2020 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)
October 25-29, 2020, Las Vegas, NV, USA (Virtual)

Anomaly Detection for Autonomous Guided Vehicles using Bayesian Surprise

Ozan Çatal, Sam Leroux, Cedric De Boom, Tim Verbelen and Bart Dhoedt*

***Abstract*— As warehouses, storage facilities and factories become more expanded and equipped with smart devices, there is a substantial need for rapid, intelligent and autonomous detection of unusual and potentially hazardous situations, also called anomalies. In particular for Autonomous Guided Vehicles (AGVs) that drive around these premises independently, unforeseen obstructions along their path—e.g. a cardboard box in the middle of a corridor or bumps in the floor—and sudden or unexpected actions executed by personnel—e.g. someone walking in a restricted area—make it hard for AGVs to navigate safely. We therefore propose a novel approach to detect such anomalies in an unsupervised manner by measuring Bayesian surprise: whenever an event is observed that does not align with the agent's prior knowledge of the world, this event is deemed surprising and could indicate an anomaly. This paper lays out the details on how to learn both the prior and posterior models of an AGV that drives around a warehouse and observes the environment through an RGBD camera. In the experiments we show that our Bayesian surprise approach outperforms a baseline that is traditionally used to detect anomalies in sequences of images.**

I. Introduction

Autonomous guided vehicles (AGVs) are gaining use in warehouses and manufacturing halls. Initially, AGVs were mainly guided by magnetic tape sensors and line following algorithms [1] or laser beacons in the infrastructure [2]. However, these solutions are limited in flexibility, often making use of a hard wired path, and come at a significant infrastructure cost. More recently, simultaneous localization and mapping (SLAM) approaches have gained attention for AGV navigation [3], which require no changes to the infrastructure, but operate using high dimensional sensors such as lidar scanners or RGB(D) cameras [4].

Although these systems allow AGVs to navigate through the environment, their sensing is typically limited to detecting (and avoiding) obstacles. However, for a human operator it would be much more informative to get notifications of any anomaly occurring in the environment. Examples of such anomalies in a warehouse could be a box left at a wrong location, people walking in prohibited areas or bumps in the floor. The biggest challenge in these use cases is how to define an anomaly, and to collect and label a dataset as such.

In this paper, we propose a novel, holistic approach for detecting anomalies, using purely unsupervised learning techniques, omitting the need labeled anomaly data. To determine whether an event is an anomaly we evaluate the Bayesian surprise, which is the difference between the prior expectation of an event happening and the actual outcome of the event. In order to estimate this quantity, we jointly learn a prior and posterior latent dynamics model from sensor data using artificial neural networks. We assume our prior model has access to proprioceptive action feedback, i.e. it knows the action commands sent to the AGV. To evaluate our approach, we collect a dataset of camera-action sequences in a warehouse using a Kuka[1] Youbot base platform. We show that our model is able to spot anomalies such as bumps in the floor, or people walking at unexpected places.

The remainder of this paper is organized as follows: We first discuss related work concerning anomaly detection and Bayesian surprise. In Section III we introduce our approach to anomaly detection and the dataset we applied it to is described in section IV. We discuss our results in section V and end with some possible directions for future work.

II. Related Work

A. Anomaly Detection

Anomaly detection refers to the problem of finding patterns in data that seldom occur [5]. We call these non-conforming patterns anomalies or surprises. There are many applications of anomaly detection, ranging from intrusion detection systems (IDS) in cybersecurity [6] to remote sensing [7] and IoT [8]. A common distinction in anomaly detection approaches is inspired by the way anomalies are modeled. An anomaly can be defined as statistical, proximity-based, or deviation-based [9]. When the data is modelled statistically, an anomaly is a data point with a low probability of occurring. Proximity-based approaches, however, will model anomalies as a sample that is isolated from the rest of the data according to some distance metric. Deviation based approaches use the reconstruction error between the original sample and its reconstruction from a low dimensional latent vector.

Anomaly detection approaches can also be classified based on their training method: supervised, semi-supervised or unsupervised. When using a supervised training method the anomaly detector learns from labeled data. Semi-supervised approaches learn from data that is only partially labeled, whilst unsupervised methods learn from unlabeled data. For an in-depth overview of recent advances in these three areas of anomaly detection we refer to [10].

Anomaly detection methods have been applied in robotics, for example to spot problems with sensors and actuators [11], or to detect anomalies in camera data [12]. In these use

*All authors are with IDlab, Ghent University, 9000 Gent, Belgium {firstname}.{lastname}@ugent.be

[1]https://www.kuka.com/en-de

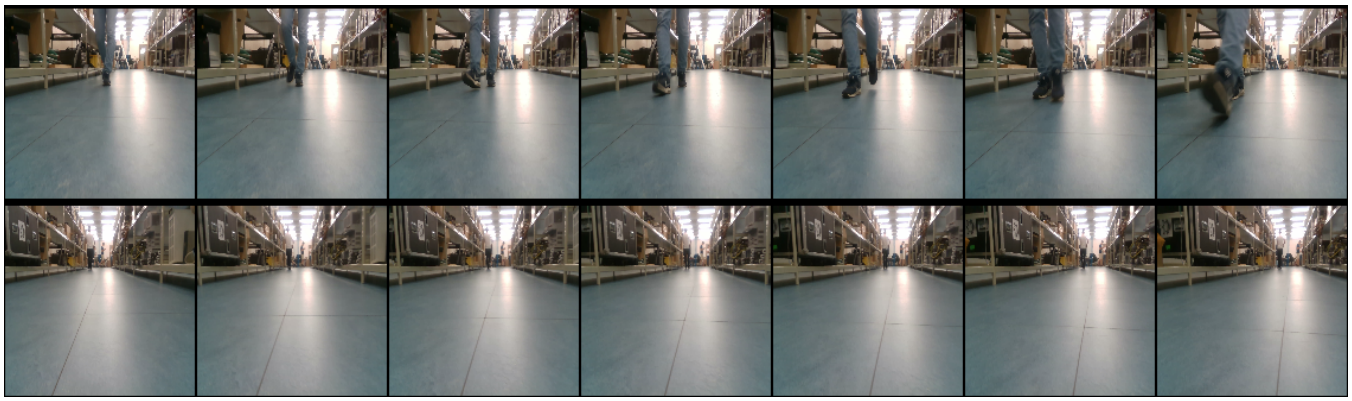


Fig. 1: An excerpt from our dataset showing two subsequences from the train set. The first subsequence shows a person walking towards the AGV. The second shows a person walking away from the AGV in the distance.

cases we often have to rely on explicit models of the robot or vehicle dynamics [13], [14]. On video data, recent approaches use deep learning and variational autoencoders (VAE) to estimate the observation likelihood [15], or use the reconstruction error as anomaly detection mechanism [16]. An example combining both approaches is given by [17], where the authors use autoregressive flows to calculate the observation likelihood in conjunction with a MSE based anomaly signal.

B. Novelty and Surprise

As discussed above, anomalies are defined as out-of-distribution events, or events not previously experienced or encountered, which is typically described as novelty [18]. However, an anomaly can also occur when an observation, even if experienced before, is unexpected in the current context. For example, an AGV might have frequently spotted human operators walking by, although not in a certain forbidden entry area. Therefore, anomalies might be better captured by surprise, which is defined as the result of a discrepancy between an expectation and an observed actuality [19]. When considering this concept from a Bayesian perspective we can interpret surprise as the discrepancy between an agent's prior beliefs and its posterior beliefs over a (latent) state space, which is characterized as the KL Divergence between a prior and posterior distribution [20]:

$$D_{KL}(P||Q) = \int_s P(s) \log \frac{P(s)}{Q(s|o)} ds \tag{1}$$

where P is the current prior belief over state s and Q the posterior belief over s after observing observation o.

III. Approach

As laid out in the introduction, we will train an AGV to detect anomalies while it is navigating through a warehouse setting. In this process, the AGV learns to maintain a compact latent state representation of the environment by observing camera input and its own actions. These state representations can then be used to calculate Bayesian surprise according to Equation (1), which is our chosen metric to spot anomalies. To estimate this Bayesian surprise, we need both a prior and posterior belief over the latent state space. We obtain this by building a generative model of the joint distribution over sequences of hidden states $\boldsymbol{s}$ and observations $\boldsymbol{o}$, where we condition on the actions $\boldsymbol{a}$—assuming that we have proprioceptive feedback available—and that the state is Markov:

$$P(\boldsymbol{o}, \boldsymbol{s}|\boldsymbol{a}) = P(s_0) \prod_{t=1}^{T} P(o_t|s_t) P(s_t|s_{t-1}, a_{t-1}) \tag{2}$$

We can then maximize the log evidence $\log P(\boldsymbol{o})$ by introducing an approximate posterior distribution $Q(s_t|s_{t-1}, a_{t-1}, o_t)$, and maximizing the evidence lower bound for every timestep [21]:

$$\begin{aligned} \text{ELBO} = &-D_{KL}[Q(s_t|s_{t-1}, a_{t-1}, o_t)||P(s_t|s_{t-1}, a_{t-1})] \\ &+ \mathbb{E}[\log P(o_t|s_t)] \end{aligned} \tag{3}$$

We do so by using deep neural networks to represent both the prior distribution $P(s_t|s_{t-1}, a_{t-1})$ and posterior distribution $Q(s_t|s_{t-1}, a_{t-1}, o_t)$, as well as the likelihood model $P(o_t|s_t)$. Both prior and posterior distributions are parameterized as i.i.d multivariate Gaussian distributions using the reparameterization trick. This is similar to the Variational Autoencoder (VAE) [22], except that we optimize over sequences in time, and also learn a separate, action-conditioned prior instead of using a fixed, standard normal prior. A visual representation of our model is shown in Fig. 2. As our prior and posterior model are conditioned on the previous state s_{t-1}, these are in fact recurrent neural networks with a stochastic hidden state. We found that, especially for the prior model that has no access to the observations, it is difficult to learn the latent space dynamics when only forwarding recurrent information in a probabilistic manner, i.e. by sampling. To mitigate this, we can also allow for a deterministic forwarding of information by providing the Prior model with an explicit recurrent memory cell where it can store relevant information. We experiment with either a long short-term memory (LSTM) [23] or a gated recurrent cell (GRU) [24]. Note that the usage of these additional

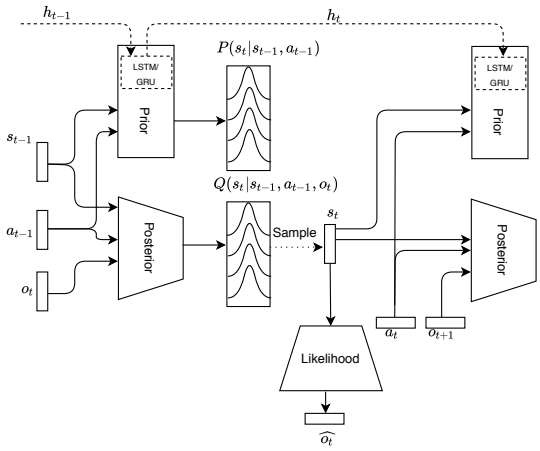


Fig. 2: Overview of how the different components interact. We process the previous state and action vectors s_{t-1} and a_{t-1} to generate a prior estimate of the current latent state distribution $P(s_t|s_{t-1}, a_{t-1})$. Next we process the previous state, action and observation vectors to a estimated posterior state distribution $Q(s_t|s_{t-1}, a_{t-1}, o_t)$. The next state vector is then sampled from the posterior distribution (indicated by the dotted line) and passed through the Likelihood model to generate a reconstruction of the input observation $\widehat{o}_t$. Optionally the Prior model can be instantiated as a GRU or LSTM cell, in which case the recurrent cell's hidden state h_t is used as a deterministic information path between successive calls to the Prior model.

recurrent cells do not break the Markov assumption, as each cell only sees information of the previous timestep.

We train the models unsupervised on recordings of observation-action sequences. To detect anomalies at inference time it is only necessary to evaluate the prior and posterior models, and calculate the KL divergence. At each timestep the Bayesian Surprise between prior and posterior is calculated, which can be used as a measure of anomalousness. We do not use the likelihood model for anomaly detection, as the high dimensional pixel likelihood values typically lack any form of interpretability.

IV. AGV Anomaly Dataset

We recorded our own dataset of an AGV navigating between racks of storage space in a warehouse setting. We use a Kuka Youbot mobile robotics platform fitted with an Intel Realsense[2] camera (shown in Fig. 3), which navigates autonomously through the aisles using ROS [25] and hector SLAM [26] with a front-facing lidar sensor.

In total we recorded 105 minutes of real world data, divided in 317 train sequences and 7 test sequences. The train set sequences consist of the AGV driving around in either empty aisles or with people walking around next to the AGV. An example of the train set sequences is given in

[2]https://www.intel.com/content/www/us/en/architecture-and-technology/realsense-overview.html

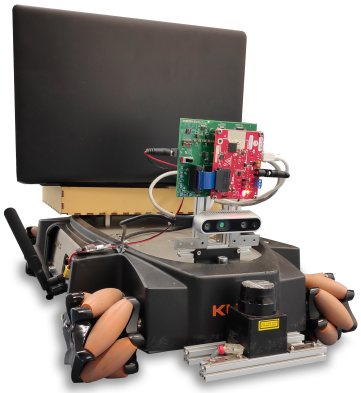


Fig. 3: The Kuka Youbot used to record the dataset.

Fig. 1. During the recording of the test sequences we also introduced a number of anomaly events:

- a cable gutter on the AGV's path (Fig. 4a)
- a PC-case on the floor (Fig. 4b)
- a metalic pole on the floor (Fig. 4c)
- a person walking on an area where there is nobody in the train set (Fig. 4d)

To enable quantitative evaluation of model performance on the dataset we label the test set. A label is provided for each recorded timestep indicating whether an anomaly is present or not

To enable quantitative evaluation of model performance on the test set, we label each frame of the test set sequences, indicating whether an anomaly is visually present or not.

V. Experiments

We evaluate our approach on the dataset described in the previous section where we train the model on the 317 sequences of the train set. We extend the dataset by taking random subsequences from the train set, to increase training stability.

We instantiate our posterior and likelihood model as a VAE where the encoder is also conditioned on a latent state vector s and an action vector a. We experiment with 3 possible architectures for the prior model. In the first we allow only for pure probabilistic hidden state, by instantiating the prior as a multilayer perceptron with variational output. In the second and third architecture we allow for deterministic hidden states as well as a probabilistic hidden state. The deterministic hidden state is drawn from the hidden states of either an LSTM cell or a GRU cell. The probabilistic hidden state is provided by a variational output from these cells. An overview of the used neural network architectures is given in Table I. All convolutional layers use a kernel of size 3x3. The Likelihood model uses convolutions with a stride of one, and upsampling the intermediate reconstruction is done by nearest neighbour interpolation.

We compare the three versions of our technique with a common anomaly detection approach for video sequences: the Spatio-Temporal Stacked frame AutoEncoder

(a) Cable gutter anomaly (b) PC case anomaly (c) Pole anomaly (d) Person anomaly

Fig. 4: Example of the different anomalies present in the test set.

	Layer	Neurons/Filters	activation function
Posterior	Convolutional	8	Leaky ReLU
	Convolutional	16	Leaky ReLU
	Convolutional	32	Leaky ReLU
	Convolutional	64	Leaky ReLU
	Convolutional	128	Leaky ReLU
	concat	N.A.	N.A.
	Linear	2 x N^o states	Softplus
Likelihood	Linear	128 x 8 x 8	Leaky ReLU
	Convolutional	128	Leaky ReLU
	Convolutional	64	Leaky ReLU
	Convolutional	32	Leaky ReLU
	Convolutional	16	Leaky ReLU
	Convolutional	8	LeakyReLU
Prior	Linear	400	Leaky ReLU
	Linear	2 x N^o states	Softplus
GRU	GRU cell	400	Leaky ReLU
	Linear	2 x N^o states	Softplus
LSTM	LSTM cell	400	Leaky ReLU
	Linear	2 x N^o states	Softplus

TABLE I: Neural network architectures. All convolutional layers have a 3x3 kernel. The convolutional layers in the Likelihood model have a stride and padding of 1 to ensure that they preserve the input shape. Upsampling is done by nearest neighbour interpolation.

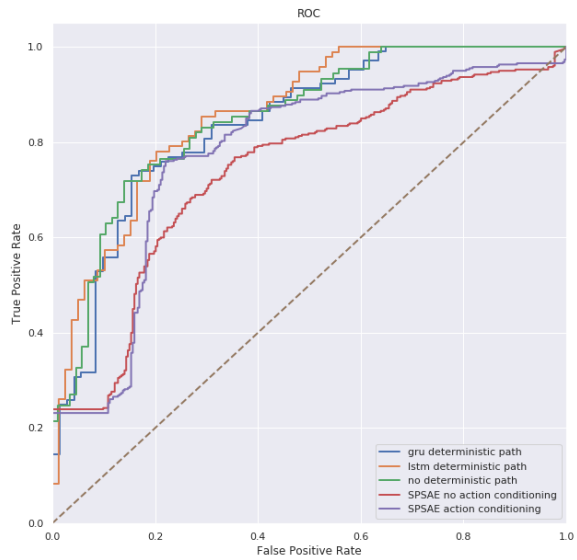


Fig. 5: ROC curve of the different approaches. Comparing various forms of our approach with a STSAE baseline and an action conditioned STSAE.

(STSAE) [16] since many successful approaches for anomaly detection on video rely on generative modeling, with many approaches building further upon the STSAE [10]. We instantiate the baseline with a neural network with capacity roughly equal to the capacity of our proposed models. We level the playing field by also experimenting with an STSAE whose encoder conditions on the current action vector. We detect anomalies in the baseline by inspecting the MSE between observation and reconstruction. We use the area under the receiver operation curve (AUROC) [27] as an evaluation metric. The AUROC values are given in Table II. The ROC curves are shown in Fig. 5.

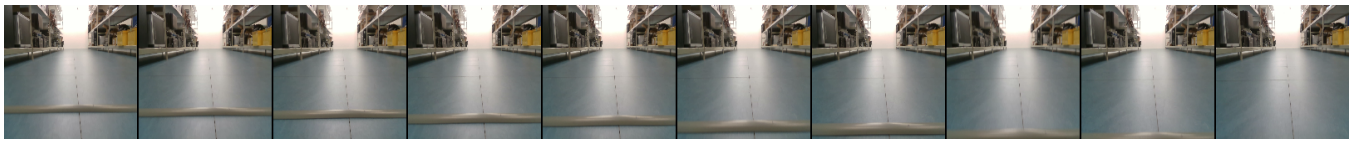
model	AUROC
STSAE, no action conditioning	0.75
STSAE, action conditioning	0.77
Ours, no cell	0.81
Ours, with GRU	0.79
Ours, with LSTM	0.86

TABLE II: Area under the ROC for the different approaches.

We see that our model using an LSTM cell performs best. An example of the surprise in conjunction with the evolution of the prior and posterior states is given in Fig. 6. The sequence consists of a cable gutter lying on the path of the AGV. We see that although the gutter is not detectable in the reconstructed observation (Fig. 6b), it is detectable in the Posterior model (Fig. 6d) and the Prior model (Fig. 6c) when the robot drives over it. The corresponding Bayesian surprise is given in Fig. 6e. In this figure we see strong surprisal peaks coinciding with the deviations in the Prior and Posterior models. The initial surprise peak indicates the bootstrapping of the Prior model, as can be seen in Fig. 6c and 6d where at the initial timestep all latent variables need to settle first. A video illustrating the surprise in conjunction with the corresponding observations for both anomalous and anomaly-free sequences can be found in the supplementary material.

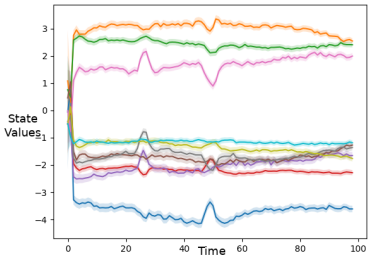
Discussion

Our approach leverages a learned latent dynamics model to enable the calculation of the difference between prior

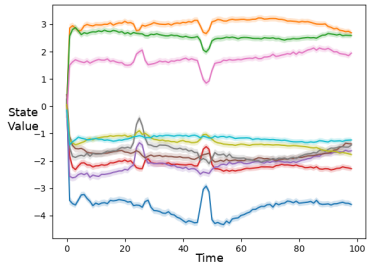


(a) Ground Truth sequence of the anomaly

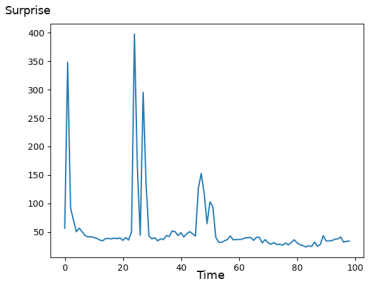
(b) Decoder reconstruction of the anomaly



(c) Prior state estimate of the entire sequence



(d) Posterior state estimate of the entire sequence



(e) Surprise evolution of the entire sequence

Fig. 6: A rollout of the model. Fig. (a) and (b) show the first 10 frames of the ground truth observation and the models reconstruction of the observation. Time flows from left to right in discrete timesteps. Fig. (c) shows the prior state estimate of the entire sequence. Fig. (d) shows the corresponding posterior estimates. In Fig. (e) the final surprise graph is plotted. We see that the model is surprised at the beginning and end of the anomaly frames and surprise reduces when the frames contain no anomaly.

and posterior believes of the world. Dynamics learning has typically been a part of some reinforcement learning approaches, but we show that the availability of a sufficient dynamics model can be of benefit for other tasks in robotics. Although dynamics model learning has typically been a less popular approach to doing RL, it has seen renewed interest recently. Some innovative new approaches to dynamics learning have been presented by Hafner et al. [28] and Lee et al. [29]. Both these approaches to behavior learning utilize a dynamics model similar to ours. PlaNet by Hafner et al. [28] utilizes a generative model similar to ours which also combines a LSTM with a VAE to achieve a hybrid stochastic-deterministic state space. Stochastic Latent Actor Critic by Lee et al. [29] uses a purely stochastic state space, utilizing a autoregressive latent variable model to achieve an expressive dynamics model. Similar to PlaNet we found that the inclusion of a deterministic component in the latent space works best in our case.

When categorizing our approach in one of the three anomaly detection types from Section II-A our approach falls in the deviation based category. Although we use a probabilistic framework and stochastic latent spaces, the probabilities within the generative should not be interpreted as a probability of occurrence.

VI. Conclusion and Future Work

To the best of our knowledge we are the first to utilize the Bayesian surprise in conjunction with a latent dynamics model to detect anomalies. We see that in the case of a mobile platform the incorporation of proprioceptive information in the form of an action vector and the use of latent space dynamics allows for better anomaly detection performance. As future work, we want to further explore the case of unsupervised anomaly detection using Bayesian surprise. On the one hand we want to collect more data in a much wider environment, to see how well the approach will scale. On the other hand we also would like to look into incorporating other sensor modalities, such as lidar or radar.

Acknowledgments

Ozan Catal is funded by a Ph.D. grant of the Flanders Research Foundation (FWO). This research received funding from the Flemish Government (AI Research Program).

References

[1] T. Muller, *Automated Guided Vehicles*. IFS, 1983.
[2] H. Durrant-Whyte, D. Rye, and E. Nebot, "Localization of autonomous guided vehicles," in *Robotics Research*. Springer, 1996, pp. 613–625.
[3] G. Bresson, Z. Alsayed, L. Yu, and S. Glaser, "Simultaneous Localization and Mapping: A Survey of Current Trends in Autonomous Driving," *IEEE Transactions on Intelligent Vehicles*, vol. 2, no. 3, pp. 194–220, 2017.

[4] J. Fuentes-Pacheco, J. Ruiz-Ascencio, and J. M. Rendón-Mancha, "Visual simultaneous localization and mapping: a survey," *Artificial Intelligence Review*, vol. 43, no. 1, pp. 55–81, 2015.

[5] V. Chandola, "Chandola, Banerjee, Kumar - 2009 - Anomaly detection A survey," *ACM Computing Surveys*, no. September, pp. 1–72, 2009.

[6] A. Javaid, Q. Niyaz, W. Sun, and M. Alam, "A Deep Learning Approach for Network Intrusion Detection System," in *Proceedings of the 9th EAI International Conference on Bio-inspired Information and Communications Technologies (Formerly BIONETICS)*, ICST, Brussels, Belgium, Belgium, 2016, pp. 21–26. [Online]. Available: http://dx.doi.org/10.4108/eai.3-12-2015.2262516

[7] J. E. Ball, D. T. Anderson, and C. S. Chan, "Comprehensive survey of deep learning in remote sensing: theories, tools, and challenges for the community," *Journal of Applied Remote Sensing*, vol. 11, no. 04, p. 1, 2017.

[8] M. Mohammadi, A. Al-Fuqaha, S. Sorour, and M. Guizani, "Deep learning for IoT big data and streaming analytics: A survey," *IEEE Communications Surveys and Tutorials*, vol. 20, no. 4, pp. 2923–2960, 2018.

[9] C. C. Aggarwal, *Outlier Analysis*, 2nd ed. Springer Publishing Company, Incorporated, 2016.

[10] R. Chalapathy and S. Chawla, "Deep Learning for Anomaly Detection: A Survey," pp. 1–50, 2019. [Online]. Available: http://arxiv.org/abs/1901.03407

[11] R. Gupta, Z. T. Kurtz, S. Scherer, and J. M. Smereka, "Open Problems in Robotic Anomaly Detection," 2018. [Online]. Available: http://arxiv.org/abs/1809.03565

[12] W. Lawson, L. Hiatt, and K. Sullivan, "Detecting Anomalous Objects on Mobile Platforms," *IEEE Computer Society Conference on Computer Vision and Pattern Recognition Workshops*, pp. 1426–1433, 2016.

[13] E. Lampiri, "Sensor anomaly detection and recovery in a nonlinear autonomous ground vehicle model," *2017 Asian Control Conference, ASCC 2017*, vol. 2018-Janua, pp. 430–435, 2018.

[14] F. van Wyk, Y. Wang, A. Khojandi, and N. Masoud, "Real-Time Sensor Anomaly Detection and Identification in Automated Vehicles," *IEEE Transactions on Intelligent Transportation Systems*, no. August, 2018.

[15] J. An and S. Cho, "SNU Data Mining Center 2015-2 Special Lecture on IE Variational Autoencoder based Anomaly Detection using Reconstruction Probability," 2015.

[25] M. Quigley, K. Conley, B. P. Gerkey, J. Faust, T. Foote, J. Leibs, R. Wheeler, and A. Y. Ng, "ROS: an open-source Robot Operating System," in *ICRA Workshop on Open Source Software*, 2009.

[16] M. Hasan, J. Choi, J. Neumann, A. K. Roy-Chowdhury, and L. S. Davis, "Learning temporal regularity in video sequences," *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, vol. 2016-Decem, pp. 733–742, 2016.

[17] D. Abati, A. Porrello, S. Calderara, and R. Cucchiara, "Latent Space Autoregression for Novelty Detection," 2018. [Online]. Available: http://arxiv.org/abs/1807.01653

[18] A. Barto, M. Mirolli, and G. Baldassarre, "Novelty or Surprise?" *Frontiers in Psychology*, vol. 4, no. DEC, pp. 1–15, 2013.

[19] P. Ekman and R. J. Davidson, *The Nature of Emotion: Fundamental Questions*. Oxford University Press USA, 1994.

[20] P. Baldi and L. Itti, "Of bits and wows: A Bayesian theory of surprise with applications to attention," *Neural Networks*, vol. 23, no. 5, pp. 649–666, 2010. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S0893608009003256

[21] S. Bohez, T. Verbelen, S. Leroux, E. De Coninck, B. Vankeirsbilck, P. Simoens, and B. Dhoedt, "Robot navigation using a variational dynamics model for state estimation and robust control," in *Deep Reinforcement Learning Workshop NeurIPS2018, part of 32nd Conference on Neural Information Processing Systems*, 2018, pp. 1–15.

[22] D. P. Kingma and M. Welling, "Auto-Encoding Variational Bayes," *arXiv preprint*, 2014.

[23] S. Hochreiter and J. Schmidhuber, "Long Short-Term Memory," *Neural Computation*, vol. 9, no. 8, pp. 1735–1780, 1997.

[24] K. Cho, B. van Merrienboer, C. Gulcehre, D. Bahdanau, F. Bougares, H. Schwenk, and Y. Bengio, "Learning Phrase Representations using RNN Encoder–Decoder for Statistical Machine Translation," *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pp. 1724–1734, 2014. [Online]. Available: http://aclweb.org/anthology/D14-1179

[26] S. Kohlbrecher, J. Meyer, O. von Stryk, and U. Klingauf, "A Flexible and Scalable SLAM System with Full 3D Motion Estimation," in *Proc. IEEE International Symposium on Safety, Security and Rescue Robotics (SSRR)*. IEEE, nov 2011.

[27] T. Fawcett, "An introduction to ROC analysis," *Pattern Recognition Letters*, vol. 27, no. 8, pp. 861–874, 2006.

[28] D. Hafner, T. Lillicrap, I. Fischer, R. Villegas, D. Ha, H. Lee, and J. Davidson, "Learning latent dynamics for planning from pixels," 2018.

[29] A. X. Lee, A. Nagabandi, P. Abbeel, and S. Levine, "Stochastic latent actor-critic: Deep reinforcement learning with a latent variable model," 2019.