Comparing autoencoder-based approaches for anomaly detection in highway driving scenario images

Citation for the original published paper (version of record):
Mosin, V., Staron, M., Tarakanov, Y. et al (2022). Comparing autoencoder-based approaches for anomaly detection in highway driving scenario images. SN Applied Sciences, 4.
http://dx.doi.org/10.1007/s42452-022-05160-3

Research Article

Comparing autoencoder-based approaches for anomaly detection in highway driving scenario images

Vasilii Mosin[1,2] · Miroslaw Staron[2] · Yury Tarakanov[1] · Darko Durisic[1]

Received: 4 September 2021 / Accepted: 7 September 2022
Published online: 28 November 2022

Abstract

Autoencoder-based anomaly detection approaches can be used for precluding scope compliance failures of the automotive perception. However, the applicability of these approaches for the automotive domain should be thoroughly investigated. We study the capability of two autoencoder-based approaches using reconstruction errors and bottleneck-values for detecting semantic anomalies in automotive images. As a use-case, we consider a specific highway driving scenario identifying if there are any vehicles in the field of view of a front-looking camera. We conduct a series of experiments with two simulated driving scenario datasets and measure anomaly detection performance for different cases. We systematically test different autoencoders and training parameters, as well as the influence of image colors. We show that the autoencoder-based approaches demonstrate promising results for detecting semantic anomalies in highway driving scenario images in some cases. However, we also observe the variability of anomaly detection performance between different experiments. The autoencoder-based approaches are capable of detecting semantic anomalies in highway driving scenario images to some extent. However, further research with other use-cases and real datasets is needed before they can be safely applied in the automotive domain.

Keywords Anomaly detection · Autoencoders · Automotive perception

1 Introduction

Contemporary automotive software systems are designed and implemented using control-loop based algorithms [1]. Despite the advantages of being predictable and controllable, these algorithms have limitations—e.g. the software design needs to handle all inputs provided to these algorithms. These active safety systems use multiple sensors to recognize driving situations and provide input to decision algorithms, which lead to a trade-off between the number (and type) of sensors used and the quality of the input as more sensors increase the quality while also increasing the costs. As much as all advanced driver-assistance systems, autonomous driving functions in modern cars rely, among other sensors, on cameras to detect roadway, lanes, and traffic objects. Due to advances in computational capacity of on-board computers and progresses in deep learning (DL), DL-based algorithms are widely used for various perception tasks (e.g., object detection and scene segmentation) in the automotive industry [2]. Supervised learning algorithms used for image recognition are one example of such DL-based systems. Despite of their advantages, they have the disadvantage of being "data hungry" which increases the cost of development or reduces reliability of these systems during operation [3]. Unsupervised learning algorithms, on the other hand, do not require labelled data and can therefore

✉ Vasilii Mosin, vasilii.mosin@volvocars.com | [1]Division of Research and Development, Volvo Car Corporation, Gothenburg, Sweden. [2]Department of Computer Science and Engineering, Chalmers | University of Gothenburg, Gothenburg, Sweden.

SN Applied Sciences (2022) 4:334 | https://doi.org/10.1007/s42452-022-05160-3

be used more flexibly. However, one of the challenges is their ability to perform correctly – the performance can differ based on their architecture, type of images or even the task for which they are used [4]. For example, traffic signs classification algorithm will not work in American countries if it was trained on European traffic signs. Scope compliance failures are of greater importance in the automotive industry because they require to pay attention to complex environmental details, while the other types of failures are just the questions of algorithmic and technical capabilities of a system and data quality. These differences need to be studied in detail in order to establish which algorithms can be used in which situations.

Therefore, in this paper, we set off to investigate one of unsupervised learning algorithms for image processing – autoencoders [5, 6]. Autoencoders are usually used to identify anomalies (or novelty) in images which is often used to reduce noise. These anomalies can also be used to detect different driving situations – for example detecting whether there is an object on the road. In this paper, we study one specific driving situation - highway driving - and study how well autoencoders perform in identifying that there is a vehicle (or multiple vehicles) in the field of view of the front-facing camera. We chose highway driving because of two main reasons. Firstly, it is considered as one of the simpler driving scenarios mostly involving vehicles that move in the same direction, as opposed to other driving scenarios which may include oncoming vehicles, traffic lights, pedestrians, road crossings, etc. Therefore, we believe it is a good starting points for our study. Secondly, it represents one of the most beneficial driving scenarios for taking the steering wheel of the driver as it is usually performed over longer distances where driving itself is monotonous and tedious. So enabling the autonomous driving in the highway driving scenario would not only release the driver from the constant need of keeping the focus on the road, but also provide the driver with some spare time for other activities.

Autoencoders can be used as part of the safety mechanisms designed to ensure that machine learning algorithms do not cause unexpected behaviour, for instance in *safety cages* [3]. The concept originates from "fault diagnosis scheme" in the aviation domain [7]. Autonomous driving requires the development of complex architectural solutions based on the combination of stochastic and deterministic components interacting with each other, where DL perception results (e.g., object detection and scene segmentation) are used by decision-level functions (e.g. path planning) for controlling a vehicle. The goal of the "safety cage" is to detect the driving situations, which are out of the operational domain of a DL-based component, and to switch to the execution of predefined safe instructions (usually referred to as a safe mode), if such driving situations are detected, instead of continuing to rely on the perception results.

In this paper, we use the definition of *anomalous data* as the samples coming from outside of the training data distribution, in terms of machine learning, which can represent change in a driving scenario or perception context, in terms of the automotive software. In case of automotive perception with cameras, the data is presented in the form of images. Importantly, that images can contain anomalies of two different types: context and semantic. For example, a context anomaly can be a different weather condition (sunny/foggy/rainy) or landscape scenario (highway/city), and a semantic anomaly can be an unknown object (e.g., vehicle, pedestrian, animal, etc.), its type, color, position, or orientation. Context anomalies are usually presented in a large part of an image, such as a background, and therefore they can be detected just using the color and textural information of a whole image. Oppositely, semantic anomalies are usually presented in a small part of an image, and therefore they can be detected only by using local information of an image. Autoencoders are widely used for anomaly detection on images and autoencoder-based anomaly detection has been tested on automotive datasets from the safety cage perspective by Henriksson et al. [8] and Kratz [9] showing promising results. However, they have only considered context anomalies. On the contrary, there are no similar studies on semantic anomaly detection, that is a fairly relevant and important case for the automotive industry. Furthermore, robustness of the autoencoder-based approaches for anomaly detection regarding possible changes of the image colors has not been investigated, but it turned out that there is a strong color bias that limits their applicability.

In this paper, we address the research question of

How capable are autoencoders in detecting semantic anomalies in highway driving scenarios?

We address the question by performing 27 experiments with two different data sets and five different types of autoencoders, including four types previously used in a similar context. Our contribution is the evaluation of five different autoencoders on generated data with the goal to understand their characteristics and provide the automotive industry with the possibility to construct robust active safety systems which are based on unsupervised learning algorithms. We use AUROC (Area Under the Receiver Operational Characteristics) to measure the performance of each studied autoencoder. Our results show that convolutional autoencoders, trained using skipped connections perform best (AUROC of 0.976 and AUROC of 0.993) for the datasets studied. Our contributions are:

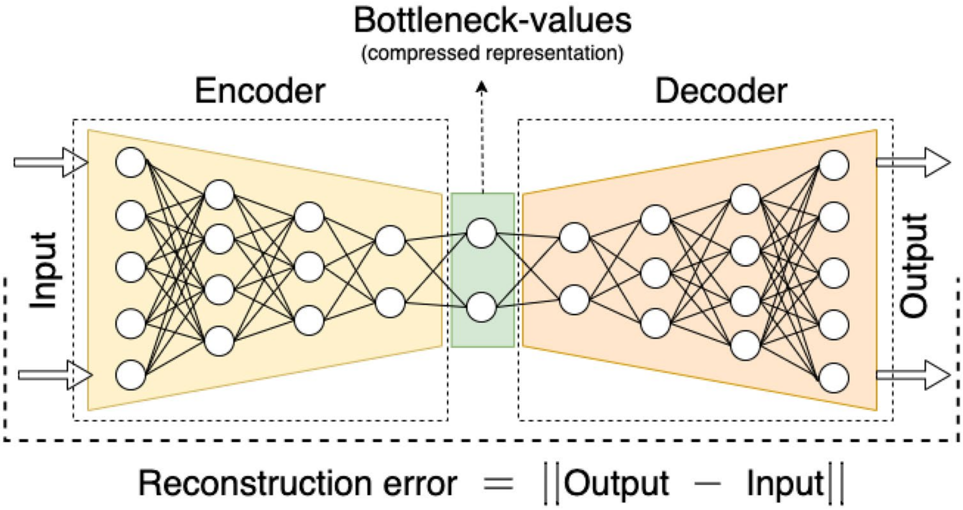
Bottleneck-values
(compressed representation)
Encoder
Decoder
Input
Output
Reconstruction error = ||Output − Input||

Fig. 1 Basic autoencoder structure

- Evaluation of five different autoencoder architectures in the context of a unique use case for this technology – a highway driving scenario.
- Development of a new dataset for this scenario, which is openly available at https://drive.google.com/file/d/1F2kPnv8N-dQqwY1_h2rDnVDpDPKx4oew/view?usp=sharing.
- Development of the autonencoder and the methodology for its training, available at https://github.com/mosin26/autoencoding_models.

These results show that the autoencoder algorithms can be used for detecting driving scenarios. However, we have observed significant differences between the performance of varios types of autoencoders, which indicates that more studies are needed to understand which autoencoder should be used for which driving situation.

The remaining of the paper is structured as follows. Section 2 contains the background information on the autoencoder-based anomaly detection. Section 3 briefly discusses recent papers about anomaly detection from autonomous driving and ML perspectives and the most popular automotive datasets. In Sect. 4, we divide the general research question mentioned above into two specific research questions. A methodology and particular implementations used in this work together with data and experiments are also explained in Sect. 4. The results of our experiments are presented in Sect. 5. Section 6 contains the discussion about the main threats to validity of our study. Finally, conclusions drawn from the results are discussed in Sect. 7.

2 Background

According to the current research studies and engineering practices, it is common to use autoencoders for anomaly detection [10]. An autoencoder is a type of neural networks that are trained to reconstruct the input they are given [11]. Traditionally, autoencoders were used for dimensionality reduction or feature learning [11]. Fig. 1 shows the structure of a general autoencoder that consists of two parts – encoder, which computes a lower-dimensional (compressed) representation of the input data in the bottleneck, and decoder, which reconstructs the output from this compressed representation of the input.

There are four different types of autoencoders presented in this work. The most basic one is a simple shallow autoencoder (SAE). It has no intermediate layers in the encoder and decoder and reconstructs the input through the bottleneck to the output directly. An autoencoder with multiple intermediate layers in the encoder and decoder is called a deep autoencoder (DAE). Variational autoencoder (VAE) is a special type of autoencoders introduced by Kingma and Welling [12]. Under the assumption that the compressed representation of the data should follow some predefined distribution, VAE is trained to learn parameters of this distribution in the bottleneck instead of learning data representation directly. Another type of autoencoders, which is usually used for images in particular, is a convolutional autoencoder (CAE). It has layers with convolutional filters for processing data that has a known grid-like topology [11].

There are two general approaches for using autoencoders for anomaly detection according to the anomaly detection review in [13]. The first approach is based on the reconstruction error (Fig. 1), which is the difference between the output and the input. Autoencoders are usually restricted in ways that allow them to reconstruct inputs only approximately, and to reconstruct only inputs that resemble the training data [11]. Therefore, the reconstruction errors will be higher for anomalous data if the autoencoder is trained only on normal data without anomalies. In this case, the reconstruction errors are used as anomaly scores. The second approach is based on the bottleneck-values (Fig. 1), which represent the compressed representation of the input data. Autoencoders often learn useful properties of the training data in the bottleneck [11]. Therefore, if the autoencoder is trained using only normal data samples without anomalies, it is possible to separate normal and anomalous data by using their bottleneck-values. In this case, to calculate anomaly scores, traditional anomaly detection methods are applied on the bottleneck-values, which contain a compressed representation of the input data learned on the training data distribution. Examples of the traditional anomaly detection approaches that can be applied on the bottleneck-values include one-class SVM (support vector machine) [14], density-based clustering [15], and distance-based methods [16]. In the paper, we refer to these two approaches as the reconstruction error approach and the bottleneck-values approach, respectively.

The advantage of using autoencoders for anomaly detection is that they are trained in unsupervised manner, meaning that no labeled data is required since the input data itself is used as the labels. Therefore, there is no need for manual labeling of driving scenarios when preparing a training dataset. The images from the training data distribution of a DL component are used for training the autoencoder. We don't need to identify anomalous scenarios in advance for the autoencoder. The idea behind the autoencoder is that it is trained with only normal data, and any other data that is not fitted to the learned normal distribution is considered as anomalous. So, having only normal data is sufficient to train the autoencoder for anomaly detection. It is extremely beneficial from the automotive perspective since anomalous driving scenarios are rare cases and it is difficult and dangerous to collect such data for the training procedure.

3 Related work

Anomaly detection is an active research field in general, and in the automotive domain in particular. In Sect. 3.1, we review the most important related work describing anomaly detection with the focus on autonomous driving systems, which show the current status of the application of autoencoders. In Sect. 3.2, we summarize the most relevant research in anomaly detection for images including autoencoder-based and related approaches, which show the state-of-the-art in the use of the technology. In Sect. 3.3, we discuss the automotive datasets that can be potentially used for the experiments with anomaly detection in driving scenarios, which can form the basis for training and evaluating the technology.

3.1 Anomaly detection for autonomous driving systems

In recent years, in the context of anomaly detection in autonomous driving systems, the research focus has been on describing the motivating the use and adaptation of anomaly detection techniques in such systems. The majority of the studies in this area was on the offline evaluation and a run-time monitoring of traffic situations.

The offline evaluation is one of the application of anomaly detection for autonomous driving systems. According to the systematic literature review on ML/DL and software engineering by Wang et al. [17], ML/DL systems are relatively less deterministic and more statistics-oriented in comparison with traditional software systems. The authors conclude that evaluation of ML/DL systems is still at an early stage, mainly relying on probing the accuracy on test data that are randomly drawn from manually labeled data. Anomaly detection can be used for selection the error prone testing data to reveal potential failures of ML/DL systems. A good example demonstrating how such selection can be used for testing DL-based autonomous driving systems is presented by Zhang et al. [18]. They used generative adversarial networks (GANs) to generate driving scenario images with anomalous weather conditions (e.g., foggy, rainy, snowy conditions). Theses images were used for evaluating an end-to-end DL model predicting steering angles for given input images. It was shown that such anomalous input images make it possible to detect behavioral inconsistencies for this model.

The run-time monitoring is another anomaly detection application for autonomous driving systems. Koopman P. and Wagner M. [19] conclude that significant

Table 1 Summary of results in anomaly detection on images.

Algorithm type	Image type	Main results	References
Neural networks–convolutional and dense	Non-automotive	AUROC between 0.64 [23] and 0.99 [24]	[23–29]
Neural networks–convolutional and dense	Automotive, signs	AUROC between 0.47 [30] to 0.99 [24]	[24–26, 30, 31]
Neural networks–convolutional and dense	Automotive, driving scenarios	Average precision from 0.18 [32] to 0.81 [33], and AUROC between 0.82 and 0.91 [34]	[32–37]
Autoencoders and GANs	Non-automotive	AUROC of 0.62 [38] to 1.0 [39]	[23, 29, 38, 40–42]
Autoencoders and GANs	Automotive, signs	AUROC between 0.64 [23] and 0.99 [43]	[23, 31, 43–45]
Autoencoders and GANs	Automotive, driving scenarios	AUROC from 0.66 to 0.87 [46], Accuracy of 0.85 – 0.9 [47]	[46–48]

Since different papers presents different performance metrics, we cite the ones that are used in the original publication

open technical challenges in autonomous vehicle safety consist of their validation against novel environmental inputs. In their another paper [20], they show that performing edge-case testing is crucial for autonomous vehicle validation because it seems likely that rare edge cases will be where ML/DL problems would be expected to occur. Koopman also argues, when presenting his papers, that the run-time monitoring of such edge-cases can increase the safety of perception systems based on ML/DL components. Anomaly detection can be used to detect the edge-cases. For example, Stocco et al. [21] present an approach for online input validation based on anomaly detection for self-driving cars. They use an autoencoder and time-series analysis to detect anomalous driving scenarios and predict 77% misbehaviours of DL-based vehicles. In their later work, Stocco et al. [22] show also how to continuously improve anomaly detection in autonomous driving systems during the run-time. They define novel driving scenarios as those which do not violate the system's behaviour. Then, false anomaly detection rate is reduced by learning to distinguish between these novel and true anomalous inputs.

3.2 Anomaly detection approaches for images

Table 1 presents a summary of the most relevant results for anomaly detection on images.

Table 1 shows that the results vary significantly between different algorithms and types of images. The most best results are achieved for the non-automotive images using autoencoders and GANs ([49]). The most interesting is the last row in the table, where we grouped publications which contain results of the highest similarity to ours.

In particular, Nitsch et al. [47] evaluated GANs on the KITTI dataset. However, their result showed significant errors in reconstruction, partly because of the presence of multiple objects at the same time, e.g., vehicles and pedestrials. Therefore, we needed to build on their results and tested the same dataset, but with a simpler algorithm, to further more to testing simpler algorithms on simpler datasets.

3.3 Automotive datasets

As image recognition in the automotive domain has been in the focus in the last two decades, there exist automotive datasets already. These are mostly created by research teams, although industrial datasets are also available. The majority of these datasets are prepared in the context of designing software safety systems, which means that the focus is often put on sensor fusion possibilities (e.g. the Apollo dataset, [50]). Such datasets are relevant for us, but they often contain too many details that could confound the results obtained by sole image classification using autoencoders.

At the same time, many of the datasets are placed in the scenarios which are very relevant for advanced driver assistance support, but not for studying the use of modern image recognition algorithms as a replacement for several sensors, e.g. the Cityscapes dataset [51]. The variability of images in these kind of datasets is suited for testing advanced driver support algorithms when they are close to the deployment in the cars, however, they are too diverse to be able to evaluate the suitability of the autoencoders. In particular, they do not provide a sufficient number of similar images to train the autoencoders in a controlled way.

Finally, there are datasets which focus on recognizing the driving environment, the weather, light and road conditions rather than on the ability to recognize objects on the road (objects like other traffic actors). An example of such dataset is the Mapillary dataset [52], where the images are prepared for traffic signals/signs recognition. These images are often taken from such an angle that is not relevant for recognizing objects on the road (focus is on the roadside).

A summary of the relevant datasets is presented in Table 2. These datasets we have considered and found to

Table 2 Summary of the most appropriate automotive datasets

Dataset	Driving scenarios	# of images	Annotations	References
Apollo	Expressway under various weather conditions	~thousands	bb, ss	[50]
BDD100K	Various road/weather/light conditions	100K	bb, ss	[54]
CamVid	Urban	> 700	ss	[55]
Cityscapes	Street scenes from different cities	20K	ss	[51]
DUSD	Urban traffic	5K	ss	[56]
HCI	a Street with a T-section	> 1000	ss	[57]
JAAD	Mainly urban, a few rural roads	82032	bb	[58]
Karlsruhe	Urban, daytime	~1800	bb	[59]
KITTI	Urban, rural, highway	$> 10K$	bb, ss	[60]
Lyft	Suburban, daytime	$> 30K$	bb	[61]
Mapillary	Various road/ weather/light conditions	25000	bb, ss	[52]
Udacity	Suburban sunny/overcast, daylight	$> 20K$	bb	[62]

bb stands for bounding boxes and *ss* stands for semantic segmentation for annotations

be potentially useful in the next steps of our studies, but they are often too complex for understanding the possibilities and limitations of the autoencoder networks in the automotive active safety. A more comprehensive review of the available datasets can be found in a review by Yin and Berger [53].

We have used one of these datasets to understand their applicability. We explored the KITTI dataset [60], since we've found it as the most appropriate for our usecase among the other datasets in terms of the number of images both with and without vehicles from suburban areas, resembling highways. There are only few images without vehicles on a highway in other datasets that makes them less appropriate for our study. However, the KITTI dataset was still not suitable enough for studying anomaly detection because of its low diversity in driving scenes. Also, the final size of the dataset after extracting the highway images for our use-case was small. Therefore, we decided to generate our own data, based on the requirements from our industrial partner, and use simulated images instead of real images for further experiments.

4 Research methodology

To answer the *RQ* (*How capable are autoencoders in detecting semantic anomalies in highway driving scenarios?*), we conduct experiments with two datasets, measure anomaly detection performance, and discuss the results. For analyzing the capability of the autoencoders in detecting semantic anomalies, we decided to use the highway driving scenario. The choice for this driving scenario came from our industrial partner (Volvo Cars). It is currently one of the most relevant driving scenarios for autonomous vehicles due to its potential to reduce usually joyless driving for the everyday drivers. For Volvo it is important to incrementally develop the highway pilot functionality in which detecting anomalies is one important segment. The simplest use-case for doing the initial analysis on this topic was decided to be the scenario in which cars are driven on the empty roads on which other objects including vehicles are considered as anomalies. Our main goal was to assess the performance of the autoencoder-based approaches for which the actual choice of the type of anomalous objects is not relevant. In this study we focus on the use-case where only vehicles are considered as anomalies on the empty highway roads. However, this use-case can be further extended to cover more realistic and complex situations by, e.g., excluding cars from anomalous objects and adding other types of object as anomalies.

This driving scenario is defined as the fully autonomous drive of the vehicle on the geographically approved highways according to their traffic rules and based on their road conditions such as weather condition or possible reconstruction of the road sections. These geographical locations may include both highways inside urban areas with high number of vehicles possible forming traffic jams, and rural highways where no or only a few moving vehicles are present. Full autonomy in this definition refers to the fact that drivers are allowed to have their eyes of the road and hands off the wheel, putting high requirements on the DL-based components on detecting anomalous data in order to switch to the correct predefined safe mode.

The images representing highway driving scenarios are the *objects* of our experiments. For simplicity, we use vehicles of any type (cars, trucks, motorcycles, bicycles) as semantic anomalies in highway driving scenarios. Thus, we refer to the images without vehicles as normal images and images with vehicles as anomalous images in our experiments.

For the first dataset, we use three different color cases of the images as the *independent variables*. We compare anomaly detection performance of the reconstruction error and bottleneck-values approaches and evaluate their robustness to the color changes of the images. For the second dataset, we use the training parameters as the *independent variables* and study their influence on the anomaly detection performance.

Anomaly detection performance is the *dependent variable* in our experiments. We use two ways for estimating it. Firstly, we visually compare the distributions of the obtained anomaly scores for normal and anomalous images and calculate Kolmogorov-Smirnov (KS) test statistics to get numerical estimations of their differences. The null hypothesis for KS tests used in the experiments is that the distribution of anomaly scores for normal and anomalous images are the same. If the null hypothesis can be rejected, then, probably, there is a good separation between anomaly scores for normal and anomalous images. Secondly, we consider anomaly detection as a binary classification problem and use receiver operating characteristic (ROC) curves and the corresponding area under ROC (AUROC) scores for the performance analysis. In our experiments, we have highly unbalanced datasets. In such a situation, the conventional accuracy measure has been shown to be biased towards the larger class (in our case, the images without anomalies). Therefore, we use AUROC scores which are better for imbalanced classes, used in the anomaly classification tasks. This has been done in a similar way for example in the following studies: [8, 9, 21]. The definitions of the ROC curve and AUROC scores can be found in [63] and they are based on calculating TPR (true positive rates) and FPR (false positive rates) according to the following formulas:

$$TPR = \frac{TP}{TP + FN} \tag{1}$$

$$FPR = \frac{FP}{FP + TN}, \tag{2}$$

where *TP* (true positives) – the number of anomalous images correctly classified as anomalous, *FN* (false negatives) – the number of anomalous images incorrectly classified as normal, *FP* (false positives) – the number of normal images incorrectly classified as anomalous, *TN* (true negatives) – the number of normal images correctly classified as normal. ROC curves show FPR/TPR values for different threshold values on the obtained anomaly scores. The choice of the particular threshold value depends on the end goal of the anomaly detection process, which is also discussed in the results. We have additionally calculated and analyzed area under precision-recall curve (AUPRC) scores. Precision-recall (PR) curves are more informative than ROC curves for imbalanced datasets [64], which is the case for some of our experiments. The description of PRC and AUPRC and the connection between ROC and PR curves can be found in [64].

To answer our general *RQ*, we analyze the capability of the autoencoder-based approaches for anomaly detection through the measured anomaly detection performance. Anomaly detection performance calculated in different ways described above is a good indicator of how effective the autoencoder is for the task being investigated. Additionally, we test robustness of the approaches to color changes in images because it also affects the autoencoder's capability for anomaly detection. For this, we compare anomaly detection performance of the autoencoder-based approaches for different color cases. It is important for the autoencoder to be robust to color changes since the color characteristics of the images in a real automotive application can be altered by, for example, environmental visual conditions. Thus, we formulate the following two detailed *RQs* in accordance with our general *RQ*:

- *RQ1 What is the performance of the autoencoder-based approaches for anomaly detection in driving scenario images?*
- *RQ2 What is the robustness of the autoencoder-based anomaly detection approaches to color changes in driving scenario images?*

4.1 Pilot study

We first decided to do a pilot study on semantic anomaly detection with KITTI [60] dataset included to this paper. We did the pilot study with the real dataset in the beginning of our project. The results of this study explain the need in using generated data for better understanding of the autoencoders' work for semantic anomaly detection.

We have selected the subset of 354 images resembling the highway driving scenarios from KITTI dataset. 266 empty road images were used for the training of the autoencoder. The trained autoencoder has been applied to other 57 empty road images and 31 images with vehicles and the anomaly scores were calculated for these images. After analyzing the pilot study, we decided to move to the experiments with generated datasets to examine the autoencoder-based approaches for semantic anomaly detection in more detail.

4.2 Generated driving scenario images

Synthetic driving scenario images generated with two different automotive simulators were used for the experiments in this paper. The choice of not going for using real

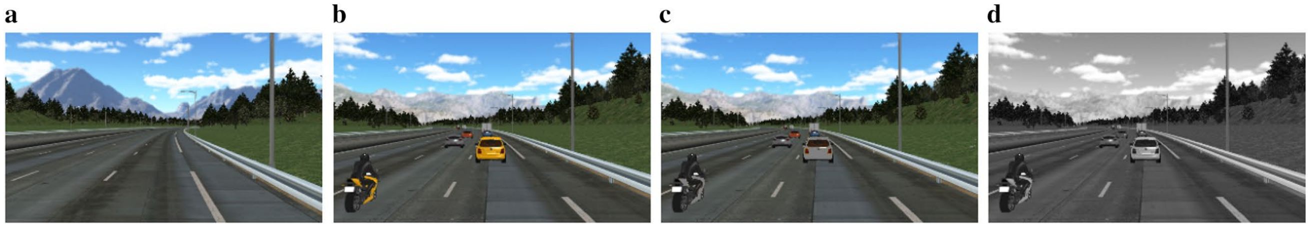

Fig. 2 An example of the images generated with Pro-SiVIC: **a** normal image, **b** anomalous original image, **c** anomalous modified image, **d** anomalous greyscaled image

images at this stage of the project is given by the need in more controlled experiments, which would require significant work to be performed with real images at this stage. With simulated images, we can generate driving scenarios needed specifically for our experiments, which is difficult to achieve with real data. Moreover, using simulation platform gives more control in terms of image manipulations, which is also more complicated when using real environment. For instance, since the focus of this paper is on semantic anomaly detection, it was required to collect the images containing different semantic anomalies but having the same context information, which is easier to get with synthetic datasets. The choice of simulation platforms was dictated by their availability and ease of use.

4.2.1 Pro-SiVIC dataset

As the first dataset, we use simulated highway environment images obtained using Pro-SiVIC[1] platform that allows to generate realistic driving scenario images. In total, there are 256 normal training images used for autoencoders training together with 100 normal and 31 anomalous testing images used for measuring the performance of the anomaly detection approaches. The original size of the images was 752x480 pixels, but we have downscaled them to 320x192 pixels. We have performed the downscaling according to the example by Henriksson et al. in [8], where they have been using the same type of data for their anomaly detection experiments. Lower resolution images allow to reduce the training time, while the objects presented in them are still clearly distinguishable. Examples of normal and anomalous images used in our project are shown in Fig. 2.

4.2.2 CARLA dataset

The second dataset was generated using CARLA [65] – an open-source autonomous driving simulator. The generated CARLA images are more feature-rich and have much more detailed environments comparing to the generated Pro-SiVIC images. An example of the images generated with CARLA is shown in Fig. 3. For instance, lights, reflections, shades, and vehicles are much more complex and natural in CARLA images than in Pro-SiVIC images. The landscape conditions were chosen to match those in Pro-SiVIC images, which is a highway environment. The original resolution was set up to 640x480 pixels, but the images were further downscaled to 320x224 pixels for the experiments to reduce a computational cost and to make them comparable to Pro-SiVIC images. Weather conditions were set up to the standard daytime sunny conditions similar to those in Pro-SiVIC images. The generated CARLA dataset contains the empty road images as well as the images with cars and riders on motorcycles and bicycles. In total, there are 389 and 130 normal images used for training and validation respectively, and 48 normal and 50 anomalous images used for testing.

4.3 Experiment design

Each experiment consists of the training and evaluation steps. During the training, an autoencoder is trained on a subset of normal images with the training parameters described in Sect. 4.4. Evaluation is performed with the trained autoencoder applied on a test set of normal and anomalous images. Firstly, anomaly scores are calculated using both the reconstruction errors and bottleneck-values for each image. Secondly, two distributions of anomaly scores for normal and anomalous images are plotted and KS test is performed for these two empirical distributions to estimate their difference. KS test is a non-parametric technique, that can be used to decide if two empirical samples are coming from the same distribution without any assumptions on underlying data distributions. The estimated difference of the two distributions serves as an indicator of the autoencoder capability in distinguishing between normal and semantically anomalous driving scenarios. Lastly, considering anomaly detection as a binary classification task, we plot ROC curves using the

[1] Pro-SiVIC by ESI Group, https://www.esi-group.com/.

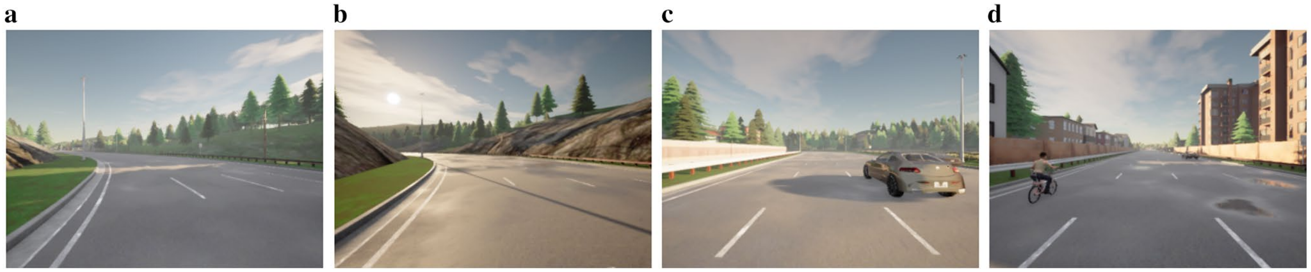

Fig. 3 An example of the images generated with CARLA: **a**, **b** normal images, **c** anomalous image with a car, **d** anomalous image with a bicycle and a cyclist

calculated anomaly scores as predicted values and calculate AUROC/AUPRC scores. This is used as an anomaly detection performance in our analysis. Additionally, we perform a visual analysis of the reconstructions obtained with the autoencoder. We provide examples of the input images, reconstructed images, and the reconstruction error images in our experiments. The reconstruction error image is calculated according to the MSE (mean squared error) definition as the per-pixel difference between the reconstructed image and the input image averaged across all color channels of the image.

We've used Pro-SiVIC dataset to analyze anomaly detection performance and compare it for different color settings. Firstly, we measure anomaly detection performance using original images (e.g. Fig. 2b) with the diversity of vehicle colors presented. This is the easiest case of anomaly detection in our experiments since most of the vehicles have colors that are very different from the background and the images containing them can be classified as anomalous based only on color information. We then use this case as a baseline for further experiments. Secondly, we change the color of all yellow vehicles to grey in the original testing images and, hereafter, we refer to such images as modified (e.g. Fig. 2c). The yellow color was selected because most of the vehicles in the original images have this color. It should be more difficult to detect anomalous images based on only color information in this case because many vehicles now have a grey color, which is presented a lot in the background as a road surface color in normal images. Comparing anomaly detection performance on modified images with the baseline demonstrates how robust anomaly detection approaches are to color changes of anomalous objects. Thirdly, we completely remove color information from all original images by converting them to greyscale (e.g. Fig. 2d). In this case, anomaly detection approaches cannot rely on colors anymore and should utilize geometric features of the environment, shapes of the objects, or other contextual information. Comparing anomaly detection performance on greyscaled images with the baseline shows how robust anomaly detection approaches are to color changes in general and how much they rely on color information for detecting anomalous images.

We've used CARLA dataset to test out different training schemes and investigate how different training parameters influence autoencoder's training ability and anomaly detection performance. In particular, we vary autoencoder's training time and architecture parameters. At the same time, CARLA images have more complex features of the environment and the objects. This makes it more challenging to perform anomaly detection for CARLA images comparing to Pro-SiVIC images. So, we investigate what is the performance of the autoencoder-based anomaly detection approaches on more complex datasets such as CARLA.

4.3.1 Anomaly detection approaches

For both the reconstruction error and bottleneck-values approaches for anomaly detection, we use the same autoencoder architecture in our experiments. The approaches differ only in a way how anomaly scores are calculated. We have experimented with a simple autoencoder model and a more complex one.

For Pro-SiVIC images, we have chosen the most generic convolutional autoencoder architecture according to existing guidelines[2]. The same autoencoder architecture was used in the experiments with KITTI dataset in the pilot study. The exact parameters of the layers were manually selected based on the pre-study experiments performed by the Autonomous Drive team internally at Volvo Cars. We provide a complete description of the autoencoder architecture based on Fig. 4 in order to make it possible for the interested reader to reproduce our results.

The autoencoder consists of three convolution layers (yellow) in the encoder part and three deconvolution layers (light blue) in the decoder part. These convolutional

[2] https://blog.keras.io/building-autoencoders-in-keras.html.

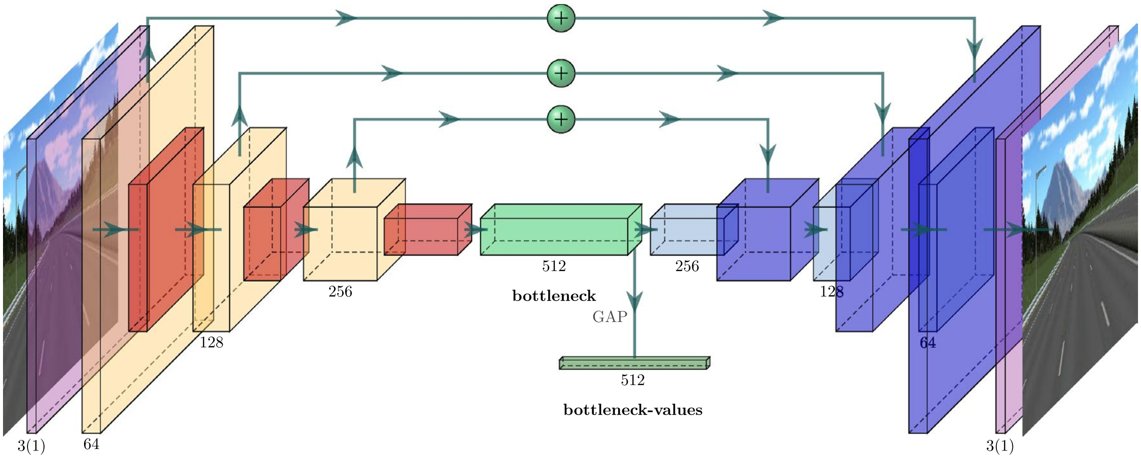

Fig. 4 Architecture of the autoencoder used in the experiments

layers allow the autoencoder to learn important features of the images, such as edges and shapes, for example. All convolution layers in the encoder part are followed by pooling layers (red), which are used to reduce the size of the images. All deconvolution layers in the decoder part are followed by unpooling layers (dark blue), which are used to increase the size of the images. We set the size of the filters in pooling and unpooling layers to 2, which means that the size of the images in each polling and unpooling layer changes by a factor of 2. The bottleneck of the autoencoder consists of a convolution layer (light green). The last layer (purple) is also a convolution layer and it is used to reconstruct the final image. The numbers under each convolutional layer in the image correspond to the numbers of filters (and, accordingly, to the numbers of feature maps) in the corresponding layers. The number of filters in the last convolution layer depends on the number of channels in the images. The size of the filters in all convolutional layers in the autoencoder was set to 3. The last convolutional layer uses sigmoid activation functions and all other convolutional layers use ReLU[3] (rectified linear unit) activation functions. The bottleneck-values (dark green) are calculated from the convolutional layer in the bottleneck by applying global average pooling (GAP) operation, which is done by taking the mean value in each of the 512 feature maps in the bottleneck. Therefore, the bottleneck-values constitute a vector of size 512. Additionally, in order to facilitate the autoencoder training procedure, we use skip-connections by summing up the outputs of the encoder layers with the outputs of the decoder layers. Skip-connections are those skipping one or more layers and they were originally used for residual networks in [66].

For CARLA images, we've tested a more complex autoencoder with a 16-layer VGG encoder [67]. It has the same structure as the autoencoder used for Pro-SiVIC images, just with more layers in the encoder and decoder parts respectively. Such autoencoder has more parameters, that potentially allows to capture more detailed features of CARLA images. We don't provide an exact figure of this autoencoder here, since it can be inferred from the reference to VGG encoder above. The size of the bottleneck obtained with the images having the corresponding size defined in Section 4.2.2 is 512, which is the same as the bottleneck size of the autoencoder used for Pro-SiVIC images. However, the bigger number of layers allows this autoencoder to learn more complex and meaningful features of the images.

The trained autoencoder is used to calculate anomaly scores. In the reconstruction error approach, anomaly score is usually directly defined by the reconstruction error ([13]). Thus, we use the squared Euclidean distance between the output and the input images of the autoencoder as the anomaly scores in the reconstruction error approach. We calculate it as the sum of squares of the per-pixel values differences. In the bottleneck-values approach, we calculate anomaly scores using One-Class SVM applied on the bottleneck-values, which is a widely used traditional anomaly detection method based on support vector machines [14]. Anomaly detection performance was measured using the calculated anomaly scores with and without normalization. We've noticed that normalization of the anomaly scores doesn't affect the results. However, we provide the results with the normalized anomaly scores for easier comparison of the anomaly scores distributions for the reconstruction error and bottleneck-values approaches.

[3] ReLU is an activation function defined as the positive part of its argument.

Table 3 Autoencoder model training parameters for CARLA dataset

Model ID	Training time	Skip-connections
Model 1	50 epochs	No
Model 2	50 epochs	Yes
Model 3	1000 epochs	No
Model 4	1000 epochs	Yes

Additionally, we test both autoencoder setups on both datasets to analyze how the results are changing based on the model complexity and skip-connections availability. Further in the paper we refer to the autoencoder model with 3 million trainable parameters used for Pro-SiVIC dataset as the simpler model and to the autoencoder model with 22 million trainable parameters used for CARLA dataset as the complex model. We also examine the influence of the training time on the results by alternating between 50 and 1000 epochs for training.

4.4 Experiment operation

4.4.1 Autoencoder training for Pro-SiVIC dataset

For Pro-SiVIC experiments, we have using the simpler autoencoder model described in Sect. 4.3.1. Since we are using the same network architecture for both anomaly detection approaches, it is enough to train the autoencoder once for each type of images. We also use the same autoencoder trained on 256 original training images for anomaly detection on both original and modified images, since the training images are the same for the set of original and modified images. For anomaly detection on greyscaled images, we retrain the autoencoder using 256 greyscaled training images. Therefore, we have been training the autoencoder twice in total. We provide the following training parameters used for both training procedures for reproducibility of our results: `batch size` – 10, `number of epochs` – 1000, `optimizer` – Adadelta, `loss function` – mean squared error (MSE).

After completing the training procedure, anomaly scores are calculated, as described in Sect. 4.3.1, using the reconstruction errors and bottleneck-values of the autoencoder for 100 normal and 31 anomalous testing images. Then, ROC curves are plotted using these anomaly scores and AUROC/AUPRC scores are calculated for each experiment.

4.4.2 Autoencoder training for CARLA dataset

For CARLA experiments, we have been using the complex model described in Sect. 4.3.1. Several training strategies resulting in different final models were tested for CARLA dataset. These strategies are depicted by different training parameters in Table 3. We have been

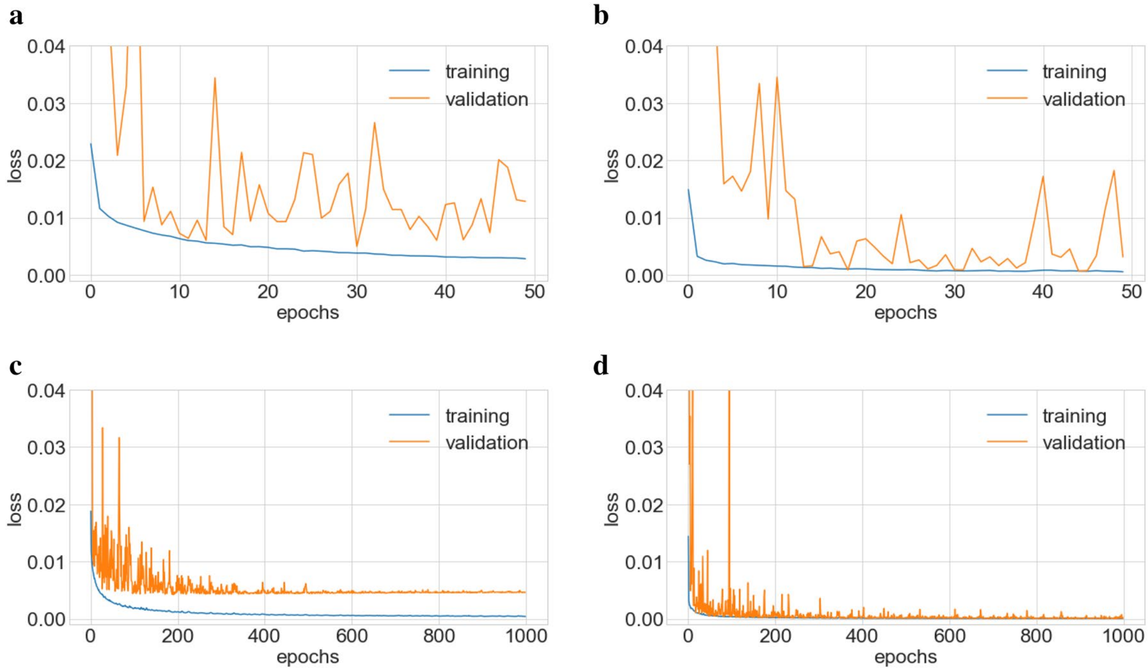

Fig. 5 Autoencoder training results for CARLA dataset: **a** for model 1, **b** for model 2, **c** for model 3, **d** for model 4

Table 4 Stocco's autoencoder models

Model	Architecture
CAE	3 Convolutional layers with 64, 32, and 16 filters of size 3 with 3 max-pooling layers of size 2 in the encoder and 3 convolutional layers with 16, 32, and 64 filters of size 3 with 3 upsampling layers of size 2 in the decoder
SAE	One hidden layer with 64 neurons
DAE	Five fully-connected layers with 256, 128, 64, 128, and 256 neurons
VAE	One intermediate layer with 512 neurons in the encoder, one intermediate layer with 512 neurons in the decoder, and a latent variable of size 2

using different combinations of the training time and skip-connections availability to investigate their influence on the anomaly detection performance.

We additionally include the analysis of the autoencoder training performance in CARLA experiments to further correspond it with the anomaly detection results. Fig. 5 demonstrates the evolution of MSE loss for training and validation subsets during the training. 50 epochs was not enough for the autoencoder training, since the validation losses of model 1 and model 2 have not converged. Moreover, high variations of the validation losses means that the predictions of model 1 and 2 will be random to some extent. Model 2 with skip-connections has lower loss values from the very beginning comparing to loss values of model 1 without skip-connections. Training for 1000 epochs allows to achieve more stable results for models 3 and 4. Further, adding the skip-connections in model 4 facilitates the training and diminishes the gap between the training and validation losses.

4.5 Stocco's autoencoders

For more extensive comparison of anomaly detection performance in addition to the autoencoders described above we use four autoencoders from Stocco et al. work [21], from now on referred to as Stocco's autoencoders. The models and description of their architectures are presented in Table 4.

These autoencoders were trained with Pro-SiVIC and CARLA datasets for 1000 epochs and applied to the same test data as described in Sect. 4.4. Training for 50 epochs was not enough for these models, since losses has not converged as in the case of training our models on CARLA dataset (Fig. 5). An important difference in the training procedure comparing to our settings is that Stocco et al. used batch normalization during the training. We keep batch normalization for training these four autoencoders.

Table 5 AUROC scores calculated on KITTI

Model	Skip-connections	Training time	RE	BV
Simpler	Yes	1000 epochs	0.613	0.471

RE the reconstruction error approach, *BV* the bottleneck-values approach

5 Results

5.1 Pilot study

Table 5 contains the calculated AUROC scores on KITTI dataset using the simpler autoencoder model with skip-connections trained for 1000 epochs. The results for the reconstruction error and bottleneck-values approaches are both around 0.5, which can be an indicator that the autoencoder-based approaches are just randomly predicting anomalous images in KITTI dataset. Therefore, we further perform a visual analysis to understand how the autoencoder actually reconstructs KITTI images.

Figure 6 contains the examples of the input images, reconstructed images with the trained autoencoder, and the reconstruction error images for KITTI dataset in the pilot study. The following observations can be made based on the results of our pilot study.

Firstly, the comparison of Fig. 6c, i indicate that the autoencoder reconstructs the empty road image (normal driving scenario, Fig. 6a) and the image with a silver car (anomalous driving scenario, Fig. 6g) in a similar way, which is not expected according to the principle of the autoencoder-based anomaly detection (Sect. 2). Secondly, the comparison of Fig. 6f, i indicate that the autoencoder reconstructs the image with a red car (Fig. 6d) and the image with a silver car (Fig. 6g) differently. These results have motivated our study on how the autoencoder performs for images with different color characteristics (Pro-SIVIC images). Thirdly, the number of training images is limited by the design of our experiments as the number of images resembling highway driving scenarios with and

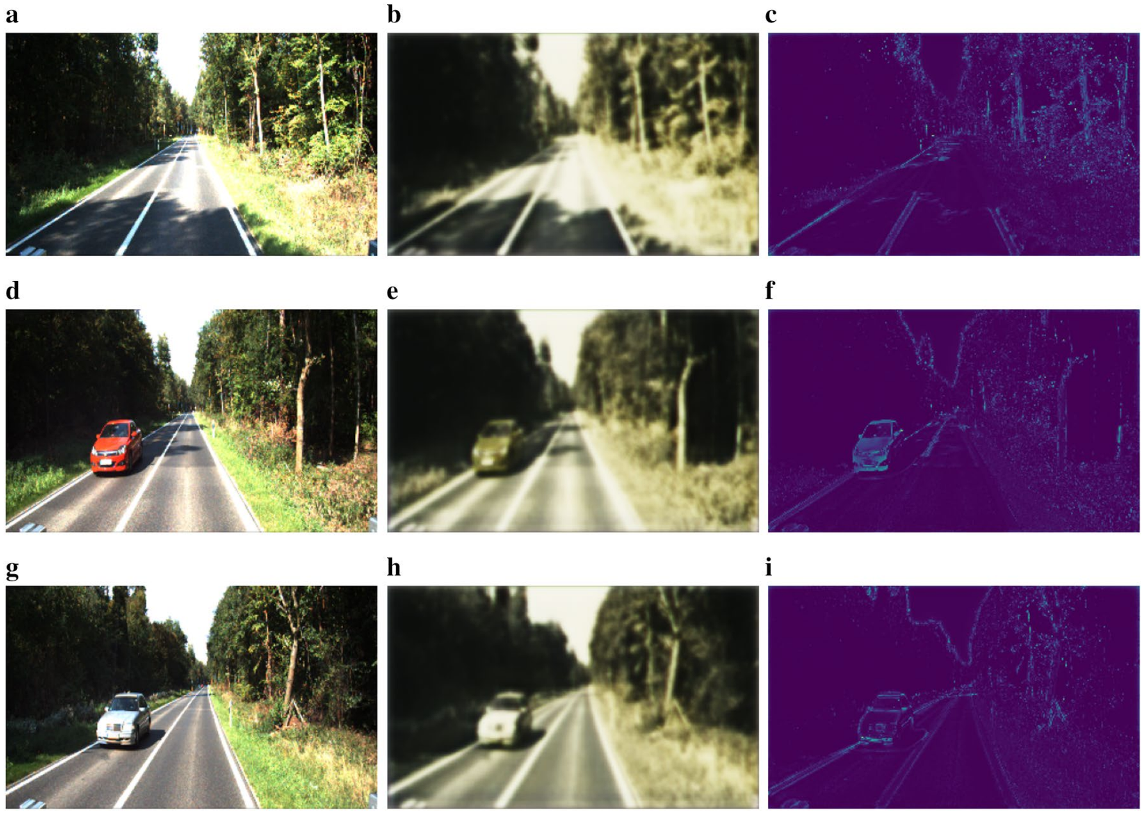

Fig. 6 Autoencoder visual example results for KITTI dataset: **a** input image for the normal driving scenario (empty road), **b** reconstructed image for the normal driving scenario (empty road), **c** reconstruction error image for the normal driving scenario (empty road), **d** input image for the anomalous driving scenario (red car), **e** reconstructed image for the anomalous driving scenario (red car), **f** reconstruction error image for the anomalous driving scenario (red car), **g** input image for the anomalous driving scenario (silver car), **h** reconstructed image for the anomalous driving scenario (silver car), **i** reconstruction error image for the anomalous driving scenario (silver car)

without vehicles in KITTI dataset is limited, which motivated us to generate more images by ourselves. Although this low number of highway-like images in KITTI dataset may have influenced the results of this pilot study, we believe that other factors can have an impact (e.g., colors). Moreover, there are the reconstruction errors due to the complex environments in the real images (such as the errors at the tree crown edges, Fig. 6i). This makes it difficult to analyze the results of semantic anomaly detection. Taking into account all these observations in the pilot study, we have decided to use generated data for further experiments to study the autoencoder's capability for semantic anomaly detection.

5.2 Pro-SiVIC dataset

Firstly, we visualize distributions of anomaly scores for 100 normal and 31 anomalous testing images using the reconstruction errors and bottleneck-values for anomaly detection under different color manipulations (Fig. 7). These distribution plots characterize how well normal and anomalous images can be separated based on their anomaly scores obtained from the autoencoder.

In general, for the case of original (Fig. 7a, b) and modified images (Fig. 7c, d), both approaches produce visually separable distributions of anomaly scores for normal and anomalous images. For greyscaled images (Fig. 7e, f), two distributions overlap each other, which means that it is more difficult to distinguish between normal and anomalous images in these cases.

We perform KS tests to compare the distributions of anomaly scores of normal and anomalous images for all three color cases. The resulted KS test statistics and their p-values are presented in Table 6. High numbers of KS test statistics for the original and modified images in this table indicate that the distributions of anomaly scores of normal and anomalous images are different. Low *p*-values (< 0.001) also show that these differences are statistically significant. This means that there is a high chance that the distributions of anomaly scores of normal and anomalous images should be separable and the performance of anomaly detection is high according to KS tests. It's worth

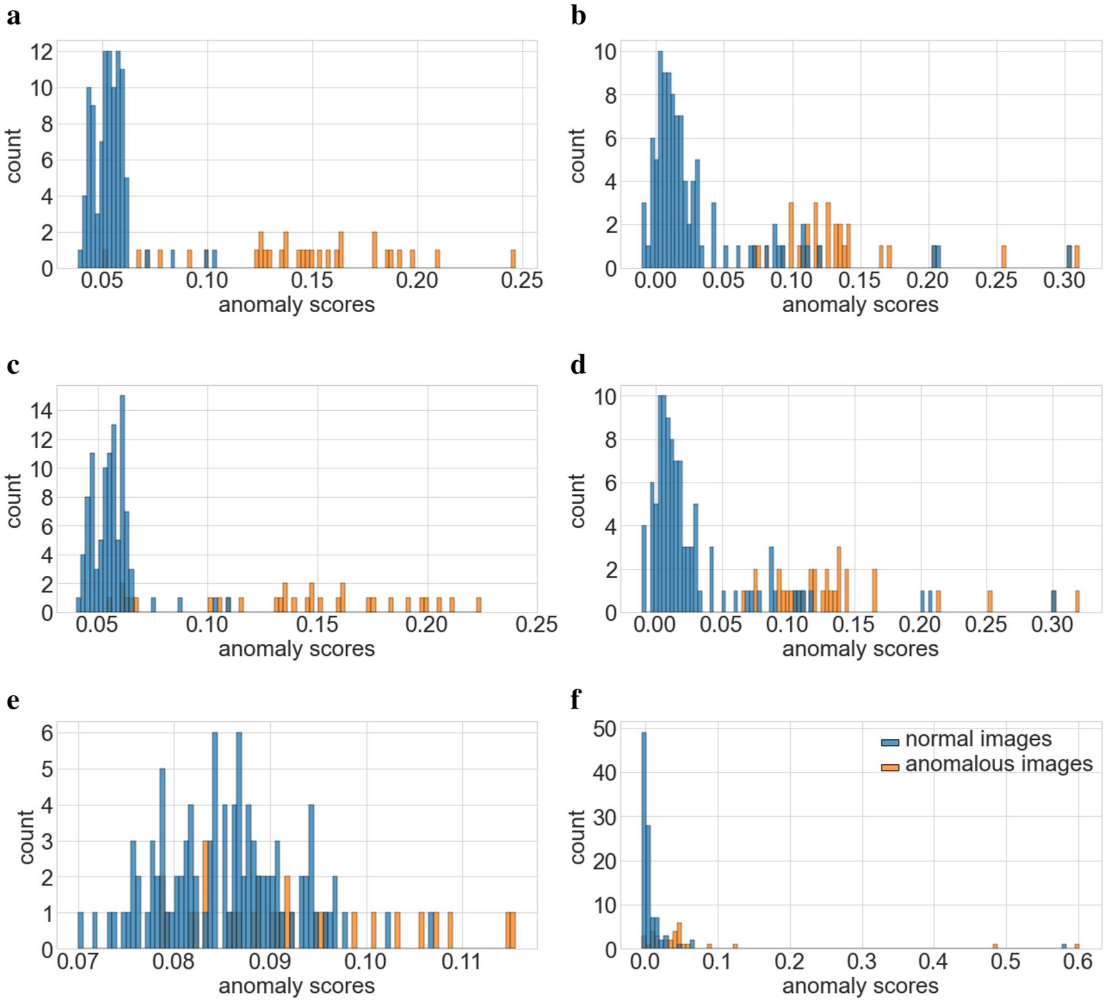

Fig. 7 Distributions of anomaly scores for Pro-SiVIC dataset: **a** for the reconstruction error approach applied on the original images, **b** for the bottleneck-values approach applied on the original images, **c** for the reconstruction error approach applied on the modified images, **d** for the bottleneck-values approach applied on the modified images, **e** for the reconstruction error approach applied on the greyscaled images, **f** for the bottleneck-values approach applied on the greyscaled images

Table 6 KS tests for Pro-SiVIC dataset

	Original images		Modified images		Greyscaled images	
	RE	BV	RE	BV	RE	BV
Statistic	0.93	0.87	0.83	0.86	0.34	0.65
p-value	< 0.001	< 0.001	< 0.001	< 0.001	0.007	< 0.001

RE the reconstruction error approach, *BV* the bottleneck-values approach

to notice that the reconstruction error approach performs better than the bottleneck-values approach according to KS tests for the original images, but worse for the modified images. Whereas, the bottleneck-values approach shows stable results for both the original and modified images. Lower KS tests statistics for the greyscaled images indicate that the anomaly scores distributions of normal and anomalous images are less separable in this case. Moreover, the highest *p*-value (0.007) for the reconstruction error approach in the case of the greyscaled images corresponds to the worst anomaly detection performance in these experiments.

Further, we plot ROC curves and calculate AUROC scores based on the computed anomaly scores. The resulting ROC curves and the corresponding AUROC scores are shown in Fig. 8. ROC curves show FPR/TPR values for different threshold values on the obtained anomaly scores. The choice of the particular threshold

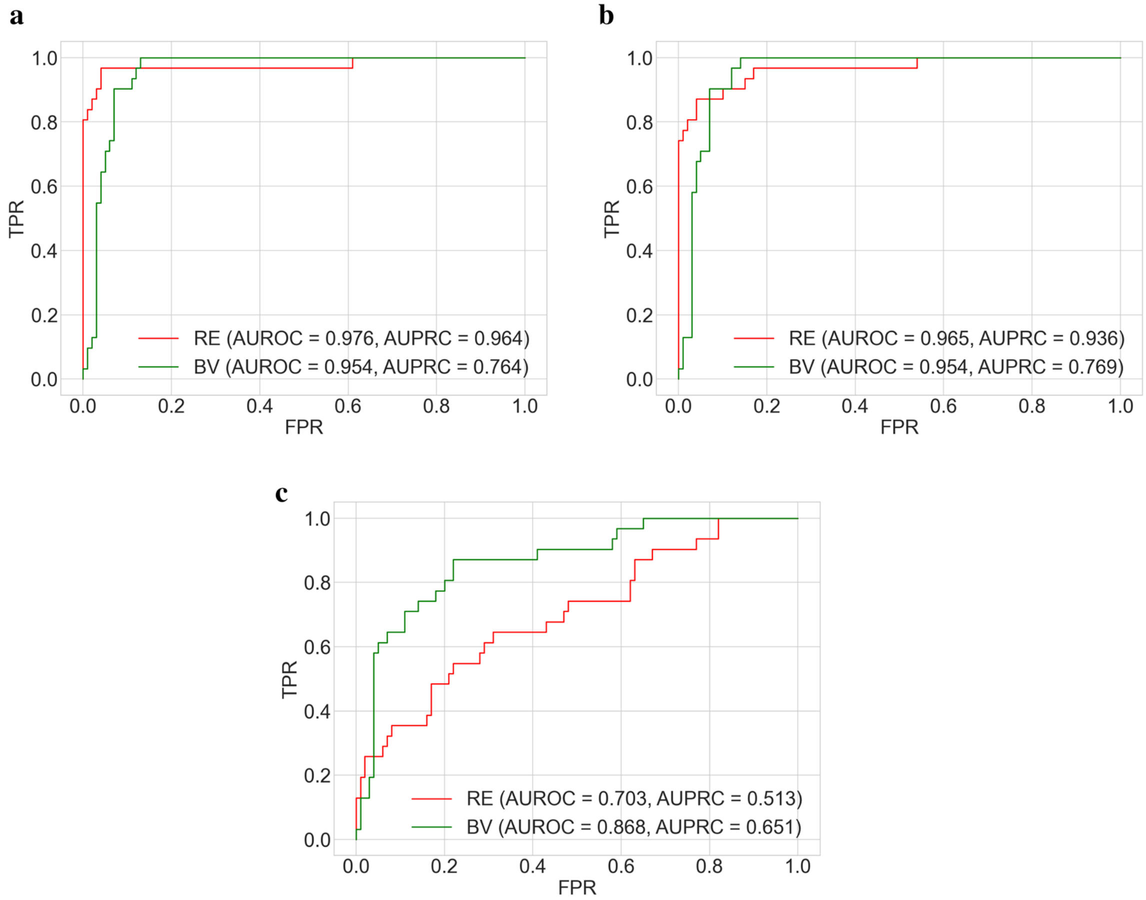

Fig. 8 ROC curves and AUROC, AUPRC scores for Pro-SiVIC dataset (RE—the reconstruction error approach, BV—the bottleneck-values approach): **a** for the original images, **b** for the modified images, **c** for the greyscaled images

value depends on the end goal of the anomaly detection process. From the automotive perspective, we would require an anomaly detection approach always detecting all of the anomalous images while minimizing false positive detections. Therefore, we focus here on FPR values at 100% TPR, which show how many of the normal images are incorrectly classified as anomalous when all of the anomalous images are detected. From Fig. 8, we see that the bottleneck-values approach outperforms the reconstruction error approach by this criteria in all experimental setups. Using FPR values at 100% TPR from the ROC curves in Fig. 8, we can summarize the following results regarding the performance of autoencoder-based anomaly detection *(RQ1)*:

- The reconstruction error approach for anomaly detection has 61%, 52%, and 82% FPR at 100% TPR for the original, modified, and greyscaled images accordingly. This means that when all anomalous images are detected correctly, among all the images detected as anomalous there will be respectively 61%, 52%, and 82% of false positive detections, which are the normal images incorrectly classified as anomalous.
- The bottleneck-values approach for anomaly detection has 13%, 14%, and 65% FPR at 100% TPR for the original, modified, and greyscaled images accordingly. This means that when all anomalous images are detected correctly, among all the images detected as anomalous there will be respectively 13%, 14%, and 65% of false positive detections, which are the normal images incorrectly classified as anomalous.

Since AUROC scores allow us to estimate the overall performance of anomaly detection approaches, then we can estimate their robustness to color changes in the images by comparing their performance in the form of AUROC scores in different experimental settings. The highest performance of anomaly detection in terms of AUROC scores is achieved on the set of original testing images (Fig. 8a). AUROC score of the the reconstruction error approach is slightly better than AUROC score of the bottleneck-values approach in this case (0.976 vs. 0.954). However, in the case of modified images (Fig. 8b), the performance of the reconstruction error approach has dropped down to AUROC score 0.947, while for the bottleneck-values approach it has not decreased. In the case of greyscaled images (Fig. 8c), AUROC scores have decreased for both approaches. However, in this case, the bottleneck-values approach significantly outperforms the the reconstruction error approach in terms of AUROC scores (0.871 vs. 0.727). Therefore, we can formulate the following results from Fig. 8 related to robustness of autoencoder-based approaches for anomaly detection to color changes of the images *(RQ2)*:

- The reconstruction error approach performance of anomaly detection depends on the colors of the anomalous objects (vehicles) while the bottleneck-values approach for anomaly detection does not depend on the colors of the anomalous objects (vehicles).
- When comparing the results for the original and greyscaled images, AUROC score difference is only 0.083 for the bottleneck-approach, whereas for the reconstruction error approach it is 0.249. This indicates that, in general, the bottleneck-values approach is more robust to color changes in the images than the reconstruction error approach.

Figure 8 additionally contains calculated AUPRC scores. The performance differences between the reconstruction error and bottleneck-values approaches for the original and modified images is higher according to AUPRC scores than according to AUROC scores. However, the overall trend in the performance according to AUPRC scores is the same as according to AUROC scores. The reconstruction error approach shows better results for the original and modified images. The bottleneck-values approach is better for the greyscaled images and shows more stable results in general according to both AUROC and AUPRC score.

Selection of the exact threshold value for the anomaly scores results in different TPR and FRP. Acceptable TPR and FPR should be documented in the “safety cage” requirements. Depending on this requirements, it is possible to decrease FPR sacrificing TPR or, vice versa, to increase TRP without preserving FPR by varying the anomaly scores threshold. For example, for the reconstruction error approach in the case of original images the selected threshold value of 0.0514 in Fig. 7a corresponds to 100% TPR and 61% FPR in Fig. 8a, but the selected threshold value of 0.061 in Fig. 7a corresponds to 97% TPR and only 5% FPR in Fig. 8a. It demonstrates, that the threshold value on the anomaly score could be varied to achieve different TPP/FPR values according to the desirable “safety cage” behaviour.

A visual analysis of the autoencoder reconstructions is performed by obtaining the reconstruction error image. For example, let us consider the anomalous image with the highest anomaly score according to the reconstruction error approach when applied to the set of original images (Fig. 9a–c). We can clearly see the area corresponding to the anomalous object (yellow car) in the reconstruction error image (Fig. 9c). However, once we change the color of the car to grey (Fig. 9d–f), the area of the anomalous object in the reconstruction error image (Fig. 9f) becomes less distinct. This is also reflected in the corresponding reconstruction error anomaly score. It has the maximum value of 0.2457 according to the distribution in Fig. 7a for the original image, whereas for the modified image it has a value of 0.1007 that is even less than the mean anomaly score according to the distribution in Fig. 7c. Thus, this example shows how the anomaly score for the same image can be decreased drastically by changing only the color of an anomalous object when using the reconstruction error approach. The same car for the greyscaled image is hardly distinguishable in the reconstruction error image (Fig. 9i). This visually demonstrates that the reconstruction error approach is less capable of detecting semantic anomalies for the greyscaled images in our experiments, which is also supported by the numerical results above about anomaly detection performance.

In contrast to the reconstruction error approach, the bottleneck-values approach for the same original and modified images outputs the similar anomaly scores of 0.1713 and 0.1662 accordingly, which correspond to the same areas of the distributions in Fig. 7b, d. This demonstrates how changing the color of the car does not affect the results of anomaly detection in driving scenario images for the bottleneck-values approach, but it does affect anomaly detection results for the reconstruction error approach.

5.3 CARLA dataset

For CARLA dataset the same measurements were performed, but comparing the results for different autoencoders trained with different training parameters. The obtained anomaly scores distributions for the reconstruction error and bottleneck values approaches are shown in Fig. 10. Through visual assessment of these distributions, it can be already seen that the anomaly scores of the normal and anomalous images are less distinct for CARLA dataset than for Pro-SiVIC dataset. Moreover, there are some oppositely wrong anomaly scores. For example, models 1 and 3 without skip-connections for the bottleneck-values approach (Fig. 10c, f) provides higher anomaly scores for the normal images and lower anomaly scores for the anomalous images. The most visually separable

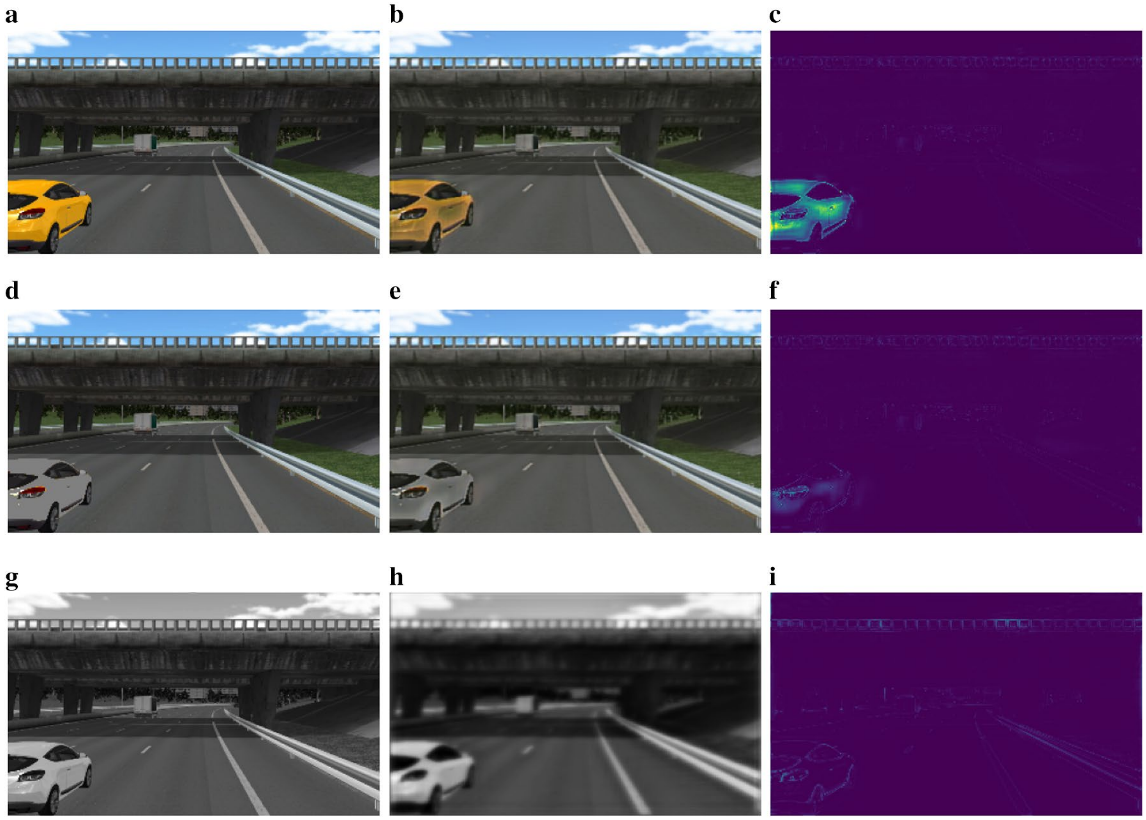

Fig. 9 Autoencoder visual example results for Pro-SiVIC dataset: **a** original input image, **b** original reconstructed image, **c** original reconstruction error image, **d** modified input image, **e** modified reconstructed image, **f** modified reconstruction error image, **g** greyscaled input image, **h** greyscaled reconstructed image, **i** greyscaled reconstruction error image

distributions of anomaly scores are obtained for the reconstruction error approach for model 4 (Fig. 10g).

KS test results for the anomaly scores distributions for CARLA dataset are shown in Table 7. They support the findings made through the visual assessment of the distributions in Fig. 10. According to the table the highest difference between the distributions of the anomaly scores of the normal and anomalous images corresponds to the reconstruction error approach for model 4. However, for the cases, where the predicted anomaly scores are oppositely wrong (such as the bottleneck-values approach for model 1 and the bottleneck-values approach for model 3), it's not really correct to rely on KS test results. The test shows the relatively high differences in the distributions of the anomaly scores, but the actual anomaly detection does not work at all in these cases.

ROC curves and the corresponding AUROC and AUPRC scores for the experiments with CARLA dataset are shown in Fig. 11. There is clearly the only one best ROC curve corresponding to the reconstruction error approach for model 4 (Fig. 11d). This curve shows that the reconstruction error approach achieves the lowest 8% FPR at 100% TPR for model 4. Anomaly detection performance according to AUROC and AUPRC scores is also the best in this case. It has slightly lower anomaly detection performance in terms of AUROC and AUPRC scores for models 2 than for model 4. The reconstruction error approach has the worst performance in terms of AUROC and AUPRC scores for model 3. The bottleneck-values approach has also the best performance in terms of AUROC and AUPRC scores for model 4. It achieves its lowest anomaly detection performance for model 1. Based on these results, the following conclusions about autoencoder-based anomaly detection performance under different training parameters can be made *(RQ1)*:

- Adding the skip-connections to the autoencoder significantly improves the performance of anomaly detection for the reconstruction error approach. When comparing the results for models 4 and 3, AUROC score is increased by 0.661 and AUPRC score is increased by 0.468. The autoencoder learns better to reconstruct the normal images in this case.
- Adding the skip-connections to the autoencoder significantly improves the performance of anomaly detection for the bottleneck-values approach. When com-

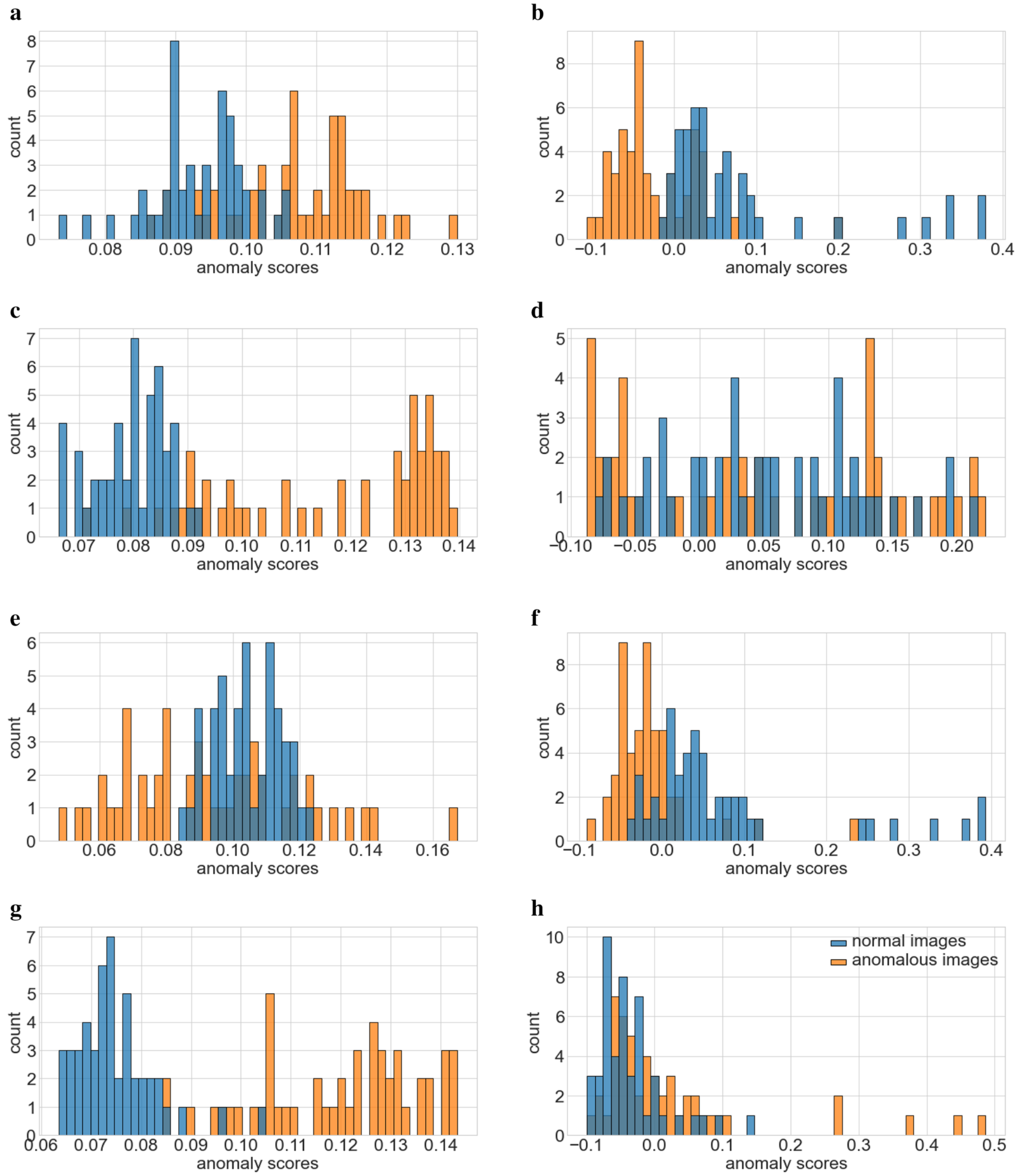

Fig. 10 Distributions of anomaly scores for CARLA dataset: **a** for the reconstruction error approach for model 1, **b** for the bottleneck-values approach for model 1, **c** for the reconstruction error approach for model 2, **d** for the bottleneck-values approach for model 2, **e** for the reconstruction error approach for model 3, **f** for the bottleneck-values approach for model 3, **g** for the reconstruction error approach for model 4, **h** for the bottleneck-values approach for model 4

Table 7 KS tests for CARLA dataset

	Model 1		Model 2		Model 3		Model 4	
	RE	BV	RE	BV	RE	BV	RE	BV
Statistic	0.70	0.68	0.89	0.26	0.44	0.71	0.92	0.31
p-value	< 0.001	< 0.001	< 0.0010.063		< 0.001	< 0.001	< 0.0010.012	

RE the reconstruction error approach, *BV* the bottleneck-values approach

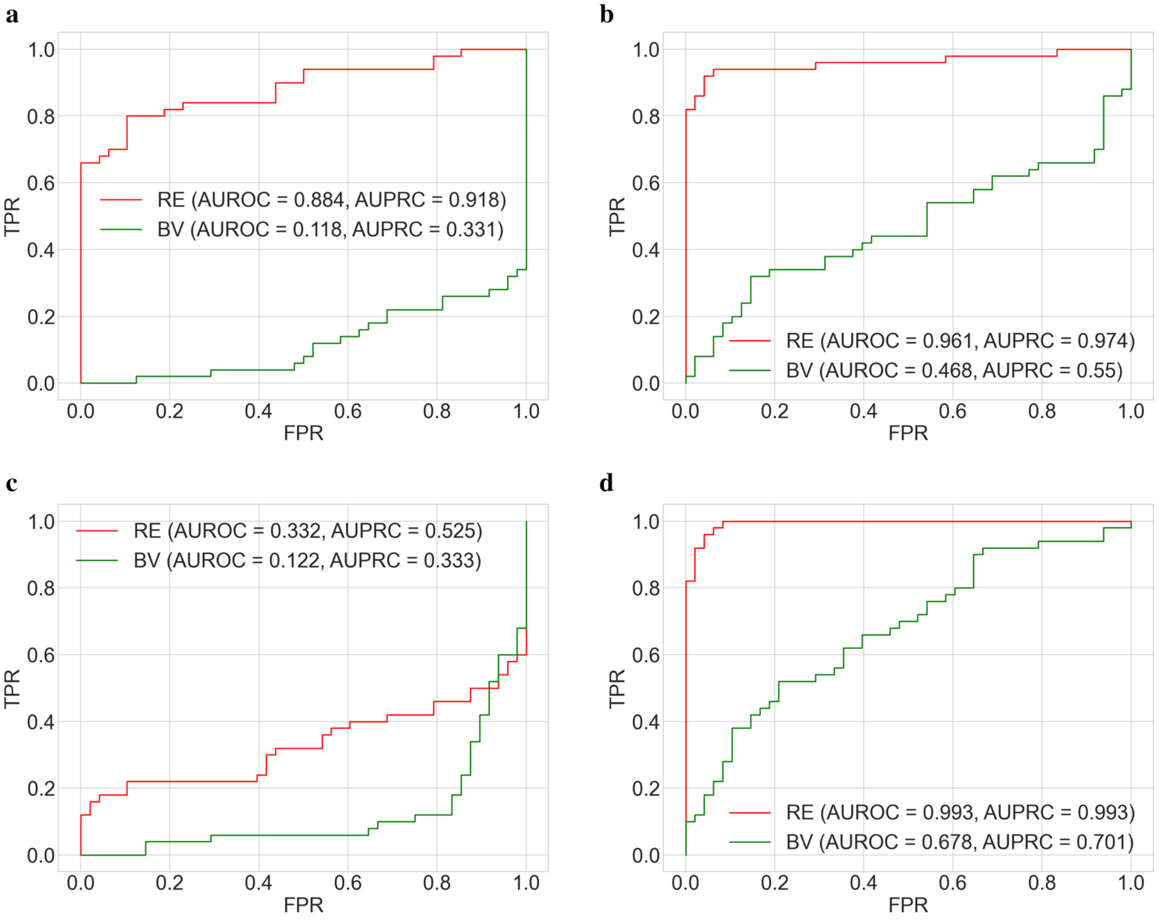

Fig. 11 ROC curves and AUROC, AUPRC scores for CARLA dataset (RE—the reconstruction error approach, BV—the bottleneck-values approach): **a** for model 1, **b** for model 2, **c** for model 3, **d** for model 4

paring the results for models 4 and 3, AUROC score is increased by 0.556 and AUPRC score is increased by 0.368. The autoencoder learns better representative features of the images in the bottleneck due to using the skip-connections in this case.

- Training the autoencoder for more epochs has a positive influence on the anomaly detection performance for both the reconstruction error and bottleneck-values approaches.
- In general, for CARLA dataset the bottleneck-values approach has not shown any good performance comparable with Pro-SiVIC experiments. This indicates that the autoencoder was not able to learn useful features of the compressed representations of the images due to the dataset complexity. We should further test different autoencoder's architectures and training parameters in order to understand if the bottleneck-values approach works for more complex datasets other than Pro-SiVIC dataset.

Figure 12 demonstrates the examples of the autoencoder's reconstructions for CARLA dataset when using model 4. In general, the reconstruction errors are lower than in Pro-SiVIC experiments even though the environmental features are more complex here. This is mostly due to the bigger autoencoder model used for CARLA dataset. The reconstruction error of the red car in Fig. 12c is more prominent than the reconstruction error of the grey car in Fig. 12f. It is probably related to the results obtained in the color experiments with Pro-SiVIC dataset. Figure 12i shows the reconstruction error image with the grey car and the cyclist as semantic anomalies. It is noteworthy that the highest reconstruction errors in this image correspond to the part of the car's rear light and the cyclist.

5.4 Summary and discussion

We have analyzed the performance of the autoencoder-based anomaly detection for two generated datasets of the highway driving scenario images. In each experiment,

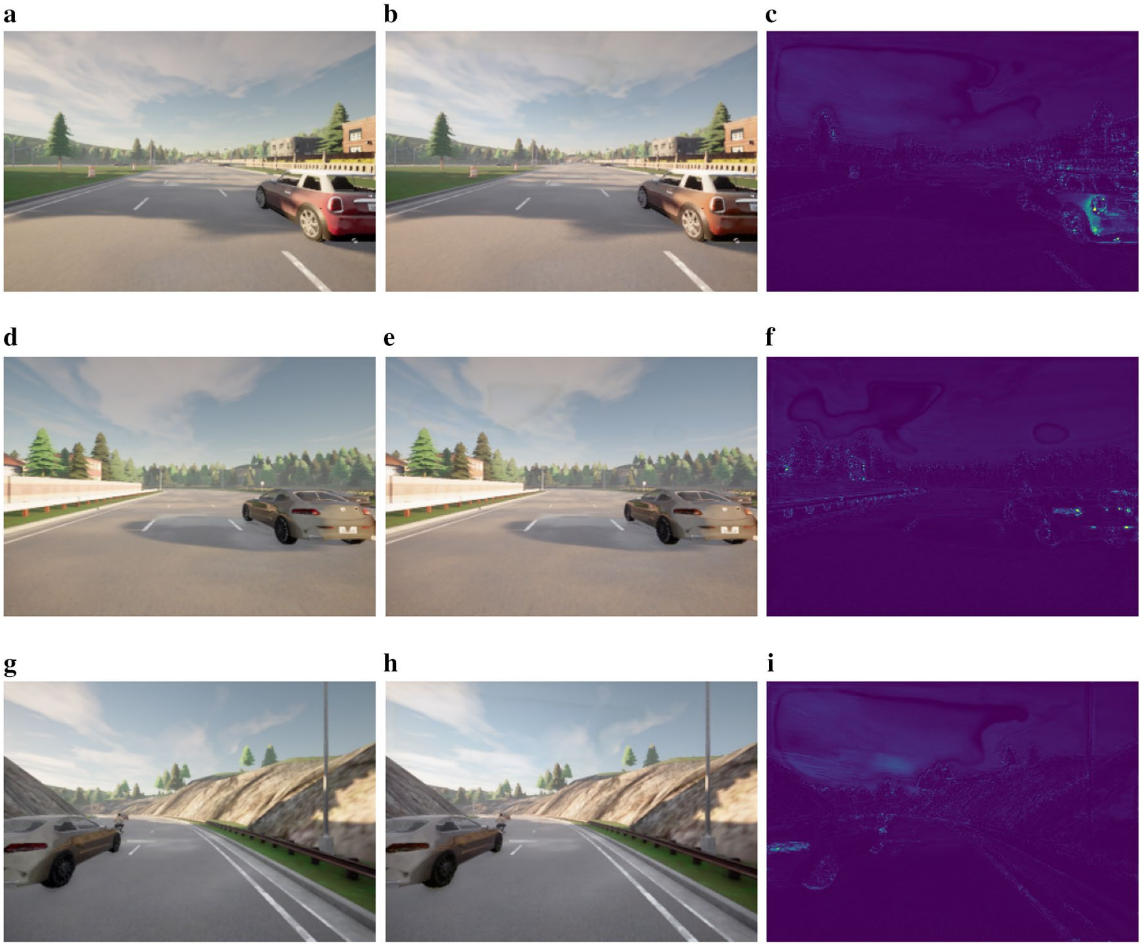

Fig. 12 Autoencoder visual example results for CARLA dataset: **a** input image for the anomalous driving scenario (red car), **b** reconstructed image for the anomalous driving scenario (red car), **c** reconstruction error image for the anomalous driving scenario (red car), **d** input image for the anomalous driving scenario (grey car), **e** reconstructed image for the anomalous driving scenario (grey car), **f** reconstruction error image for the anomalous driving scenario (grey car), **g** input image for the anomalous driving scenario (grey car and cyclist), **h** reconstructed image for the anomalous driving scenario (grey car and cyclist), **i** reconstruction error image for the anomalous driving scenario (grey car and cyclist)

we have calculated anomaly scores with both the reconstruction error and bottleneck-values approaches and compared their performance. The same performance measures were used throughout all the experiments.

With Pro-SiVIC dataset, the influence of the color characteristics on the autoencoder-based anomaly detection performance was estimated. Table 8 summarizes the calculated AUROC scores for the color experiments. The results show that the overall color scheme of the images as well as the colors of the anomalous objects may have an impact on the autoencoder-based anomaly detection performance. According to AUROC scores, the bottleneck-values values approach is less sensitive to color changes in the images than the reconstruction error approach.

Table 9 contains the calculated AUROC scores on Pro-SiVIC and CARLA datasets for all the autoencoder setups (the simpler and complex models with and without skip-connections) trained for both 50 and 1000 epochs. The base simpler model with skip-connections trained for 1000 epochs performs the best for anomaly detection on Pro-SiVIC dataset in terms of AUROC scores for both the reconstruction error and bottleneck-values approaches. Whereas on CARLA dataset, the best setup, also for both approaches, is the complex model with skip-connections trained for 1000 epochs. For all the models that are trained for 1000 epochs, except the simpler model without skip-connections on Pro-SiVIC dataset, the bottleneck-values approach is worse than the reconstruction error approach. When trained for 50 epochs the models were not well trained as it was shown by the examples of the training results in Sect. 4.4.2, therefore the presented anomaly detection results may be not so suitable for the analysis. However, it is noticeable that the training duration positively affects the results for the reconstruction error

Table 8 AUROC scores calculated on Pro-SiVIC for different color cases

			Original		Modified		Greyscaled	
Model	Skip-connections	Training time	RE	BV	RE	BV	RE	BV
Simpler	Yes	1000 epochs	0.976	0.954	0.965	0.954	0.703	0.868

RE the reconstruction error approach, *BV* the bottleneck-values approach

Table 9 AUROC scores calculated on Pro-SiVIC and CARLA datasets for different autoencoder setups and training duration (values in bold correspond to the best result for each setup using different anomaly detection approaches applied on different datasets)

			Pro-SiVIC		CARLA	
Model	Skip-connections	Training time	RE	BV	RE	BV
Simpler	Yes	50 epochs	0.405	0.801	0.878	0.518
		1000 epochs	**0.976**	**0.954**	0.525	0.414
	No	50 epochs	0.386	0.849	0.906	0.528
		1000 epochs	0.808	0.878	0.929	0.582
Complex	Yes	50 epochs	0.734	0.672	0.961	0.468
		1000 epochs	0.898	0.709	**0.993**	**0.678**
	no	50 epochs	0.716	0.716	0.884	0.118
		1000 epochs	0.902	0.696	0.332	0.122

RE the reconstruction error approach, *BV* the bottleneck-values approach

approach on Pro-SiVIC dataset. The bottleneck-values approach on Pro-SiVIC dataset demonstrates better performance for simpler model than for complex model. The simpler model for CARLA performs always better without skip-connections for both approaches.

Table 10 demonstrates the additional comparison of anomaly detection performance of our best performing autoencoders from the previous experiments with Stocco's autoencoders. The best performing autoencoder models according to the calculated AUROC scores from our experiments are simpler and complex models for Pro-SiVIC and CARLA datasets respectfully both with skip-connections and trained for 1000 epochs.

Table 10 AUROC scores calculated on Pro-SiVIC and CARLA datasets with the best autoencoder models from our experiments and four Stocco's autoencoders

	Pro-SiVIC		CARLA	
model	RE	BV	RE	BV
Best of ours	0.976	0.954	0.993	0.678
Stocco's CAE	0.813	0.907	0.903	0.624
Stocco's SAE	0.820	0.543	0.046	0.490
Stocco's DAE	0.897	0.702	0.165	0.052
Stocco's VAE	0.845	0.560	0.202	0.199

RE the reconstruction error approach, *BV* the bottleneck-values approach

We can make the following summary from Table 10:

- None of the Stocco's autoencoders could show better results than the best of our models.
- For Pro-SiVIC dataset, Stocco's CAE has worse reconstruction error approach performance than the other non-convolutional models. On the contrary, Stocco's CAE has better bottleneck-values approach performance among other Stocco's autoencoders, which can indicate that convolutions help to learn useful features of the images in the autoencoder's bottleneck.
- For CARLA dataset, only CAE out of other Stocco's models demonstrates reasonable anomaly detection performance for both reconstruction error and bottleneck-values approaches.

To summarise, all the obtained results in this study indicate that the autoencoder as a technology for semantic anomaly detection needs to be tested further in different settings. Unlike in Stocco et al. [21], where the autoencoders are applied for context anomaly detection in an online fashion, we use the autoencoders for semantic anomaly detection in an offline fashion since they have not been investigated enough for this task before applying them in an online fashion. Moreover, in order to use the autoencoders for real driving scenarios, we need to have more variability in the real images. It is important to understand how the autoencoders will work for semantic anomaly detection in real settings under different environmental conditions which correspond to different data distributions.

6 Threats to validity

We notice and discuss threats to validity of our study, which we've found are the most important. The classification scheme by Wohlin et al. [68] was used to classify the observed threats according to four different types: conclusion, internal, construct, and external validity.

Conclusion validity is concerned with the relationship between our experiments with the autoencoder-based anomaly detection and the obtained results. Firstly, there is a risk that the autoencoders are not properly trained or overfitted due to our experimental setup. To address this we monitor the training and validation losses during the autoencoders' training. On the example of the experiments with CARLA dataset, we show that for 1000 epochs the training converges with no signs of overfitting. Secondly, the statistical power of the experiments is partially assured by using KS test and its statistical significance. However, we admit that the size of the datasets should be increased in the future studies to further address this threat.

Internal validity is dealing with the uncontrolled influence on the experiments. To mitigate the threats to the internal validity we control the objects and the independent variables in our experiments. We generate the images by ourselves, so we know what they contain and we are sure that the normal and anomalous images are defined correctly before the experiments. Moreover, the generated images have the same landscape and environmental conditions. Thus, we make sure that the normal and anomalous images are different only by the presence of semantic anomalies. We randomly divide the datasets into the training and testing sets and use the same images for training and testing across the experiments to address the selection threat. Also, we change only one independent variable (colors of the images for Pro-SiVIC dataset and training parameters for CARLA dataset) at a time to control the changes in the results. By visually assessing the reconstruction error images, we test whether the autoencoders really detect the semantic anomalies, and there are not other factors that affect the calculated anomaly scores.

Construct validity relates to the generalizability of the experiments' results with respect to the RQ. There is a risk that the construct of "capability" in the RQ is not sufficiently defined. We measure the differences in the anomaly scores distributions (KS tests) and consider anomaly detection as a binary classification task with the corresponding metrics (AUROC and AUPRC scores) to estimate the autoencoder's capability to detect semantic anomalies. Perhaps, there are more ways to define the capability, but the measures used in this study were chosen based on the objectiveness and the absence of the confounding factors. By using several different measures we mitigate the risk of the measurement bias. At the same time, we admit our experiments' restricted generalizability to other types of highway driving scenarios different from what we've generated. Therefore, the variability in the landscape and weather conditions in the data is required further. Still, the study generalizes to the highway driving scenarios with the simulators' landscapes and daytime sunny weather conditions.

External validity is dealing with the generalization ability of the obtained results to the industrial practice. Experiments with the generated data are the first steps towards the understanding of the autoencoders' capability in detecting semantic anomalies in the highway driving scenarios, but for full generalizability, testing on the real data is required further. Still, testing the autoencoders for semantic anomaly detection on the generated data allows generalizing the results to some extent.

7 Conclusion

This work provides a comparison between two autoencoder-based approaches for anomaly detection in terms of their performance and robustness to color changes in driving scenario images towards answering the general research question (*RQ: How capable are autoencoders in detecting semantic anomalies in highway driving scenarios?*). According to our experiments, the bottleneck-values approach has demonstrated better results than the reconstruction error approach in terms of anomaly detection performance and robustness to color changes in the images for the simpler dataset. However, for the more complex dataset the only working solution was the reconstruction error approach when using the skip-connections.

We have addressed anomaly detection performance in *RQ1 (What is the performance of the autoencoder-based approaches for anomaly detection in driving scenario images?)*. On the example of generated Pro-SiVIC images, the reconstruction error approach had slightly higher AUROC scores than the bottleneck-values approach in case of original and modified images, but the latter approach had lower FPR values at 100% TPR in all cases. It shows that the bottleneck-values approach for anomaly detection is more suitable for the application in safety mechanisms, for example in "safety cages". Lower FPR values at 100% TPR mean that a "safety cage" will always switch to the safe mode when it is necessary while reducing the number of times switching to the safe mode when it is not required. For example, if switching to the safe mode requests the human driver to take-over a control, then the "safety cage" with an anomaly detector with lower FRP will ask the driver for this request less times when it's not necessary. We have also investigated the influence of the training parameters on the autoencoder-based anomaly detection performance when using CARLA dataset with more complex environments compared to Pro-SiVIC dataset. We have found that for complex model the reconstruction error approach has been working only if the skip-connections were used in the autoencoder. The skip-connections help to train the autoencoder better and, as a consequence, they improve the performance of anomaly detection for the reconstruction error approach. At the same time, we have observed that the bottleneck-values approach has not shown any good performance for CARLA dataset. Perhaps, the autoencoder was not able to properly learn compressed representations of more complex CARLA images and more advanced architectures should be experimented with, which is a future study proposal. It is important to understand this limitation of the autoencoder-based anomaly detection approach before applying it in practice.

We have analyzed anomaly detection robustness to both color changes of anomalous objects and color changes of images in general in *RQ2 (What is the robustness of the autoencoder-based anomaly detection approaches to color changes in driving scenario images?)* on the example of generated Pro-SiVIC images. On the whole, the bottleneck-values approach has been shown to be more robust to color changes in driving scenario images than the reconstruction error approach. Firstly, in case of modified images, anomaly detection performance of the reconstruction error approach decreased when the color of some of the anomalous objects was changed, while it was not the case for the bottleneck-values approach. This indicates that the bottleneck-values approach can be insensitive to color changes of the anomalous objects, which is a good quality of an anomaly detection approach. Thus, the bottleneck-values approach is less depended on the color information of the anomalous objects that could be of different colors in reality, but it rather relies on their other important characteristics, which make them distinguishable. Secondly, in case of greyscaled images, anomaly detection performance decreased for both approaches, though it decreased less for the bottleneck-values approach than for the reconstruction error approach. Therefore, both approaches depend on the color information presented in the images in general, but to a different extent. This result should be taken into account in the automotive domain. It means that any change in visual conditions in the environment or the camera, which affects the color characteristics of the images, can also affect the autoencoder-based anomaly detection performance. Therefore, this can affect the overall safety of a car as a consequence.

We have considered the case of the autoencoder-based anomaly detection in driving scenario images, which can be used as a reference for the future research in this area. While the generated data provided flexibility in our experiments, previous research studies have shown that anomaly detection performance is consistently lower for real images than for simulated images. Therefore, the autoencoder-based anomaly detection with real environment images must be tested before reaching the production stage in the automotive industry. We are currently setting up the experiments on the two real datasets: Volvo Highway Dataset[4] by Volvo Group (accessed through AI Sweden) and Cirrus[5] dataset by Volvo Cars. Both datasets have annotations, which makes it possible to define semantic anomalies in the images. Next, different types of anomalous objects (such as pedestrians, animals, etc.) can be considered to cover more driving scenarios in the future. Also, while in this study we've been focusing on the most general sunny daytime highway road case, the considered approaches for semantic anomaly detection should be tested further in different contexts, e. g. by varying weather, time, or landscape conditions. Furthermore, the comparison with Stocco's autoencoders shows that the architecture and complexity of the models influence the anomaly detection performance. In order to explore the possibility of improving anomaly detection performance, different autoencoder architectures should be experimented with, especially deeper autoencoders with a higher number of layers, which may learn more complicated features in the images. Additionally, to complete the comparison of autoencoder-based approaches for anomaly detection, different methods for anomaly scores calculation in the reconstruction error and the bottleneck-values approaches should be examined. Besides, it's worth to compare the autoencoder-based approach for semantic anomaly detection to other existing approaches, e. g. GANs. Finally, autoencoder-based approaches could be applied for sequences of images, which may improve anomaly detection results by taking into account time dependence. These will be the steps towards more thorough research in the area of autoencoder-based anomaly detection with the extended scope of the use-cases in driving scenario context.

[4] https://www.ai.se/en/data-factory-datasets/volvo-highway-dataset.

[5] https://developer.volvocars.com/open-datasets/cirrus.

Funding Open access funding provided by University of Gothenburg. This study was funded by Vinnova,FFI, Fordonsstrategisk forskning och innovation (Grant Number 2018-02725).

Data availability The data that support the findings of this study are available from Volvo Car Corporation. However, restrictions apply to the availability of Pro-SiVIC dataset, which is not publicly available. This dataset is available from the authors upon reasonable request and with permission of Volvo Car Corporation. CARLA dataset was published openly along with the submitted paper.

Declarations

Conflict of interest The authors declare that they have no conflict of interest.

Ethical approval No ethical approval is required for this study.

References

1. Staron M (2021) Current trends in automotive software architectures. In: Automotive software architectures, Springer, pp 259–268
2. Singh KB, Arat MA (2019) Deep learning in the automotive industry: recent advances and application examples, arXiv:1906.08834
3. Borg M, Englund C, Wnuk K, Duran B, Levandowski C, Gao S, Tan Y, Kaijser H, Lönn H, Törnqvist J (2019) Safely entering the deep: a review of verification and validation for machine learning and a challenge elicitation in the automotive industry. J Automot Softw Eng 1:1–19
4. Kläs M, Vollmer AM (2018) Uncertainty in machine learning applications: a practice-driven classification of uncertainty. In: Gallina B, Skavhaug A, Schoitsch E, Bitsch F (eds) Computer safety, reliability, and security. Springer International Publishing, Cham, pp 431–438
5. Rumelhart DE, Hinton GE, Williams RJ (1986) Learning representations by back-propagating errors. Nature 323(6088):533–536
6. Masci J, Meier U, Cireşan D, Schmidhuber J (2011) Stacked convolutional auto-encoders for hierarchical feature extraction. In: international conference on artificial neural networks. Springer, pp 52–59
7. Polycarpou M, Xiaodong Z, Xu R, Yanli Y, Chiman K (2004) A neural network based approach to adaptive fault tolerant flight control. In: proceedings of the 2004 IEEE International Symposium on Intelligent Control, 2004, pp 61–66
8. Henriksson J, Berger C, Borg M, Tornberg L, Englund C, Sathyamoorthy SR, Ursing S (2019) Towards structured evaluation of deep neural network supervisors
9. Kratz E (2019) Novel scenario detection in road traffic images. Master's thesis, Chalmers University of Technology, Department of Electrical Engineering
10. Chalapathy R, Chawla S (2019) Deep learning for anomaly detection: a survey
11. Goodfellow I, Bengio Y, Courville A (2016) Deep Learning. MIT Press. http://www.deeplearningbook.org
12. Kingma DP, Welling M (2013) Auto-encoding variational bayes, arXiv preprint arXiv:1312.6114
13. Ruff L, Kauffmann JR, Vandermeulen RA, Montavon G, Samek W, Kloft M, Dietterich TG, Müller K-R (2021) A unifying review of deep and shallow anomaly detection. In: Proceedings of the IEEE
14. Erfani SM, Rajasegarar S, Karunasekera S, Leckie C (2016) High-dimensional and large-scale anomaly detection using a linear one-class svm with deep learning. Pattern Recognit 58:121–134
15. Amarbayasgalan T, Jargalsaikhan B, Ryu KH (2018) Unsupervised novelty detection using deep autoencoders with density based clustering. Appl Sci 8(9):1468
16. Sarafijanovic-Djukic N, Davis J (2019) Fast distance-based anomaly detection in images using an inception-like autoencoder. In: Kralj Novak P, Šmuc T, Džeroski S (eds) Discovery science, Springer International Publishing, Cham, pp 493–508
17. Wang S, Huang L, Ge J, Zhang T, Feng H, Li M, Zhang H, Ng V (2020) Synergy between machine/deep learning and software engineering: How far are we?
18. Zhang M, Zhang Y, Zhang L, Liu C, Khurshid S (2018) Deeproad: gan-based metamorphic testing and input validation framework for autonomous driving systems. In: 2018 33rd IEEE/ACM international conference on automated software engineering (ASE), pp 132–142
19. Koopman P, Wagner M (2017) Autonomous vehicle safety: an interdisciplinary challenge. IEEE Intell Transp Syst Mag 9(1):90–96
20. Koopman P, Wagner M (2016) Challenges in autonomous vehicle testing and validation. SAE Int J Trans Saf 4(04):15–24
21. Stocco A, Weiss M, Calzana M, Tonella P (2020) Misbehaviour prediction for autonomous driving systems. In: 2020 IEEE/ACM 42nd international conference on software engineering (ICSE), pp 359–371
22. Stocco A, Tonella P (2020) Towards anomaly detectors that learn continuously. In: 2020 IEEE international symposium on software reliability engineering workshops (ISSREW), pp 201–208
23. Chalapathy R, Menon AK, Chawla S (2018) Anomaly detection using one-class neural networks, arXiv preprint arXiv:1802.06360
24. Mohseni S, Pitale M, Yadawa J, Wang Z (2020) Self-supervised learning for generalizable out-of-distribution detection. In: proceedings of the AAAI conference on artificial intelligence, vol 34, pp 5216–5223
25. Liang S, Li Y, Srikant R (2017) Enhancing the reliability of out-of-distribution image detection in neural networks, arXiv preprint arXiv:1706.02690
26. DeVries T, Taylor GW (2018) Learning confidence for out-of-distribution detection in neural networks, arXiv preprint arXiv:1802.04865
27. Bergmann P, Fauser M, Sattlegger D, Steger C (2020) Uninformed students: student-teacher anomaly detection with discriminative latent embeddings. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition, pp 4183–4192
28. Rippel O, Mertens P, Merhof D (2020) Modeling the distribution of normal data in pre-trained deep features for anomaly detection, arXiv preprint arXiv:2005.14140

29. Perera P, Patel VM (2019) Learning deep features for one-class classification. IEEE Trans Image Process 28(11):5450–5463
30. Masana M, Ruiz I, Serrat J, van de Weijer J, Lopez AM (2018) Metric learning for novelty and anomaly detection, arXiv preprint arXiv:1808.05492
31. Ali R, Muk K, Kyung CM (2020) Self-supervised representation learning for visual anomaly detection, arXiv preprint arXiv:2006.09654
32. Blum H, Sarlin P-E, Nieto J, Siegwart R, Cadena C (2019) Fishyscapes: a benchmark for safe semantic segmentation in autonomous driving. In: proceedings of the IEEE/CVF international conference on computer vision workshops, pp 0
33. Lis K, Honari S, Fua P, Salzmann M (2020) Detecting road obstacles by erasing them, arXiv preprint arXiv:2012.13633
34. Jourdan N, Rehder E, Franke U (2020) Identification of uncertainty in artificial neural networks. In: proceedings of the 13th Uni-DAS eV workshop Fahrerassistenz und automatisiertes Fahren, vol 2
35. Bevandić P, Krešo I, Oršić M, Šegvić S (2018)Discriminative out-of-distribution detection for semantic segmentation, arXiv preprint arXiv:1808.07703
36. Angus M, . Czarnecki K, Salay R (2019) Efficacy of pixel-level ood detection for semantic segmentation, arXiv preprint arXiv:1911.02897
37. Brüggemann D, Chan R, Rottmann M, Gottschalk H, Bracke S (2020) Detecting out of distribution objects in semantic segmentation of street scenes. In: the 30th European safety and reliability conference (ESREL), vol 2
38. Golan I, El-Yaniv R (2018) Deep anomaly detection using geometric transformations, arXiv preprint arXiv:1805.10917
39. Ran X, Xu M, Mei L, Xu Q, Liu Q (2020) "Detecting out-of-distribution samples via variational auto-encoder with reliable uncertainty estimation, arXiv preprint arXiv:2007.08128
40. Ruff L, Vandermeulen R, Goernitz N, Deecke L, Siddiqui SA, Binder A, Müller E, Kloft M (2018) Deep one-class classification. In: international conference on machine learning, PMLR, pp 4393–4402
41. Cohen N, Hoshen Y (2020) Sub-image anomaly detection with deep pyramid correspondences, arXiv preprint arXiv:2005.02357
42. Zaheer MZ, Lee J.-h, Astrid M, Lee S-I (2020) Old is gold: redefining the adversarially learned one-class classifier training paradigm. In: proceedings of the IEEE/CVF conference on computer vision and pattern recognition, pp 14183–14193
43. Che T, Liu X, Li S, Ge Y, Zhang R, Xiong C, Bengio Y (2019) Deep verifier networks: verification of deep discriminative models with deep generative models, arXiv preprint arXiv:1911.07421
44. Lee K, Lee H, Lee K, Shin J (2017) Training confidence-calibrated classifiers for detecting out-of-distribution samples, arXiv preprint arXiv:1711.09325
45. Shalev G, Adi Y, Keshet J (2018) Out-of-distribution detection using multiple semantic label representations, arXiv preprint arXiv:1808.06664
46. Hendrycks D, Basart S, Mazeika M, Mostajabi M, Steinhardt J, Song D (2019) A benchmark for anomaly segmentation, arXiv preprint arXiv:1911.11132
47. Nitsch J, Itkina M, Senanayake R, Nieto J, Schmidt M, Siegwart R, Kochenderfer MJ, Cadena C (2020) Out-of-distribution detection for automotive perception, arXiv preprint arXiv:2011.01413
48. Xia Y, Zhang Y, Liu F, Shen W, Yuille AL (2020) Synthesize then compare: detecting failures and anomalies for semantic segmentation. In: European conference on computer vision, Springer, pp 145–161
49. Goodfellow IJ, Pouget-Abadie J, Mirza M, Xu B, Warde-Farley D, Ozair S, Courville A, Bengio Y (2014) Generative adversarial networks, arXiv preprint arXiv:1406.2661
50. Lu W, Zhou Y, Wan G, Hou S, Song S (2019) L3-net: towards learning based lidar localization for autonomous driving. In: proceedings of the IEEE conference on computer vision and pattern recognition, pp 6389–6398
51. Cordts M, Omran M, Ramos S, Rehfeld T, Enzweiler M, Benenson R, Franke U, Roth S, Schiele B (2016) The cityscapes dataset for semantic urban scene understanding. In: proceedings of the IEEE conference on computer vision and pattern recognition (CVPR)
52. Ertler C, Mislej J, Ollmann T, Porzi L, Neuhold G, Kuang Y (2020) The mapillary traffic sign dataset for detection and classification on a global scale. In: European conference on computer vision, Springer pp 68–84
53. Yin H, Berger C (2017) When to use what data set for your self-driving car algorithm: an overview of publicly available driving datasets. In: 2017 IEEE 20th international conference on intelligent transportation systems (ITSC), IEEE, pp 1–8
54. Yu F, Xian W, Chen Y, Liu F, Liao M, Madhavan V, Darrell T (2018) "Bdd100k: a diverse driving video database with scalable annotation tooling, arXiv preprint arXiv:1805.04687, vol 2, no 5, p 6
55. Brostow GJ, Fauqueur J, Cipolla R (2009) Semantic object classes in video: a high-definition ground truth database. Pattern Recognit Lett 30(2):88–97
56. Scharwächter T, Enzweiler M, Franke U, Roth S (2013) Efficient multi-cue scene segmentation. In: German conference on pattern recognition, Springer, pp 435–445
57. Kondermann, D, Nair R, Honauer K, Krispin K, Andrulis J, Brock A, Gussefeld B, Rahimimoghaddam M, Hofmann S, Brenner C et al. (2016) The hci benchmark suite: stereo and flow ground truth with uncertainties for urban autonomous driving. In: proceedings of the IEEE conference on computer vision and pattern recognition workshops, pp 19–28
58. Kotseruba I, Rasouli A, Tsotsos JK (2016) Joint attention in autonomous driving (jaad), arXiv preprint arXiv:1609.04741
59. Geiger A, Wojek C, Urtasun R (2011) Joint 3d estimation of objects and scene layout. In: advances in neural information processing systems (NIPS)
60. Behley J, Garbade M, Milioto A, Quenzel J, Behnke S, Stachniss C, Gall J (2019)Semantickitti: a dataset for semantic scene understanding of lidar sequences. In: proceedings of the IEEE/CVF international conference on computer vision, pp 9297–9307
61. Kesten R, Usman M, Houston J, Pandya T, Nadhamuni K, Ferreira A, Yuan M, Low B, Jain A, Ondruska P, Omari S, Shah S, Kulkarni A, Kazakova A, Tao C, Platinsky L, Jiang W, Shet V (2019) Lyft level 5 av dataset 2019. urlhttps://level5.lyft.com/dataset/
62. Meyer M, Kuschk G (2019) Automotive radar dataset for deep learning based 3d object detection. In: 2019 16th European radar conference (EuRAD), IEEE, pp 129–132
63. Tan P-N (2009) Receiver operating characteristic. Springer, Boston, MA, USA, pp 2349–2352
64. Davis J, Goadrich M (2006) The relationship between precision-recall and roc curves, vol 06, 06
65. Dosovitskiy A, Ros G, Codevilla F, Lopez A, Koltun V (2017) CARLA: an open urban driving simulator. In: proceedings of the 1st annual conference on robot learning, pp 1–16
66. He K, Zhang X, Ren S, Sun J (2016) Deep residual learning for image recognition. In: 2016 IEEE conference on computer vision and pattern recognition (CVPR), pp 770–778
67. Simonyan K, Zisserman A (2014) Very deep convolutional networks for large-scale image recognition, arXiv preprint arXiv:1409.1556
68. Wohlin C, Runeson P, Höst M, Ohlsson MC, Regnell B, Wesslén A (2012) Planning, Springer, Berlin Heidelberg, pp 89–116

Publisher's Note Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.