Safe Robot Navigation via Multi-Modal Anomaly Detection

Originally published in:
IEEE Robotics and Automation Letters 5(2), https://doi.org/10.1109/lra.2020.2967706

Funding acknowledgement:
780883 - subTerranean Haptic INvestiGator (EC)

Safe Robot Navigation via Multi-Modal Anomaly Detection

Lorenz Wellhausen[1], René Ranftl[2] and Marco Hutter[1]

***Abstract*—Navigation in natural outdoor environments requires a robust and reliable traversability classification method to handle the plethora of situations a robot can encounter. Binary classification algorithms perform well in their native domain but tend to provide overconfident predictions when presented with out-of-distribution samples, which can lead to catastrophic failure when navigating unknown environments. We propose to overcome this issue by using anomaly detection on multi-modal images for traversability classification, which is easily scalable by training in a self-supervised fashion from robot experience. In this work, we evaluate multiple anomaly detection methods with a combination of uni- and multi-modal images in their performance on data from different environmental conditions. Our results show that an approach using a feature extractor and normalizing flow with an input of RGB, depth and surface normals performs best. It achieves over 95% area under the ROC curve and is robust to out-of-distribution samples.**

***Index Terms*—Visual-Based Navigation; Visual Learning; RGB-D Perception; AI-Based Methods**

I. Introduction

ROBOT navigation through natural outdoor environments introduces challenges which are usually not considered when deploying autonomous systems in indoor and man-made environments. The most notable difference is that perceived geometry can not be assumed to be rigid. The implications for this are two-fold: First, while flat terrain is typically assumed to be traversable, it can actually be untraversable or dangerous for robot navigation if the terrain is non-rigid. Treacherous terrain like deep sand, mud and bodies of water show flat geometry but are potentially fatal for many robots. Second, while obstacles are often simply considered as the presence of geometry, this does not hold when a robot can ”push through” a compliant obstruction. Vegetation like grass and small bushes are difficult to identify from purely geometric information but are frequently encountered in natural environments.

This implies that semantic environment information is desirable, if not necessary, in addition to geometric information to navigate such environments. While analytical models have been successfully used to infer traversability from geometric information [1], deriving an analytical model for semantic information from image data is infeasible, due to the high dimensionality of the problem.

Fig. 1: Anomaly detection allows robots to operate in environments with unforeseen and rarely occurring obstacles.

Machine-learning models achieve state-of-the-art performance in semantic image processing, using manually labelled data [2], [3]. However, manually labelling data is cumbersome and not scalable to larger quantities of data. Additionally, it relies on a human expert who often lacks a good intuition about traversability for environments where the robot has not been operated before and cannot provide quantitative terrain information.

When collecting self-supervised samples through robot experience, which we have shown in previous work [4], positive samples for traversable terrain can be gathered safely and in large quantities. Collecting negative samples, however, implies provoking robot failure which can be harmful for the robot. In addition, labelling negative samples can never cover the entire domain of untraversable terrains and possible obstructions, which can lead to over-confident classifier output when presented with out-of-training-distribution samples.

Detection of out-of-distribution samples, also called anomaly detection, or novelty detection, has received increased attention with the recent success of deep learning in general, and semantic image processing specifically. These methods can be trained using positive samples only, and are by design robust to out-of-distribution samples. However, existing work does not fully commit to the concept [5], [6], assumes constant appearance of the environment [7], and doesn't leverage geometric information. In this work, we present an approach to fully leverage anomaly detection using appearance and geometric information for safe robot navigation in various environments.

The main contribution of this work is an extensive eval-

Manuscript received: September, 11th, 2019; Revised December, 6th, 2019; Accepted January, 2nd, 2020.

This paper was recommended for publication by Editor Eric Marchand upon evaluation of the Associate Editor and Reviewers' comments.

This work was supported by the Intel Network on Intelligent Systems, the Swiss National Science Foundation (SNF) through the NCCR Robotics, and the EU Horizon 2020 research and innovation programme (No 780883). It has been conducted as part of ANYmal Research, a community to advance legged robotics

[1]Robotic Systems Lab, ETH Zürich, Switzerland

[2]Intel Labs, Munich, Germany

Digital Object Identifier (DOI): see top of this page.

uation of multiple anomaly detection methods and sensor modality combinations with respect to their performance on self-supervised data. We release the dataset used in this work to enable reproduction of our results and to develop the concept of anomaly-based navigation further. Lastly, we combine ideas from other works [8], [9] into a new anomaly detection approach, which trains a feature embedding directly by maximizing the log-likelihood.

We show that we are able to train an anomaly detection method using only positive examples of multi-modal data to be highly discriminative. Our best model reaches more than 95% area under the receiver-operator characteristic curve (AUROC), which enables safe robot navigation.

II. Related Work

Traditional navigation approaches for mobile robots use a geometric environment representation as their only basis for traversability estimation [1], [10], [11], [12]. This line of work is well developed and provides good performance in man-made environments, but fails to capture compliant terrain.

Semantic-aware navigation approaches typically leverage additional sensor modalities to infer additional terrain information [13], [14], [15], [16], [17], [18], [2], [19], [20]. Approaches using more unconventional sensors either require a long observation duration [19] or a bulky sensor payload [20] which exceeds the capabilities of our target platform. Therefore, most work is focused on camera-based methods and either performs semantic segmentation of the environment [13], [15], [16], [2] or directly predicts a traversability label [14], [17], [18]. Semantic segmentation approaches [15], [2] can perform well in environments similar to their training domain but do not transfer to unknown environments [9]. While this is not prohibitive in some domains [13], [16], in most cases even small changes in the environmental conditions, for example due to weather, can drastically change the appearance of terrain classes.

In previous work [4] we have shown that we can predict terrain properties ahead of the robot without classifying the terrain. This can be used to make informed navigation decisions on terrain that is traversable, but does not provide a traversability classification itself. We therefore require an additional method to provide traversability labels, which should also be learned in a self-supervised fashion to maintain scalability of the navigation pipeline.

Some recent work has proposed weakly- and self-supervised learning for navigation purposes by combining multiple sensors [16], [17] or proprioception [14], [18]. All of these approaches use binary classification with a fixed set of predefined classes, however. This has shown to be prone to overconfident predictions in the presence of out-of-distribution data [21], which can lead to disastrous consequences [22].

Anomaly detection could solve this problem by learning the distribution of safe terrain, which makes the approach more robust to out-of-distribution samples. Numerous work is available for anomaly detection, which uses autoencoders [23], support vectors [24], [8], generative adversarial networks[25] and normalizing flow [26].

Anomaly detection has been used for indoor navigation [6], planetary exploration [5], and for navigation in agricultural fields [7]. However, these approaches are either reduced in scope [6], rely on a consistent terrain appearance [7], or use an additional binary classifier to make the final anomaly decision [5].

We propose a scalable approach for safe navigation which can be trained in a fully self-supervised fashion from only traversable examples. We learn the distribution of terrain which the robot has safely traversed before and consider out-of-distribution samples as unsafe. This enables safe robot locomotion, even in the presence of unknown obstacles.

III. Method

We aim to learn a model of the typical appearance of terrain that the robot has safely navigated before. We can use this model for safe navigation by classifying new sensory inputs into "known" and "unknown" terrain classes. We use an automated pipeline that automatically generates positive labels from sensory data. We then evaluate the performance of different novelty detection methods and input modalities in various scenarios. We further briefly outline how the resulting image-based labels can be used in a robot navigation framework.

A. Data Collection

We collect positive terrain samples from robot-experience in a self-supervised fashion. The basic pipeline was presented in our earlier work [4]. The quadrupedal robot ANYmal is teleoperated over various terrain while we record the image streams of an onboard camera, as well as the foothold contact locations in a robot-centered frame. We use Visual SLAM on the image stream to recover the camera poses and foothold locations in a common coordinate frame. This allows us to project all footholds along the robot trajectory into all camera images. We consequently obtain image locations which correspond to positive labels for traversability. In a final step, we extract the image patches at the foothold locations to generate our training dataset.

Our pipeline can be applied to any dense exteroceptive sensor. in this work we use a RGB-D camera to sense both appearance and geometry of the environment. We hereafter refer to *images* as the stack of RGB and depth images and potentially derived quantities.

B. Anomaly Detection

We evaluate multiple approaches with respect to to their anomaly detection performance on our data. Our investigation is focused on deep learning approaches with fully convolutional architectures, since they achieve state-of-the-art performance, and are efficient during test time even on larger input images.

We define a feature encoder $f(\mathbf{x}) \rightarrow \mathbf{y}$, which maps an image patch $\mathbf{x} \in \mathbb{R}^{w \times h \times c}$ of width w and height h and a channel depth of c to a feature vector $\mathbf{y} \in \mathbb{R}^d$. The encoder architecture will be shared by all novelty detection approaches. We implement $f(\mathbf{x})$ with a fully-convolutional neural network

in order to support inference on arbitrary image sizes ($W \times H$). We will obtain an output tensor $\mathbf{y}_{\text{inf}} \in \mathbb{R}^{j \times k \times d}$ with $j \approx \frac{W}{4}$ and $k \approx \frac{H}{4}$, which we use for localized anomaly detection in the full-size image.

The network architecture for the encoder uses three consecutive blocks, each consisting of a convolutional layer with kernel size 5 with a leaky ReLU non-linearity. The first two blocks are followed by a MaxPool layer of size 2, while the last block is followed by a final convolution with kernel size 1. The number of channels is, in sequence, $[c, 32, 64, 128, 128]$, where c is the number of input channels.

We further denote the training loss as $\mathcal{L}$ and the anomaly decision criterion as $\mathcal{C}$. We use a simple threshold on the decision criterion to classify patches into their respective classes.

1) Autoencoder [27]: Autoencoders are neural networks that consist of an encoder $f(\mathbf{x})$ to generate a (low-dimensional) latent feature vector $\mathbf{y}$ from the image patch and a decoder $f'(\mathbf{y})$, which tries to reconstruct the input patch from this latent vector. Since the feature vector is low-dimensional when compared to the dimensionality of the input patch, an internal information bottleneck is introduced. The autoencoder is thus forced to learn descriptive image features in order to be able to reconstruct the input. In the context of anomaly detection, the basic assumption is that the autoencoder will over-fit to the training distribution. Anomalous input images will therefore be reconstructed with less accuracy than images that are similar to the training images.

Our implementation uses a decoder network $f'(\mathbf{y})$ that is composed of convolution layers of the same dimensions as the encoder, but with nearest-neighbor upscaling layers instead of MaxPooling. The training loss for the autoencoder is given by

$$\mathcal{L}_{\text{AE}}(\mathbf{y}) = \mathcal{C}_{\text{AE}}(\mathbf{y}) = \frac{1}{n} \sum_{n} (f'(\mathbf{y}) - \mathbf{x})^2. \tag{1}$$

Note, that we use the reconstruction error in image space as the decision criterion.

2) Deep SVDD: Ruff et al. [8] propose an anomaly detection approach based on deep networks. In this approach, a neural network is trained to extract image features that are contained in a hypersphere, where the hypersphere is jointly adapted during training of the feature extractor. At test time, samples which fall outside of the hypersphere are assumed to be anomal samples. Ruff et al. [8] propose two different variants of this general idea. A soft-boundary formulation with the training loss

$$\mathcal{L}_{\text{Soft}}(\mathbf{y}) = R^2 + \frac{1}{\nu} \max\{0, \|\mathbf{y} - \mathbf{c}\|_2^2 - R^2\}, \tag{2}$$

and a hard-boundary formulation with loss

$$\mathcal{L}_{\text{Hard}}(\mathbf{y}) = \|\mathbf{y} - \mathbf{c}\|_2^2. \tag{3}$$

The center of the hypersphere $\mathbf{c} \in \mathbb{R}^d$ is an arbitrary, fixed, non-zero vector that needs to be chosen in advance. We follow the recommendations of the original authors and initialize it with an initial forward-pass on the untrained network [8]. Note, that the decision radius R in the soft-boundary formulation is optimized together with the parameters of the feature generator.

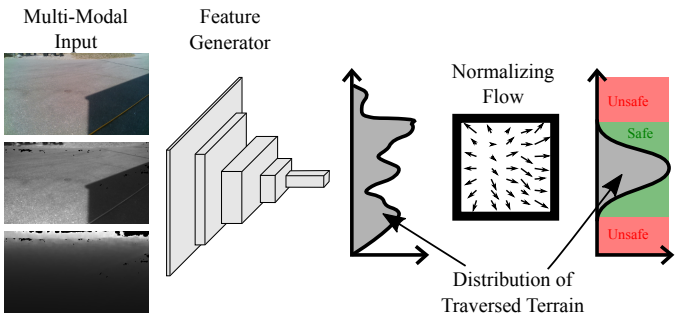


Fig. 2: Multi-modal input images are projected into a feature space to form a distribution of safe terrain features. Normalizing flow is used to transform this distribution and facilitate exact likelihood computation.

We use the squared distance to the center of the hypersphere as decision criterion for both formulations:

$$\mathcal{C}_{\text{SVDD}} = \|\mathbf{y} - \mathbf{c}\|_2^2. \tag{4}$$

3) Embedding + Real-NVP: Normalizing flow models [28], [29], [30] are powerful methods which can be used to learn arbitrary probability distributions by maximizing the likelihood of training samples. The normalizing flow approach is inherently probabilistic. As a consequence, it naturally handles variations and noise in the input data. Furthermore, since the likelihood is a metric for how likely it is that a given distribution has generated a given feature, it is a very natural decision criterion for anomaly detection.

Since normalizing flow methods limit themselves to be composed of specific, invertible network modules, they allow for tractable computation of the log-determinant of their Jacobian. However, this restriction combined with a high-dimensional feature space when operating on images makes deep architectures and large amounts of training data necessary. This makes them impractical to use for our application.

Similar to a method proposed by Blum et al. [9], we combine normalizing flows with a convolutional embedding network, which generates a lower-dimensional feature vector from an image patch. We then learn the safe terrain distribution in the low-dimensional latent space. A schematic overview of the approach is depicted in Figure 2.

We use Real-NVP [29] as our normalizing flow method. Let $g(\mathbf{y}) \rightarrow \mathbf{z}$ be a bijection, which transforms the latent variable $\mathbf{y}$ into another vector $\mathbf{z} \in \mathbb{R}^d$ of same dimensionality. We assume a given prior distribution $p_Z(\mathbf{z})$ on the transformed vector $\mathbf{z}$. The prior distribution can be chosen arbitrarily, as long as its log-likelihood can easily be computed. Using the change-of-variable formula we can obtain the log-likelihood of the posterior distribution in latent space, which serves as our loss function:

$$\mathcal{L}_{\text{NVP}}(\mathbf{y}) = -\log(p_Z(g(\mathbf{y}))) - \log(|\det\left(\frac{\delta g(\mathbf{y})}{\delta \mathbf{y}^T}\right)|). \tag{5}$$

Real-NVP specifically limits the modules to scaling and translation of the intermediate features. The log-determinant can consequently be computed as the sum of the scaling factors [29]. We further directly use the log-likelyhood as the decision criterion:

$$\mathcal{C}_{\text{NVP}}(\mathbf{y}) = \mathcal{L}_{\text{NVP}}(\mathbf{y}). \tag{6}$$

Our Real-NVP flow network has 6 affine coupling layers where scaling and translation coefficients are obtained from a MLP with two hidden layers.

C. Input Modalities

Intel RealSense cameras provide RGB, infrared and depth streams. We ignore the infrared stream in this work, due to inconsistencies between infrared imagers of different camera models. However, in addition to RGB and raw depth, we also consider modalities that are derived from depth and the robot state information:

1) Gravity Aligned Depth: We project the depth image into 3D space using the camera intrinsics $\mathbf{K}$ and then rotate these points with the orientation of the camera in the gravity-aligned odometry frame $\mathbf{R}_{oc} \in SO(3)$ which is provided by the inertial-kinematic state estimator. We then combine the two horizontal axes into the distance in the horizontal plane. Let d be the depth value at image coordinates $[u, v]$.

$$\begin{aligned} \mathbf{p} &= \mathbf{R}_{oc} \cdot \mathbf{K}^{-1} \cdot \begin{bmatrix} u \\ v \\ 1 \end{bmatrix} \cdot d, \\ \mathbf{d}_g &= \begin{bmatrix} d_{\text{horz}} \\ d_{\text{vert}} \end{bmatrix} = \begin{bmatrix} \sqrt{p_x^2 + p_y^2} \\ p_z \end{bmatrix}. \end{aligned} \tag{7}$$

2) Gravity Aligned Surface Normals: We compute gravity-aligned surface normals $\mathbf{n}_g$ from $\mathbf{p}$ using the FALS algorithm [31] and combine horizontal components in the same way we did for $\mathbf{d}_g$:

$$\mathbf{n}_g = \begin{bmatrix} n_{\text{horz}} \\ n_{\text{vert}} \end{bmatrix} = \begin{bmatrix} \sqrt{n_x^2 + n_y^2} \\ n_z \end{bmatrix}. \tag{8}$$

3) Surface Normal Angle: We compute the angle between surface normal and the horizontal plane n_{ang} as

$$n_{\text{ang}} = \arctan\left(\frac{n_{\text{vert}}}{n_{\text{horz}}}\right). \tag{9}$$

D. Navigation

All anomaly detection methods are trained on image patches and are fully convolutional. This means we can deploy them on larger images than they were trained on to obtain an anomaly mask for the input image. We use this mask as a measure of traversability. We then use the depth channel of our input image to project the anomaly mask to 3D space, which gives us point estimates for traversability in 3D. Finally, these measurements can be used in a mapping framework to obtain a environment representation that can be used for planning. In our case, we opted for a 2D grid representation, which is common for ground robots and can be used for efficient path planning.

IV. Experimental Results

Experiments were performed on data collected with the ANYmal [32] quadruped, with image data captured using Intel Realsense cameras. Data for the base training set was captured on a Realsense ZR300, while test data was recorded on a Realsense D435. ANYmal was teleoperated over various terrain, with the forward-facing cameras at a slight downward angle, which varied between sorties.

We provide code and dataset online to reproduce our results and to encourage further research on anomaly navigation.[1]

A. Dataset

We use the data collected for our previous work [4] as our base training set, which represents about 2.5 hours of continuous robot operation under sunny and overcast lighting conditions. It was collected by teleoperating the robot through an urban park, a forest, and farmland, and covers various terrain types like asphalt, grass, dirt and sand.

We also recorded new data in a search-and-rescue training facility to evaluate this work. We chose this particular training site, because we can artificially create anomalous obstacles and events in a safe and controlled fashion. Note that this method is not specific to search-and-rescue scenarios and can be used for general-purpose navigation. In this new location, the robot followed the same general path multiple times under different environmental conditions.

1) *Sun*: Direct sunlight in the afternoon.
2) *Fire*: Direct sunlight in the afternoon, but with a controlled fire in the robot field-of-view.
3) *Rain*: During rain, with varying intensity from light to moderately heavy rain.
4) *Wet*: In the late afternoon under direct sunlight, with wet ground from preceding rain.
5) *Twilight*: Just after sunset during twilight.

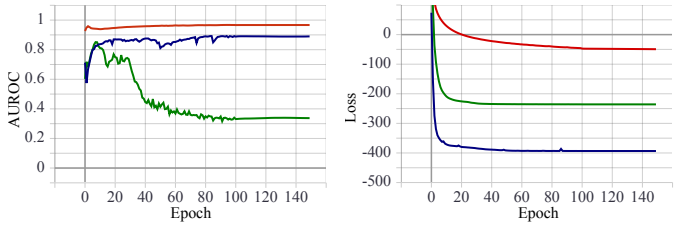
For *Sun* and *Twilight* we each performed two sorties following the same path, while we could only perform a shorter second sortie for *Rain* and no second sortie for *Wet* and *Fire*.

As network input at training time, we choose image patches of size 32×32. Patches of traversable terrain are extracted in a self-supervised fashion, as described in Section III-A. While image patches of traversable terrain are sufficient to train our approaches, patches of untraversable terrain are necessary for a quantitative performance analysis. Because we do not have self-supervised data of untraversable terrain we manually label 500 negative samples in each sortie for our evaluation. Note that we do not need to manually label any training data, as all approaches are trained using positive samples only. Through this approach we obtained 10 000 training image patches and around 500 positive and negative samples for each test sortie.

B. Network Training

Both Deep-SVDD (*SVDD Soft* + *SVDD Hard*) and Real-NVP (*NVP*) methods are evaluated with randomly initialized weights (*No Pretraining*), as well as with the feature generator

[1] http://github.com/leggedrobotics/anomaly_navigation



(a) Test Set AUROC (b) Training Loss

Fig. 3: Training curves of NVP methods show that high likelihood of the training distribution does not correlate to high discriminative performance on the test set. *Blue* - No Pretraining, *Green* - Pretrained, *Red* - Pretrained + fixed feature generator

pretrained using the autoencoder (*Pretrained*). For the Real-NVP architecture we also tried fixing the feature generator weights after pretraining (*Fixed Features*). We pretrain for 350 epochs for relevant methods and then train the full method for 150 epoch. We use Adam [33] with a learning rate of 1e-4. The hyperparameters were chosen to be the same as in the original Deep SVDD paper [8] for all experiments.

C. Numerical Evaluation

For quantitative analysis of anomaly detection approaches and sensor modality combinations, we train our approaches on the base training set and use data from one *Sun* sortie as test set. We use the threshold-independent AUROC as performance metric. Table I shows results for all evaluated approaches and sensor modality combinations.

1) Anomaly Detection Methods: We can see that the Real-NVP based approaches clearly outperform the autoencoder and Deep SVDD approaches. This approach tries to explicitly learn the posterior distribution of traversable image features and allows us to learn arbitrary distributions, whereas Deep SVDD assumes a uni-modal distribution, since it classifies all features inside a hypersphere as inliers. Additionally, the objective function does not force the network to learn the joint distribution over all input modalities. It can in principle converge to a solution which ignores some input modalities if others allow easier mapping to a fixed feature point. The autoencoder approach is able to learn a good approximation of the underlying distribution, as evidenced by generally higher performance than Deep SVDD which rivals Real-NVP, when provided with surface normals. Its otherwise inferior performance to Real-NVP stems from the appearance-based decision criterion, which is a poor similarity measure. The higher performance of the Real-NVP version trained with fixed feature generator weights we assume to be caused by joint distribution learning of multiple modalities. Some parts of the actual underlying distribution are ignored without fixed features, in favor of mapping to a simpler posterior distribution, where higher likelihood can be achieved. An indicator is a lower training loss while also having a lower discriminative performance pictured in Figure 3.

2) Sensor Modalities: Unsurprisingly, geometric modalities enable consistently high performance, given that it is the preferred modality for traversability classification in literature [1], [11]. Providing explicit surface normal information (*N*) provides significant gains over depth (*D*) and gravity-aligned depth (*G*) hinting that the convolutional layers of the feature generator do not learn to fully leverage the presented geometric information. Interestingly, the surface normal angle (*A*), which directly corresponds to terrain inclination, commonly used for traversability estimation in analytical approaches [1], does not provide the same performance boost as the normal vectors.

Using *RGB*-only shows significantly worse performance than any combination with geometric information. This is not surprising, given that geometry is a major deciding factor in whether or not terrain is traversable. Inferring geometric information from color images is a hard problem even when networks are explicitly trained for this task which makes it unlikely that our network learns to reason about it. Hence, the network cannot distinguish between concrete walls and asphalt streets, which have very similar texture and are common in our dataset.

However, in many cases geometry alone is not enough to infer traversability. For example, tall grass leads to a geometry that suggests untraversable, but can easily be recognized as traversable from the RGB image. Adding *RGB* information to any geometric modality combination improves performance, because it helps to distinguish rigid from non-rigid geometry and gives additional information in image regions with missing depth due to stereo matching failure.

Qualitative results of the highest performing method, *Real-NVP Fixed Features* with *RGB+G+N* are shown in Figure 4.

D. Incremental Learning

In this section we will demonstrate how adding more input data from new environments allows our method to scale and improve performance over time. We use the modality combination without any post-processing of the depth information, *RGB+D*, and Real-NVP on the latent space of a feature generator with fixed weights that were pretrained with an autoencoder. We will train our network with increasingly more data and see how the performance evolves for different environmental conditions. We reduce the training data set size from 10 000 to 500 image patches, which will serve as base training set, while we incrementally add 500 image patches of one sortie in a given condition to the training data. The second sortie under that condition will serve as the test set. We use the shorter sortie under *Rain* conditions as train set and compensate for the shorter duration by sampling multiple patches per image to reach 500 samples. Additionally, *Wet* and *Fire* conditions will remain purely test sets as we only have data of a single sortie. Because of the reduced dataset size we only train for 10 epochs.

The results in Figure 5 show an increase in performance for all conditions when adding more data. With additional data, the AUROC for *Sun* conditions improves over the results with the full training set shown in Table I, which was tested with the same data. The highest gains are achieved for *Rain* conditions, which shows that diverse training data is crucial to handle operation in various environmental conditions. Strong performance on *Fire*, where unsafe terrain is dominated by

TABLE I: Different anomaly detection methods and sensor modality combinations evalauted using the AUROC. We test in *Sun* conditions and indicate the standard deviation over 10 runs. The background color follows a gradient corresponding to the AUROC 0 50 100. Addtionally, the highest performing combination is highlighted in bold text. For a description of anomaly detection methods refer to Section III-B. Modality short-hands are: *RGB* - color, *D* - Depth, *G* - gravity aligned depth, *N* - gravity aligned surface normals, *A* - surface normal angle.

	Autoencoder	**SVDD Soft No Pretraining**	**SVDD Hard No Pretraining**	**SVDD Soft Pretrained**	**SVDD Hard Pretrained**	**NVP No Pretraining**	**NVP Pretrained**	**NVP Fixed Features**
RGB	38.97±0.39	18.55±2.85	58.46±12.89	60.09±6.14	70.57±3.95	73.52±0.85	76.16±2.95	64.34±0.68
D	76.07±0.30	80.04±1.88	79.72±2.38	48.89±6.42	78.67±2.02	77.69±2.46	31.14±0.93	81.41±0.05
RGB+D	63.92±0.73	47.87±7.03	50.40±8.41	51.96±17.67	71.79±4.09	72.35±3.02	74.84±3.48	84.64±0.21
RGB+G	68.44±0.29	59.58±5.39	70.77±13.20	41.57±27.75	74.64±4.56	85.69±1.43	83.12±3.21	87.12±0.85
RGB+N	92.92±0.20	60.63±30.35	43.39±6.22	40.00±3.41	53.44±9.55	86.45±0.86	45.04±13.28	93.12±1.00
RGB+A	67.79±0.08	20.27±7.26	63.17±6.89	36.09±33.44	69.44±4.81	90.06±1.10	68.43±12.04	87.69±0.14
D+N	92.81±0.09	75.40±24.53	57.76±7.20	89.30±1.50	62.27±10.90	83.93±1.18	49.72±10.93	90.08±0.97
D+A	79.91±0.11	80.63±2.55	61.91±0.56	49.43±1.47	70.10±3.59	77.44±0.71	55.69±2.78	84.12±0.57
G+A	80.72±0.17	56.94±21.08	71.07±7.44	52.79±20.09	78.53±7.80	87.47±0.42	80.14±1.45	86.29±0.84
RGB+D+N	94.17±0.27	44.28±21.54	51.99±2.13	52.48±5.61	54.28±5.62	85.50±2.28	46.29±14.02	94.99±0.41
RGB+D+A	76.19±0.44	44.13±12.67	55.39±14.01	41.16±20.11	69.45±5.57	89.45±1.23	82.16±3.77	90.11±0.71
RGB+G+N	94.51±0.04	38.95±35.13	61.76±4.43	38.35±10.48	62.28±3.63	87.53±2.10	50.42±11.81	**95.14±1.47**
RGB+G+A	78.77±0.54	58.44±10.18	73.67±9.14	33.76±28.66	75.25±5.44	92.85±0.26	80.06±2.36	91.60±0.16

bright fire and billows of black smoke, show that our anomaly detection based approach can safely handle unknown environmental hazards. An important additional note is, that the true-positive rate at 5% false-positive rate (TPR @ 5% FPR) improves drastically for all but *Fire* conditions, when adding more data. It also improves over the full training data TPR @ 5% FPR of 43% for *Sun* conditions. This measure is a good indicator for a navigation task operating point, since we want a low false-positive rate to minimize the chance of catastrophic robot failure. Interestingly, *Rain* and *Twilight* data seem to be much more significant for improving TPR @ 5% FPR than *Sun* data, even when evaluating under *Sun* conditions.

E. Network Inference Time

All networks run in real-time on mobile computation hardware. Table II reports inference times once per base approach, since the different training methods do not alter the inference time.

TABLE II: Inference times for the three base approaches on an Nvidia Jetson Xavier (15W mode) with input image size 848×480.

RGB+G+N	**Autoencoder**	**SVDD**	**Real-NVP**
time [ms]	9.4	4.6	42.9
rate [Hz]	106.1	216.6	23.3

V. Conclusion

In this work we demonstrated a method for safe robot navigation in the presence of unknown obstacles using anomaly detection. Our approach combining a feature embedding with normalizing flow is able to operate in a variety of environments and scales well with additional data. Our semi-supervised data collection pipeline enables to collect multi-modal data from experience without any manual labelling. The highest performance was achieved with a sensor modality combination of RGB images, depth and surface normals. Our work opens up several avenues for future research. An active exploration approach could automate the collection of new data and ease the expansion of the robot's operating range. While the current approach trains only on samples of safe terrain, extending it to use sparse experiences of robot failure could sharpen decision boundaries in ambiguous environments. Additionally, increasing the receptive field size of our network could allow the robot to reason about even more complex environments where translucent and reflective objects are present.

References

[1] M. Wermelinger, P. Fankhauser, R. Diethelm, P. Krüsi, R. Siegwart, and M. Hutter, "Navigation planning for legged robots in challenging terrain," in *IROS*. IEEE, 2016, pp. 1184–1189.

[2] A. Valada, J. Vertens, A. Dhall, and W. Burgard, "Adapnet: Adaptive semantic segmentation in adverse environmental conditions," in *ICRA*. IEEE, 2017, pp. 4644–4651.

[3] E. Romera, J. M. Alvarez, L. M. Bergasa, and R. Arroyo, "ERFNet: Efficient residual factorized convnet for real-time semantic segmentation," *Trans. on Intelligent Transportation Systems*, vol. 19, no. 1, pp. 263–272, 2018.

[4] L. Wellhausen, A. Dosovitskiy, R. Ranftl, K. Walas, C. Cadena, and M. Hutter, "Where should i walk? predicting terrain properties from images via self-supervised learning," *IEEE Robotics and Automation Letters*, vol. 4, no. 2, pp. 1509–1516, 2019.

[5] H. R. Kerner, D. F. Wellington, K. L. Wagstaff, J. F. Bell, and H. B. Amor, "Novelty detection for multispectral images with application to planetary exploration," 2019.

[6] C. Richter and N. Roy, "Safe visual navigation via deep learning and novelty detection," 2017.

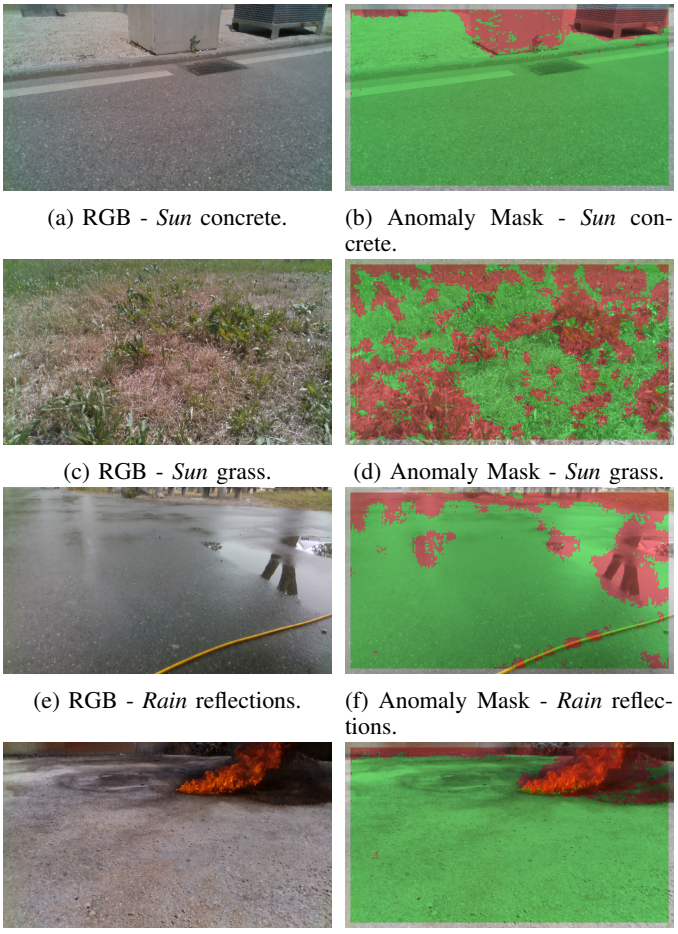
[7] P. Christiansen, L. Nielsen, K. Steen, R. Jørgensen, and H. Karstoft, "Deepanomaly: Combining background subtraction and deep learning for detecting obstacles and anomalies in an agricultural field," *Sensors*, vol. 16, no. 11, p. 1904, 2016.

[8] L. Ruff, N. Görnitz, L. Deecke, S. A. Siddiqui, R. Vandermeulen, A. Binder, E. Müller, and M. Kloft, "Deep one-class classification," in *International Conference on Machine Learning*, 2018, pp. 4390–4399.

[9] H. Blum, P.-E. Sarlin, J. Nieto, R. Siegwart, and C. Cadena, "The fishyscapes benchmark: Measuring blind spots in semantic segmentation," *arXiv preprint arXiv:1904.03215*, 2019.

[10] R. O. Chavez-Garcia, J. Guzzi, L. M. Gambardella, and A. Giusti, "Image classification for ground traversability estimation in robotics," in *Int. Conf. on Advanced Concepts for Intelligent Vision Systems*. Springer, 2017, pp. 325–336.

[11] P. Krüsi, P. Furgale, M. Bosse, and R. Siegwart, "Driving on point clouds: Motion planning, trajectory optimization, and terrain assessment in generic nonplanar environments," *Journal of Field Robotics*, vol. 34, no. 5, pp. 940–984, 2017.



(a) RGB - *Sun* concrete. (b) Anomaly Mask - *Sun* concrete.

(c) RGB - *Sun* grass. (d) Anomaly Mask - *Sun* grass.

(e) RGB - *Rain* reflections. (f) Anomaly Mask - *Rain* reflections.

(g) RGB - *Fire*. (h) Anomaly Mask - *Fire*.

Fig. 4: Qualitative results of anomaly detection in different environments. (a)+(b) Depth information helps to classify an ambiguous texture as obstacle. (c)+(d) While short grass is present in the training set, taller vegetation is not and gets classified as outlier. (e)+(f) Specular reflections which are not in the training set are classified as outliers. (g)+(f) Fire as an unknown obstacle is clearly identified.

[12] H. Oleynikova, Z. Taylor, M. Fehr, R. Siegwart, and J. Nieto, "Voxblox: Incremental 3d euclidean signed distance fields for on-board mav planning," in *IROS*. IEEE, 2017, pp. 1366–1373.

[13] B. Rothrock, R. Kennedy, C. Cunningham, J. Papon, M. Heverly, and M. Ono, "Spoc: Deep learning-based terrain classification for mars rover missions," in *AIAA SPACE 2016*, 2016, p. 5539.

[14] D. Kim, J. Sun, S. M. Oh, J. M. Rehg, and A. F. Bobick, "Traversability classification using unsupervised on-line visual learning for outdoor robot navigation," in *ICRA*. IEEE, 2006, pp. 518–525.

[15] D. M. Bradley, J. K. Chang, D. Silver, M. Powers, H. Herman, P. Rander, and A. Stentz, "Scene understanding for a high-mobility walking robot," in *IROS*. IEEE, 2015, pp. 1144–1151.

[16] K. Otsu, M. Ono, T. J. Fuchs, I. Baldwin, and T. Kubota, "Autonomous terrain classification with co-and self-training approach," *IEEE Robotics and Automation Letters*, vol. 1, no. 2, pp. 814–819, 2016.

[17] D. Barnes, W. Maddern, and I. Posner, "Find your own way: Weakly-supervised segmentation of path proposals for urban autonomy," in *ICRA*. IEEE, 2017, pp. 203–210.

[18] N. Hirose, A. Sadeghian, M. Vázquez, P. Goebel, and S. Savarese, "Gonet: A semi-supervised deep learning approach for traversability estimation," *arXiv preprint arXiv:1803.03254*, 2018.

[19] C. Cunningham, W. L. Whittaker, and I. A. Nesnas, "Improving slip prediction on mars using thermal inertia measurements," *RSS*, 2017.

[20] C. Ordonez, R. Alicea, B. Rothrock, K. Ladyko, M. Harper, S. Karumanchi, L. Matthies, and E. Collins, "Modeling and traversal of pliable materials for tracked robot navigation," in *Unmanned Systems Technology XX*, vol. 10640. SPIE, 2018, p. 106400F.

[21] M. Hein, M. Andriushchenko, and J. Bitterwolf, "Why relu networks yield high-confidence predictions far away from the training data and how to mitigate the problem," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2019, pp. 41–50.

[22] D. Bozhinoski, D. Di Ruscio, I. Malavolta, P. Pelliccione, and I. Crnkovic, "Safety for mobile robotic systems: A systematic mapping study from a software engineering perspective," *Journal of Systems and Software*, vol. 151, pp. 150–179, 2019.

[23] M. Haselmann, D. P. Gruber, and P. Tabatabai, "Anomaly detection using deep learning based image completion," in *2018 17th IEEE International Conference on Machine Learning and Applications (ICMLA)*. IEEE, 2018, pp. 1237–1242.

[24] B. Schölkopf, R. C. Williamson, A. J. Smola, J. Shawe-Taylor, and J. C. Platt, "Support vector method for novelty detection," in *Advances in neural information processing systems*, 2000, pp. 582–588.

[25] T. Schlegl, P. Seeböck, S. M. Waldstein, U. Schmidt-Erfurth, and G. Langs, "Unsupervised anomaly detection with generative adversarial networks to guide marker discovery," in *International Conference on Information Processing in Medical Imaging*. Springer, 2017, pp. 146–157.

[26] H. Choi and E. Jang, "Generative ensembles for robust anomaly detection," *arXiv preprint arXiv:1810.01392*, 2018.

[27] M. Sakurada and T. Yairi, "Anomaly detection using autoencoders with nonlinear dimensionality reduction," in *Proceedings of the MLSDA 2014 2nd Workshop on Machine Learning for Sensory Data Analysis*. ACM, 2014, p. 4.

[28] D. J. Rezende and S. Mohamed, "Variational inference with normalizing flows," *arXiv preprint arXiv:1505.05770*, 2015.

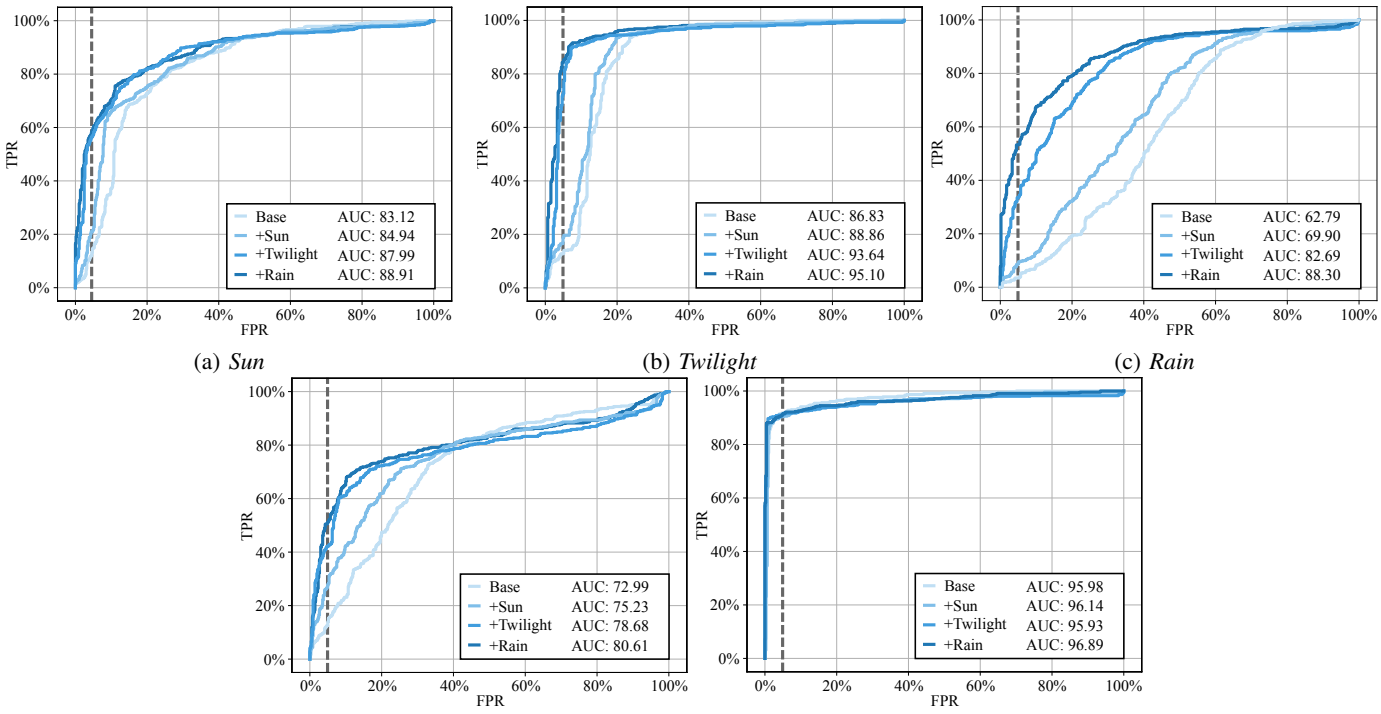
[29] L. Dinh, J. Sohl-Dickstein, and S. Bengio, "Density estimation using real nvp," *arXiv preprint arXiv:1605.08803*, 2016.

[30] D. P. Kingma and P. Dhariwal, "Glow: Generative flow with invertible 1x1 convolutions," in *Advances in Neural Information Processing Systems*, 2018, pp. 10 215–10 224.

[31] H. Badino, D. Huber, Y. Park, and T. Kanade, "Fast and accurate computation of surface normals from range images," in *2011 IEEE International Conference on Robotics and Automation*. IEEE, 2011, pp. 3084–3091.

[32] M. Hutter, C. Gehring, D. Jud, A. Lauber, C. D. Bellicoso, V. Tsounis, J. Hwangbo, K. Bodie, P. Fankhauser, M. Bloesch *et al.*, "Anymal-a highly mobile and dynamic quadrupedal robot," in *IROS*. IEEE, 2016, pp. 38–44.

[33] D. P. Kingma and J. Ba, "Adam: A method for stochastic optimization," *arXiv preprint arXiv:1412.6980*, 2014.



(a) *Sun* (b) *Twilight* (c) *Rain*

(d) *Wet* (e) *Fire*

Fig. 5: ROC curves for *NVP Fixed Features RGB+D* on test sets trained with incrementally more data. The 5%FPR threshold is indicates by a dashed grey line. The curves shift towards the left with more data, which implies improved performance at low false-positive rates. Note that a low false-positive rate is our desired operating domain as false positives can cause catastrophic failure.