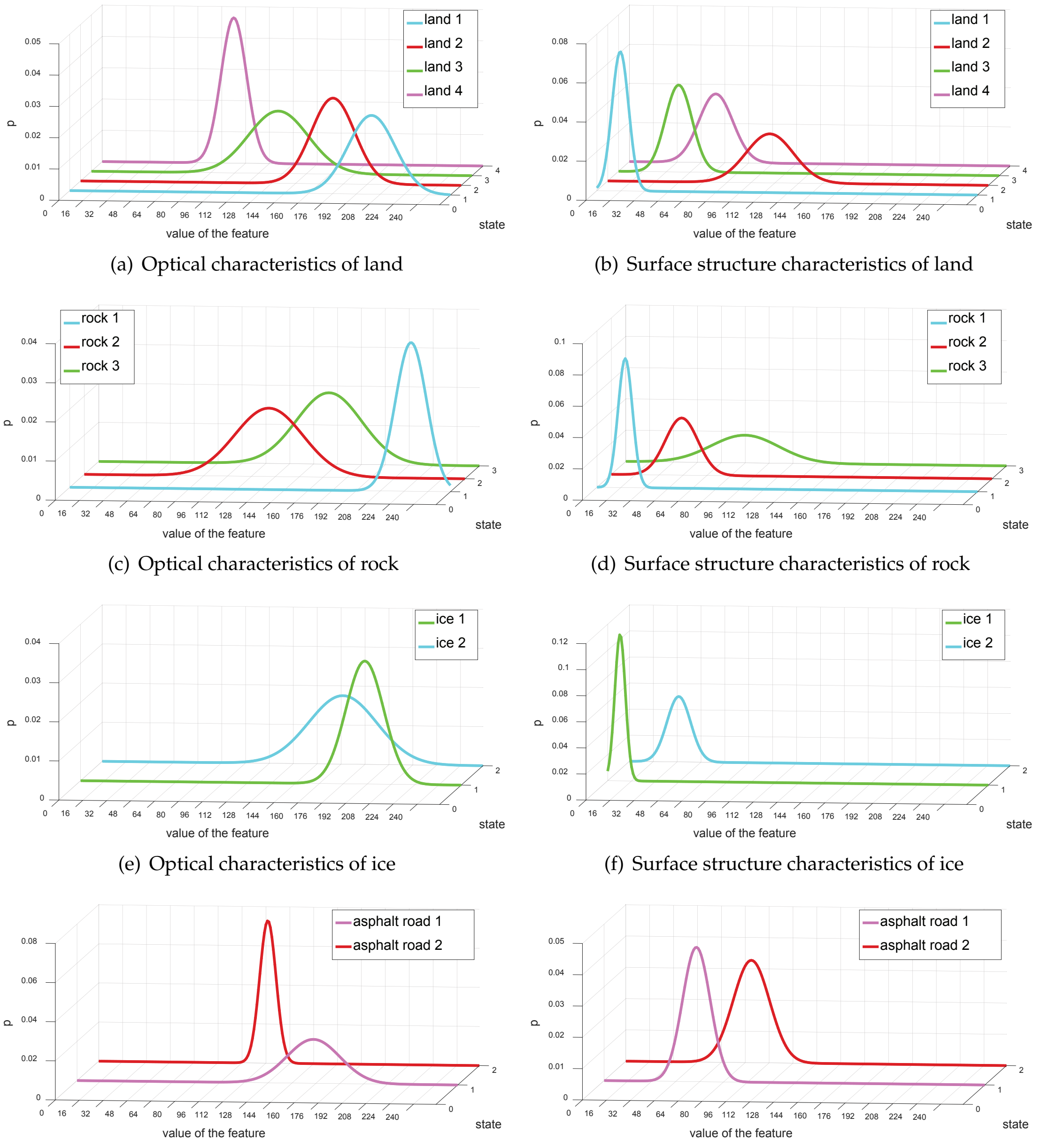
After the above steps, we got the Gaussian distributions of optical characteristics and surface structure characteristics of land, ice, rock and asphalt road in different states. Their respective means μ and variances σ^2 are shown in Tables 1 and 2, and their respective curves are shown in Figure 11.

Table 1. Means and variances of optical characteristics.

Terrain in Different States	μ	σ
land 1	202.2	15.9
land 2	169.1	14.6
land 3	125.2	20.0
land 4	88.1	8.6
rock 1	228.6	10.5
rock 2	123.7	22.8
rock 3	154.3	21.9
ice 1	190.4	12.7
ice 2	161.2	22.8
asphalt road 1	158.4	17.8
asphalt road 2	113.3	5.5

Table 2. Means and variances of surface structure characteristics.

Terrain in Different States	μ	σ
land 1	15.7	5.6
land 2	108.6	15.9
land 3	40.3	8.9
land 4	57.9	11.4
rock 1	19.0	4.8
rock 2	47.1	10.9
rock 3	79.8	22.6
ice 1	8.3	3.6
ice 2	33.1	7.9
asphalt road 1	61.8	9.3
asphalt road 2	84.1	12.2



(a) Optical characteristics of land

(b) Surface structure characteristics of land

(c) Optical characteristics of rock

(d) Surface structure characteristics of rock

(e) Optical characteristics of ice

(f) Surface structure characteristics of ice

(g) Optical characteristics of asphalt road

(h) Surface structure characteristics of asphalt road

Figure 11. The Gaussian distributions of optical characteristics and surface structure characteristics of land, rock, ice and asphalt road in different states.

In the subgraphs of Figure 11, each curve represents the corresponding terrain in a certain state. The position on the horizontal axis and shape of the curve reflect the distribution regularity of the feature of the terrain in this state. We can see that the waves of different curves in the same subgraph can be staggered significantly, which means terrain in different states has good distinguishability. Therefore, it is reasonable to use the above feature extraction methods to extract the optical characteristics and surface structure characteristics of the terrain image.

To reduce the impact of the instability of a single feature during inference, we established a decision mechanism to achieve the mapping from terrain types to physical properties with both feature extraction methods.

We obtained 3300 data points of optical characteristics and 3300 data points of surface structure characteristics through the previous steps. To better quantify these two kinds of features, we used *k*-means algorithm to cluster the data corresponding to each feature separately. Based on the prior information of the Gaussian distribution, we set a more reasonable parameter *k*.

We can take the data of optical characteristics as an example to illustrate the setting of *k*. In the Gaussian distribution, the probability of data distribution is 68.27% in the range of ($\mu - \sigma$, $\mu + \sigma$), and 95.45% in the range of ($\mu - 2\sigma$, $\mu + 2\sigma$). Therefore, we can take $\mu - \sigma$ and $\mu + \sigma$ of all optical characteristics in Table 1 as the initial segmentation positions. Then better positions are selected as the final segmentation positions of optical characteristics in the range of ($\mu - 2\sigma$, $\mu - \sigma$) and ($\mu + \sigma$, $\mu + 2\sigma$). Accordingly, the optical characteristic value is divided into 11 segments in the range of 0 to 255 and we set *k* in *k*-means to 11. After clustering, the cluster number and the range of optical characteristic values are shown in Table 3. Additionally, we can do the same for the surface structure characteristics. The clustering results are shown in Table 4.

Table 3. Cluster number and the range of optical characteristic values.

Range of Value	0–64.2	64.2–90.0	90.0–112.8	112.8–128.3	128.3–150.8	150.8–176.4	176.4–192.3	192.3–208.2	208.2–224.5	224.5–246.9	246.9–255
cluster number	1	2	3	4	5	6	7	8	9	10	11

Table 4. Cluster number and the range of surface structure characteristic values.

Range of Value	0–31.7	31.7–47.4	47.4–70.3	70.3–95.7	95.7–129.0	129.0–146.1	146.1–255
cluster number	1	2	3	4	5	6	7

To establish the mapping to physical properties, we describe the friction properties and stiffness properties in a discrete manner. We employed a method similar to that of [6]. Ten masters of the laboratory were asked to use the Likert scale to judge the level of friction and stiffness of each terrain in different states of the dataset. We used numbers 1–6 for the friction level (i.e., 1 most slippery; 6 least slippery) and 1–5 for the stiffness level (i.e., 1 most deformable; 5 least deformable). Then we measured the friction coefficient between terrains in each level and the robot foot (rubber material) and calibrated the corresponding friction coefficient of each level. We obtained the stiffnesses of the terrains [12,43] and calibrated them as well. The levels of friction and stiffness and the corresponding physical values are shown in Tables 5 and 6. It should be noted that we assumed that the robot does not walk in the water, so we set the friction level and stiffness level of the water to 0. A level of 0 means the most dangerous and not suitable for walking. We also set the friction level and stiffness level of the unknown background the same way.

Table 5. Friction coefficient and level.

Level	**1**	**2**	**3**	**4**	**5**
friction coefficient	<0.1	0.1–0.25	0.25–0.5	0.5–0.7	0.7–0.8

Table 6. Stiffness and level.

Level	1	2	3	4
stiffness (Nm^{-1})	5.7×10^4	1.6×10^5–3.4×10^5	1.7×10^6	2.3×10^7–3.4×10^9

Using semantic features, optical characteristics and surface structure characteristics as elements of the input vectors, and corresponding friction levels and stiffness levels as output values, we establish a dataset of mapping relationships from terrain types to physical properties. Table 7 shows some examples of this dataset. We then feed these data to CART [44]. CART is a decision-making model, which is the abbreviation of classification and regression tree. It selects the best feature and the best cut point by calculating the Gini index shown in Equations (7) and (8) iteratively.

$$Gini(D) = 1 - \sum_{j=1}^{J} (\frac{|C_j|}{|D|})^2 \tag{7}$$

where D is the training set; C_j is a subset of samples belonging to class j. Then $Gini(D)$ is used to calculate $Gini(D_1)$ and $Gini(D_2)$ in Equation (8):

$$Gini(D, A) = \frac{|D_1|}{|D|} Gini(D_1) - \frac{|D_2|}{|D|} Gini(D_2) \tag{8}$$

where D_1 and D_2 are subsets of D divided by the value of feature A. Finally, we get two decision trees for inferring friction properties and stiffness properties, respectively.

Table 7. Some examples of the dataset for mapping relationships from terrain types to physical properties.

Semantic ID	Cluster Number (Optical)	Cluster Number (Surface Structure)	Friction Level	Stiffness Level
0	—	—	3	2
1	7,8,9,10,11	1	4	3
1	6,7,8,9	4,5,6,7	3	2
1	4,5,6	2,3	4	2
1	2,3	3,4	2	1
2	8,9,10,11	1,2	3	4
2	2,3,4,5	2,3	4	4
2	5,6,7,8	4,5,6	5	4
3	6,7,8,9	1	1	4
3	3,4,5,6,7,8	2,3	2	4
4	—	—	0	0
5	6,7,8,9,10	2,3,4	5	4
5	3,4,5	4,5	3	4
6	—	—	0	0

Through the above steps, the algorithm realizes the mapping from terrain type to physical properties.

Last but not least, we only need to modify the last two columns of Table 7 when faced with different robot walking parts or adding inferences of other attributes (such as damping, bearing properties and shearing properties). This is a simple operation, because we do not need to relabel the image dataset. Modifying Table 7 is much easier than modifying the image dataset at the pixel level. This shows that our framework is easier to adapt to different situations.

3. Experiment

To evaluate the effectiveness of the proposed method, we conducted comprehensive experiments. We used the deep learning framework Pytorch to build TerrainNet in Ubuntu 18.04, and used the FITI dataset to train and test the network on the dual Titan XP GPUs. We also analyzed the contributions of network components to the final accuracy. Finally, we tested the complete vision-based two-stage framework of estimating the physical properties of terrains in the field.

3.1. TerrainNet Experiment

After building the TerrainNet based on Pytorch, we used FITI to train and test it on our workstation. For a practical deep learning system, the devil is always in the details. We carefully set the hyperparameters. Batch-size was set to 4 in the mini-batch gradient descent. β_1 was set to 0.9, β_2 was set to 0.999 and ϵ was set to 10^{-8} in the Adam optimization algorithm. We trained a total of 200 epochs, and finally decided that 150 was the best epoch number. The weight decay parameter λ in L2 regularization was set to 2×10^{-4}. The initial learning rate was set to 5×10^{-4}, the gamma value was set to 0.5 and the learning rate decays once every 50 epochs. The input image had a resolution of 2048×1024. The configuration of the workstation was as follows: two NVIDIA TITAN Xp GPUs with 11G memory respectively; single Intel (R) Core (TM) i9-7940X CPU with 16G memory.

The performance of our network on the training and test sets is shown in Figure 12. It can be seen from the figure that the mean of class-wise intersection over union ($mIoU$, Equation (9)) and loss value gradually converge.

$$mIoU = \frac{1}{m+1} \sum_{i=0}^{m} \frac{p_{ii}}{\sum_{j=0}^{m} p_{ij} + \sum_{j=0}^{m} p_{ji} - p_{ii}} \tag{9}$$

where m + 1 is the total number of terrain types and p_{ij} is the number of pixels of class i inferred to belong to class j. That is to say, p_{ii} represents the number of true positives, while p_{ij} and p_{ji} are the false positives and false negatives respectively. The final $mIoU$ accuracy of the network on the training set was 66.52% and the accuracy on the test set was 62.94%.

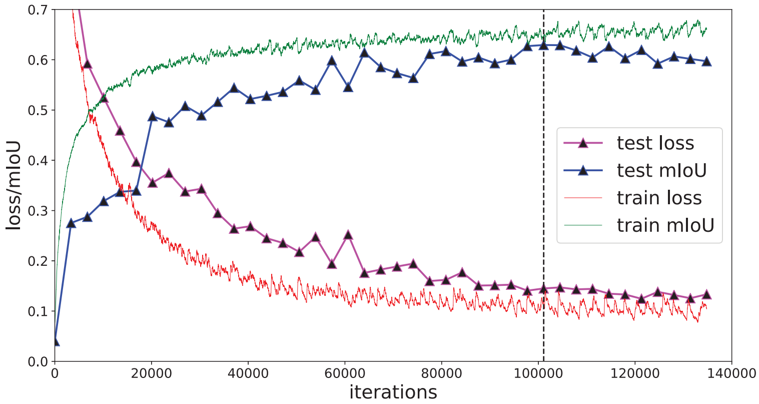
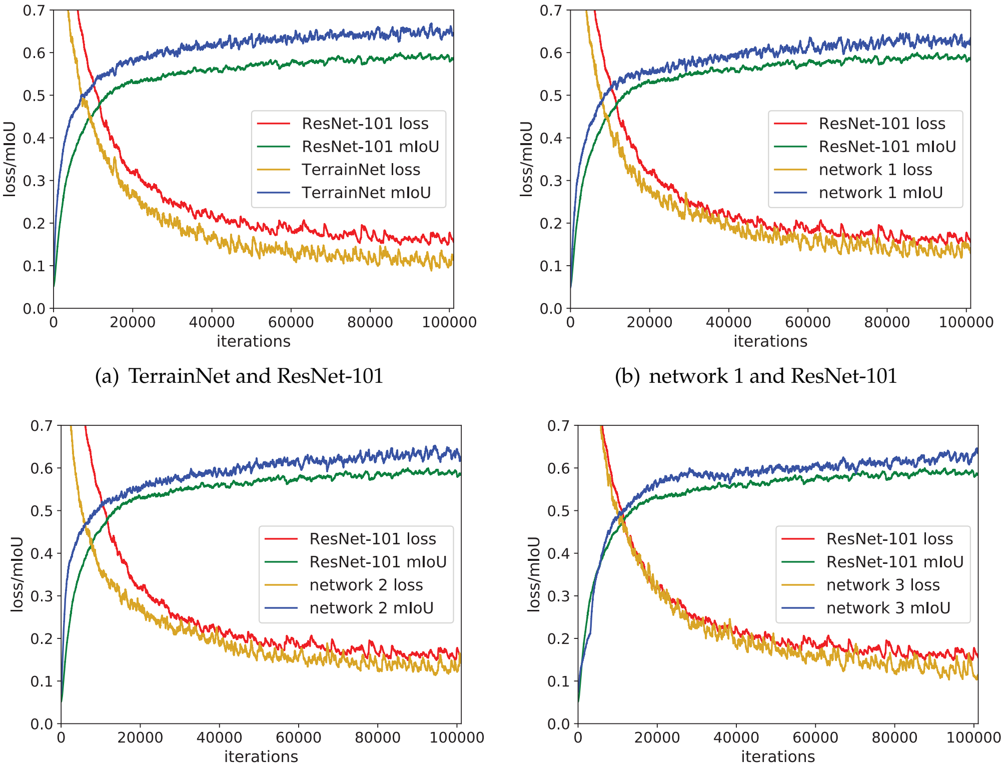


Figure 12. Training and testing curves of TerrainNet. The black vertical dotted line indicates the number of iterations corresponding to the best epoch number.

To better evaluate our network and analyze the contributions of network components in the final accuracy, we carried out ablation experiments with several settings, including raw ResNet-101, ResNet-101 with a parallel feature extraction structure (network 1), ResNet-101 with long skip connections between blocks (network 2) and ResNet-101 with a pyramid pooling module (network 3). Moreover, the hyperparameters of them are consistent with those of TerrainNet. The experimental results are shown in Figure 13 and Table 8. Figure 13 shows the iterative curves of the networks on the training set, and Table 8 shows the final $mIoU$ of the networks on the training set and the test set.

It can be seen that the performances of network 1, network 2 and network 3 in both the training set and test set improved compared with the raw ResNet. This shows the advantages of the parallel feature extraction structure, long skip connections between blocks and the pyramid pooling module. Moreover, the TerrainNet designed in this paper, which includes the above three structures, had a greater performance improvement. $mIoU$ of TerrainNet on the test set was 62.94%–10.68% higher than ResNet's.



(a) TerrainNet and ResNet-101

(b) network 1 and ResNet-101

(c) network 2 and ResNet-101

(d) network 3 and ResNet-101

Figure 13. Iterative curves of the networks on the training set.

Table 8. Ablation experiments on FITI.

Network Structure	mIoU (Train)	mIoU (Test)	Improvement (Test)
ResNet-101 (baseline)	58.89%	52.26%	—
network 1	63.31%	59.08%	+6.28%
network 2	64.11%	60.39%	+8.13%
network 3	63.72%	59.27%	+7.01%
TerrainNet	66.52%	62.94%	+10.68%

To show the advantages of our TerrainNet more intuitively, we compare the semantic segmentation effect of ResNet-101 with that of the TerrainNet in Figure 14. We can see from the figure that the TerrainNet has better performance. For example, TerrainNet does not mistake the rock for ice in the image of the fifth column. As can be seen in the second column, the segmentation of water by our TerrainNet is more correct. In the first, third, fourth and sixth columns, due to the learning of fine texture features of high-resolution images, TerrainNet can infer land, rock and asphalt road more accurately.

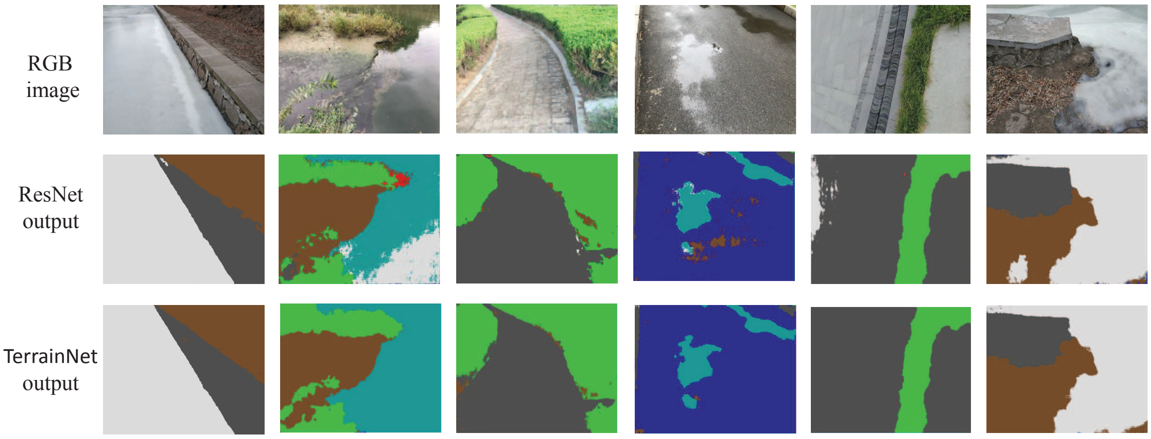


Figure 14. Comparison of semantic segmentation between ResNet-101 and TerrainNet. The second row is the prediction output by ResNet-101, and the third row is the prediction output by TerrainNet. The segmentation effect of the third row is better than that of the second row.

We calculated the IoU of each terrain to determine which terrain was the most difficult to identify. The results are shown in Table 9. We can see that the IoUs of ice and water were relatively low. This may have been due to the fact that both of them belonged to non-Lambert surfaces.

Table 9. IoU of each terrain.

Terrain	Grassland	Land	Rock	Ice	Water	Asphalt Road
IoU	79.10%	63.74%	65.24%	57.56%	53.38%	64.02%

We also compared TerrainNet with FCN, SegNet and DeepLabv3 on FITI, as shown in Table 10. It can be seen that TerrainNet had the best accuracy on FITI. TerrainNet outperformed FCN and SegNet by a clear margin. Although the $mIoU$ of DeepLabv3 on the training set was slightly lower than that of TerrainNet, the accuracy on the test set was far worse. This shows that TerrainNet is better at segmenting terrains than these networks used to segmenting things.

Table 10. Comparative experiments with other networks on FITI.

Network	mIoU (Train)	mIoU (Test)
FCN	52.31%	49.64%
SegNet	50.32%	47.45%
DeepLabv3	63.84%	60.24%
TerrainNet	66.52%	62.94%

3.2. Vision-Based Two-Stage Framework Experiment

Combined with the decision trees established in the previous section, we tested the vision-based two-stage framework proposed for estimating the terrain's physical properties in the field.

As shown in Figure 15, we visualized the experimental results of the framework. The figure shows the inferences in the test set. In the results of friction property inference, the brighter the color is, the lower the friction level is, that is, the more slippery it is. In the same way, in the results of stiffness property inference, the brighter the color is, the lower the stiffness level is. The values of friction coefficient and stiffness corresponding to friction levels and stiffness levels are shown in Tables 5 and 6.

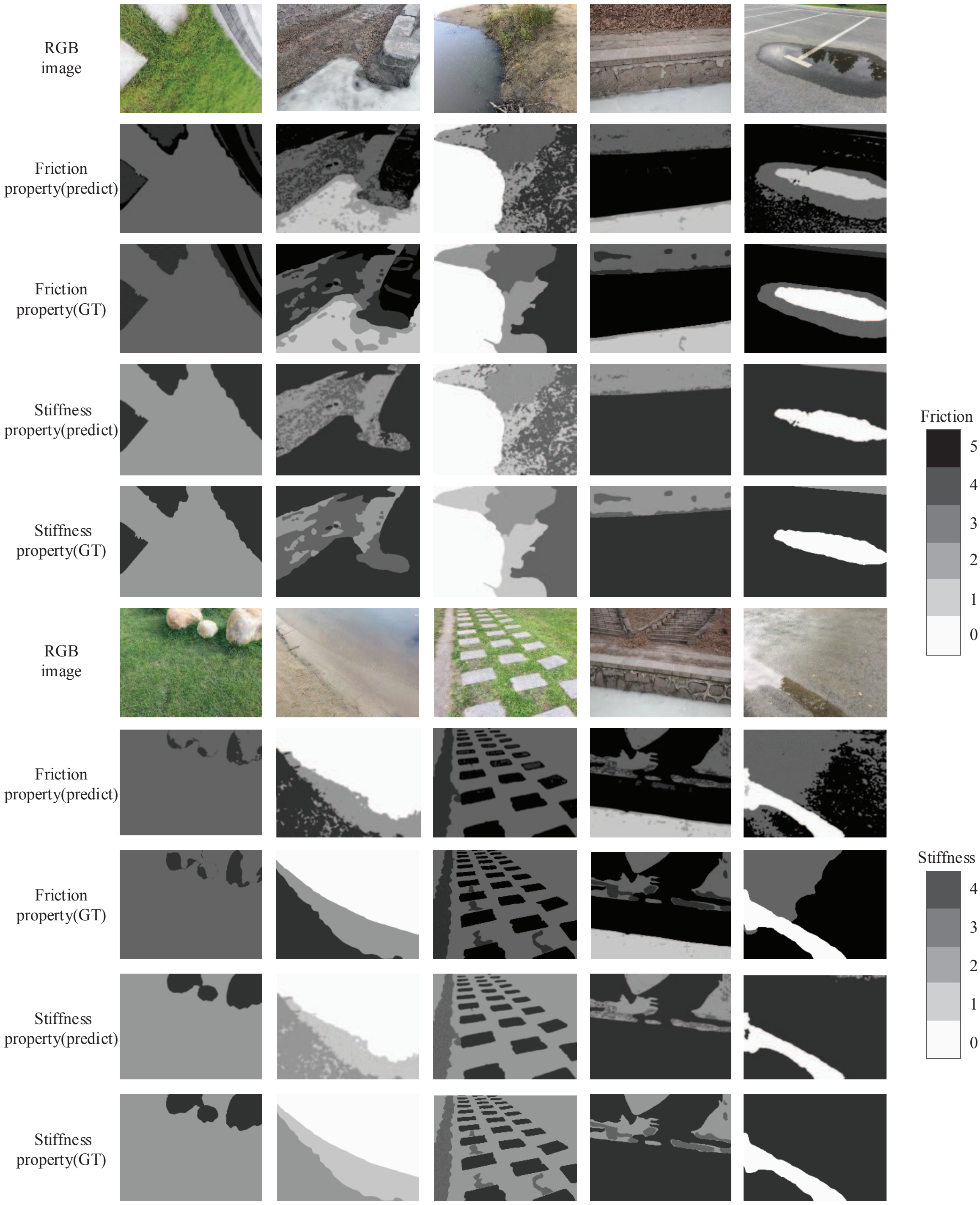


Figure 15. The experimental results of the complete framework proposed in this paper. The first and sixth rows are the input RGB images; the second and seventh rows are the inference results of friction property; the third and eighth rows are the ground truths of friction property; the fourth and ninth rows are the inference results of stiffness property; and the fifth and tenth rows are the ground truths of stiffness property. For better visualization, we mapped the inferential results of physical properties level to a more easily observed gray-scale image. The corresponding relationship between friction level and grayscale is shown in the rightmost column, and so is the stiffness level.

It can be seen from the figure that the framework has good inference performance. The framework can correctly infer the friction level of the dirty ice's surface and the clean ice's surface. The physical properties of the land in different states can also be well inferred; for example, the pixel brightness of the stiffness inference results of deciduous or hay-covered land is higher than that of the ordinary land. The friction levels of dry asphalt road and wet asphalt road can also be well distinguished. The regions with different friction properties on the rock surface can also be well segmented.

We tested the accuracy of our vision-based two-stage framework quantitatively. The physical properties of all images in the test set were estimated by our two-stage framework, and the ground truths of physical properties were used to quantitatively evaluate the inference results. We used $mIoU$ and pixel accuracy (PA) as the metrics to evaluate the accuracy of our framework. The calculation of PA is shown in Equation (10).

$$PA = \frac{\sum_{i=0}^{m} p_{ii}}{\sum_{i=0}^{m} \sum_{j=0}^{m} p_{ij}} \tag{10}$$

where the symbols have the same meaning as in Equation (9).

The experimental results can be seen in Table 11. The $mIoU$ of our framework was 60.18% in the inference of friction property and 61.21% in the inference of stiffness property. The PA of our framework was 75.85% in the inference of friction property and 76.63% in the inference of stiffness property.

Table 11. Accuracy of the vision-based two-stage framework.

Property	PA	$mIoU$
friction property	75.85%	60.18%
stiffness property	76.63%	61.21%

4. Conclusions

In many mobile robot tasks, inferring the friction and stiffness properties in advance is very beneficial. In this article, we proposed a vision-based two-stage framework for the physical property estimation of terrain. Considering the lack of a relevant dataset, an image dataset FITI for terrain segmentation was established. In the first stage of the framework, we designed a corresponding terrain segmentation network named TerrainNet to infer the types of different terrains of the image densely. TerrainNet has three novel components. We carried out experiments and the results show that the $mIoU$ of the proposed network is 10.68% higher than that of the baseline ResNet-101, and has advantages over FCN, SegNet and DeepLabv3. In the second stage, we realize the mapping from the terrain type to physical properties. Our two-stage framework can infer the physical properties of the same terrain in different states. Finally, a complete experiment proved the validity of the vision-based two-stage framework. We evaluated our framework with quantitative metrics $mIoU$ and PA. The results show that our framework has high accuracy.

In future work, we will consider the combination of physical properties and three-dimensional information, and realize the navigation of the mobile robot.

Author Contributions: Conceptualization, Y.D. and F.Z.; methodology, Y.D.; software, C.C.; validation, Y.D. and Y.L.; formal analysis, W.G.; investigation, L.S.; resources, F.Z.; data curation, F.Z.; writing–original draft preparation, Y.D.; writing–review and editing, Y.D.; visualization, Y.L.; supervision, W.G.; project administration, W.G.; funding acquisition, F.Z. All authors have read and agreed to the published version of the manuscript.

Funding: This work was supported by the Natural Science Foundation of China (grant number 61773139), The Foundation for Innovative Research Groups of the National Natural Science Foundation of China (grant number 51521003), Shenzhen Science and Technology Research and Development Foundation (grant number JCYJ20190813171009236) and Shenzhen Science and Technology Program (grant number KQTD2016112515134654).

Conflicts of Interest: The authors declare no conflict of interest.

References

1. Reggeti, J.C.A.; Armada, E.G. Parameter Identification and Modeling of Contact Properties for Robotic Applications. Ph.D. Thesis, Universidad Politécnica de Madrid, Madrid, Spain, 2017.
2. Miller, B.D.; Cartes, D.; Clark, J.E. Leg stiffness adaptation for running on unknown terrains. In Proceedings of the 2013 IEEE/RSJ International Conference on Intelligent Robots and Systems, Tokyo, Japan, 3–7 November 2013; pp. 5108–5113.
3. Lin, Y.C.; Ponton, B.; Righetti, L.; Berenson, D. Efficient humanoid contact planning using learned centroidal dynamics prediction. In Proceedings of the 2019 International Conference on Robotics and Automation (ICRA), Montreal, Canada, 20–24 May 2019; pp. 5280–5286.
4. Giftsun, N.; Del Prete, A.; Lamiraux, F. Robustness to Inertial Parameter Errors for Legged Robots Balancing on Level Ground. In Proceedings of the 2017 International Conference on Informatics in Control, Automation and Robotics, Madrid, Spain, 27 July 2017; hal-01533136.
5. Herzog, A.; Rotella, N.; Mason, S.; Grimminger, F.; Schaal, S.; Righetti, L. Momentum control with hierarchical inverse dynamics on a torque-controlled humanoid. *Auton. Robot.* **2016**, *40*, 473–491. [CrossRef]
6. Konduri, S.; Orlando, E.; Torres, C.; Pagilla, P.R. Effect of wheel slip in the coordination of wheeled mobile robots. *IFAC Proc. Vol.* **2014**, *47*, 8097–8102. [CrossRef]
7. Wang, C.; Meng, L.; She, S.; Mitchell, I.M.; Li, T.; Tung, F.; Wan, W.; Meng, M.Q.H.; de Silva, C.W. Autonomous mobile robot navigation in uneven and unstructured indoor environments. In Proceedings of the 2017 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), Vancouver, BC, Canada, 24–28 September 2017; pp. 109–116.
8. Qin, T.; Li, P.; Shen, S. Vins-mono: A robust and versatile monocular visual-inertial state estimator. *IEEE Trans. Robot.* **2018**, *34*, 1004–1020.
9. Hogg, R.W.; Rankin, A.L.; Roumeliotis, S.I.; McHenry, M.C.; Helmick, D.M.; Bergh, C.F.; Matthies, L. Algorithms and sensors for small robot path following. In Proceedings of the 2002 IEEE International Conference on Robotics and Automation (Cat. No. 02CH37292), Washington, DC, USA, 11–15 May 2002; Volume 4, pp. 3850–3857.
10. Mastalli, C.; Focchi, M.; Havoutis, I.; Radulescu, A.; Calinon, S.; Buchli, J.; Caldwell, D.G.; Semini, C. Trajectory and foothold optimization using low-dimensional models for rough terrain locomotion. In Proceedings of the 2017 IEEE International Conference on Robotics and Automation (ICRA), Singapore, 29 May–3 June 2017; pp. 1096–1103.
11. Wang, K.; Ding, W.; Shen, S. Quadtree-accelerated real-time monocular dense mapping. In Proceedings of the 2018 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), Madrid, Spain, 1–5 October 2018; pp. 1–9.
12. Ding, L.; Gao, H.; Deng, Z.; Song, J.; Liu, Y.; Liu, G.; Iagnemma, K. Foot–terrain interaction mechanics for legged robots: Modeling and experimental validation. *Int. J. Robot. Res.* **2013**, *32*, 1585–1606. [CrossRef]
13. Tsaprounis, C.; Aspragathos, N.A. A linear differential formulation of friction forces for adaptive estimator algorithms. *Robotica* **2001**, *19*, 407–421. [CrossRef]
14. Fnadi, M.; Plumet, F.; Benamar, F. Nonlinear Tire Cornering Stiffness Observer for a Double Steering Off-Road Mobile Robot. In Proceedings of the 2019 International Conference on Robotics and Automation (ICRA), Montreal, QC, Canada, 20–24 May 2019; pp. 7529–7534.
15. Fleming, R.W.; Wiebel, C.; Gegenfurtner, K. Perceptual qualities and material classes. *J. Vis.* **2013**, *13*, 9. [CrossRef] [PubMed]
16. Tiest, W. Visual and haptic perception of roughness. *Perception* **2005**, *34*, 45–46.
17. Hiramatsu, C.; Goda, N.; Komatsu, H. Transformation from image-based to perceptual representation of materials along the human ventral visual pathway. *Neuroimage* **2011**, *57*, 482–494. [CrossRef] [PubMed]
18. Tang, J.; Ren, Y.; Liu, S. Real-time robot localization, vision, and speech recognition on Nvidia Jetson TX1. *arXiv* **2017**, arXiv:1705.10945.
19. Lee, K.; Lee, K.; Min, K.; Zhang, Y.; Shin, J.; Lee, H. Hierarchical novelty detection for visual object recognition. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Salt Lake City, UT, USA, 18–22 June 2018; pp. 1034–1042.

20. Ummenhofer, B.; Zhou, H.; Uhrig, J.; Mayer, N.; Ilg, E.; Dosovitskiy, A.; Brox, T. Demon: Depth and motion network for learning monocular stereo. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Honolulu, HI, USA, 21–26 July 2017; pp. 5038–5047.
21. Rosinol, A.; Sattler, T.; Pollefeys, M.; Carlone, L. Incremental Visual-Inertial 3D Mesh Generation with Structural Regularities. In Proceedings of the 2019 International Conference on Robotics and Automation (ICRA), Montreal, QC, Canada, 20–24 May 2019; pp. 8220–8226.
22. Cordts, M.; Omran, M.; Ramos, S.; Rehfeld, T.; Enzweiler, M.; Benenson, R.; Franke, U.; Roth, S.; Schiele, B. The cityscapes dataset for semantic urban scene understanding. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Las Vegas, NV, USA, 26 June–1 July 2016; pp. 3213–3223.
23. Lin, T.Y.; Maire, M.; Belongie, S.; Hays, J.; Perona, P.; Ramanan, D.; Dollár, P.; Zitnick, C.L. Microsoft coco: Common objects in context. In Proceedings of the European Conference on Computer Vision, Zurich, Switzerland, 6–12 September 2014; pp. 740–755.
24. Armeni, I.; Sax, S.; Zamir, A.R.; Savarese, S. Joint 2d-3d-semantic data for indoor scene understanding. *arXiv* **2017**, arXiv:1702.01105.
25. Pont-Tuset, J.; Perazzi, F.; Caelles, S.; Arbeláez, P.; Sorkine-Hornung, A.; Van Gool, L. The 2017 davis challenge on video object segmentation. *arXiv* **2017**, arXiv:1704.00675.
26. Caesar, H.; Uijlings, J.; Ferrari, V. Coco-stuff: Thing and stuff classes in context. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Salt Lake City, UT, USA, 18–22 June 2018; pp. 1209–1218.
27. Schilling, F.; Chen, X.; Folkesson, J.; Jensfelt, P. Geometric and visual terrain classification for autonomous mobile navigation. In Proceedings of the 2017 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), Vancouver, BC, Canada, 24–28 September 2017; pp. 2678–2684.
28. Garcia-Garcia, A.; Orts-Escolano, S.; Oprea, S.; Villena-Martinez, V.; Garcia-Rodriguez, J. A review on deep learning techniques applied to semantic segmentation. *arXiv* **2017**, arXiv:1704.06857.
29. Long, J.; Shelhamer, E.; Darrell, T. Fully convolutional networks for semantic segmentation. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Boston, MA, USA, 7–12 June 2015; pp. 3431–3440.
30. Badrinarayanan, V.; Kendall, A.; Cipolla, R. Segnet: A deep convolutional encoder-decoder architecture for image segmentation. *IEEE Trans. Pattern Anal. Mach. Intell.* **2017**, *39*, 2481–2495. [CrossRef] [PubMed]
31. Chen, L.C.; Papandreou, G.; Schroff, F.; Adam, H. Rethinking atrous convolution for semantic image segmentation. *arXiv* **2017**, arXiv:1706.05587.
32. He, K.; Zhang, X.; Ren, S.; Sun, J. Deep residual learning for image recognition. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Las Vegas, NV, USA, 27–30 June 2016; pp. 770–778.
33. Adelson, E.H. On seeing stuff: The perception of materials by humans and machines. In Proceedings of the Human Vision and Electronic Imaging VI. International Society for Optics and Photonics, San Jose, CA, USA, 8 June 2001; Volume 4299, pp. 1–12.
34. Komatsu, H.; Goda, N. Neural mechanisms of material perception: Quest on Shitsukan. *Neuroscience* **2018**, *392*, 329–347. [CrossRef] [PubMed]
35. Bajracharya, M.; Ma, J.; Malchano, M.; Perkins, A.; Rizzi, A.A.; Matthies, L. High fidelity day/night stereo mapping with vegetation and negative obstacle detection for vision-in-the-loop walking. In Proceedings of the 2013 IEEE/RSJ International Conference on Intelligent Robots and Systems, Tokyo, Japan, 3–7 November 2013; pp. 3663–3670.
36. Wooden, D.; Malchano, M.; Blankespoor, K.; Howardy, A.; Rizzi, A.A.; Raibert, M. Autonomous navigation for BigDog. In Proceedings of the 2010 IEEE International Conference on Robotics and Automation, Anchorage, AK, USA, 3–7 May 2010; pp. 4736–4741.
37. Deng, F.; Zhu, X.; He, C. Vision-based real-time traversable region detection for mobile robot in the outdoors. *Sensors* **2017**, *17*, 2101. [CrossRef] [PubMed]
38. Shinzato, P.Y.; Fernandes, L.C.; Osorio, F.S.; Wolf, D.F. Path recognition for outdoor navigation using artificial neural networks: Case study. In Proceedings of the 2010 IEEE International Conference on Industrial Technology, Vina del Mar, Chile, 14–17 March 2010; pp. 1457–1462.
39. Russell, B.C.; Torralba, A.; Murphy, K.P.; Freeman, W.T. LabelMe: A database and web-based tool for image annotation. *Int. J. Comput. Vis.* **2008**, *77*, 157–173.

40. Ng, A. Machine Learning Yearning. 2017. Available online: http://www.mlyearning.org/(96) (accessed on 9 June 2018).
41. Kingma, D.P.; Ba, J. Adam: A method for stochastic optimization. *arXiv* **2014**, arXiv:1412.6980.
42. Kanopoulos, N.; Vasanthavada, N.; Baker, R.L. Design of an image edge detection filter using the Sobel operator. *IEEE J. Solid-State Circuits* **1988**, *23*, 358–367. [CrossRef]
43. Silva, M.F.; Machado, J.T.; Lopes, A.M. Modelling and simulation of artificial locomotion systems. *Robotica* **2005**, *23*, 595–606. [CrossRef]
44. Breiman, L.; Friedman, J.; Stone, C.J.; Olshen, R.A. *Classification and Regression Trees*; CRC Press: Boca Raton, FL, USA, 1984.