AGILE: GIScience Series, 4, 7, 2023. https://doi.org/10.5194/agile-giss-4-7-2023
Proceedings of the 26th AGILE Conference on Geographic Information Science, 2023.
Editors: P. van Oosterom, H. Ploeger, A. Mansourian, S. Scheider, R. Lemmens, and B. van Loenen.
This contribution underwent peer review based on a full paper submission.

Detecting Road Damages in Mobile Mapping Point Clouds using Competitive Reconstruction Networks

Paul Mattes[1], Rico Richter[2], and Jürgen Döllner[1]

[1]University of Potsdam, Digital Engineering Faculty, Hasso Plattner Institute, Potsdam
[2]University of Potsdam, Digital Engineering Faculty, Potsdam

Correspondence: Paul Mattes (pmattes@uni-potsdam.de), Rico Richter (rico.richter.1@uni-potsdam.de), Jürgen Döllner (doellner@uni-potsdam.de)

Abstract.

LiDAR scanning technology is an established method for capturing landscapes, buildings, or roads in order to create a so-called spatial digital twin of the reality, stored as a large collection of 3D coordinates called 3D point cloud. This spatial data offers high density and precision at the cost of hard to extract shape or object information. One popular application of LiDAR 3D point clouds is road condition quality exams. This task is challenging due to a lack of dedicated algorithms to extract and evaluate road point cloud features and due to the large variety of road damages. Deep learning approaches are very promising, but require extensive training data. The data and damage characteristics make data labeling a very difficult and tedious task that often results in mislabeled data, even when performed by trained human operators.

We propose a semi supervised generative adversarial network (GAN) based approach for labeling 2D images rendered from LiDAR point cloud data captured by mobile mapping vehicles, named Competitive Reconstruction Networks (CRN). Our solution trains multiple networks with the same architecture in an ”all vs all” fashion. Our method achieves reliable and robust results on two road image datasets as well as the MVTecAD dataset, and surpass comparable anomaly detection approaches in anomaly detection performance. We also implemented a data generation pipeline to render training images from 3D point cloud of roads and remap anomaly scores back to those 3D point clouds to use the full potential of the 3D data for further analysis.

Keywords. Mobile mapping, anomaly detection, neural networks, generative adversarial networks, LiDAR, 3D point clouds

1 Introduction

Light detection and ranging (LiDAR) systems are widely regarded as the most advanced technology for capturing three-dimensional (3D) spatial data. LiDAR scanners utilize laser beams to scan surfaces and generate 3D point clouds from the reflections. These point clouds are characterized by a high density of points and a large volume of unstructured data (Richter and Döllner (2014)), allowing for the precise measurement of small-scale features on real-world objects (e.g., millimeter range). LiDAR scanners can be mounted on unmanned aerial vehicles (UAVs) or mobile mapping vehicles (MMVs) for various applications, including the analysis of indoor objects and structures, infrastructure networks (e.g., roads, railways), and entire cities and countries (Wolf et al. (2019)).

Evaluating road quality is a crucial task for cities and road construction authorities in order to maintain functioning roads and to have enough data to place road construction sites where there are most needed. Automating and optimizing this task makes road damage repairs cheaper, since damages can be repaired before they get serious, and the negative impacts of road works on traffic flow can be minimized due to smarter planning.

Proper evaluation is very challenging due to the large variety of different damages and missing labeled data. There are different techniques to capture road data in order to identify need of repairs, taking 2D images or videos being the most used one according to a review by Ragnoli et al. (2018). The aforementioned LiDAR scans are only recently being adapted in that area. Most existing road assessment approaches are optimized for the use of 2D data. The incorporation of the third dimension of surface data has the potential to enhance the results of 2D assessment algorithms. In this work, we implement a data preparation pipeline to extract 2D images from 3D road scans, and introduce an anomaly detection approach based on generative adversarial networks. This approach is used to de-

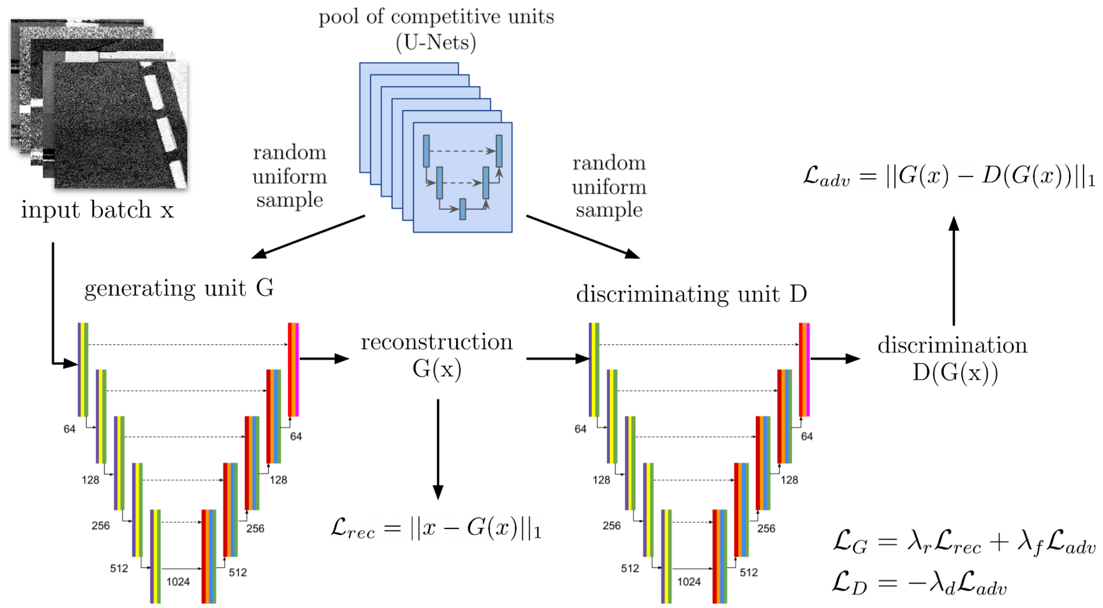
pool of competitive units
(U-Nets)
input batch x
random uniform sample
random uniform sample
$\mathcal{L}_{adv} = ||G(x) - D(G(x))||_1$
generating unit G
discriminating unit D
reconstruction G(x)
discrimination D(G(x))
64
128
256
512
1024
$\mathcal{L}_{rec} = ||x - G(x)||_1$
$\mathcal{L}_G = \lambda_r \mathcal{L}_{rec} + \lambda_f \mathcal{L}_{adv}$
$\mathcal{L}_D = -\lambda_d \mathcal{L}_{adv}$

Figure 1. The training of Competitive Reconstruction Networks, including the loss computation for each training pair. λ_r, λ_f and λ_d are tunable hyperparameters.

termine which of these 2D images contains faulty road segments and which are healthy. We do this in a semi supervised manner in order to circumvent the issues of labeling the training data by hand. Note that to assess road damages, we only work with 2D images enriched/rendered from by 3D data sources, not with the 3D point clouds themselves.

Generative adversarial network (GAN) based anomaly detection algorithms have seen a rise in recent years after the first approach called AnoGAN (short for Anomaly GAN) was introduced by Schlegl et al. (2017). GAN networks first proposed by Goodfellow et al. (2020) consist of a generator and discriminator network that are competing with each other. The generator creates images, and the discriminator tries to differentiate between real images and images created by the generator. The generator then learns to create images that fool the discriminator, and the discriminator learns to better differentiate between real samples and the generated images. That way, after some time, the generator learns to generate very real looking images. To use these networks in anomaly detection, the generator is trained to reconstruct real images instead of creating new ones from random noise. The discriminator is trained to distinguish the reconstructed from the real version of the image. When the generator is fed only healthy images, then its reconstruction abilities will be focused on features found on images without anomalies. When confronted with anomalies, the generator fails to reconstruct the anomalies and thus the reconstruction error is increased when compared to non-anomalous examples.

The initial AnoGAN architecture suffered from learning instabilities and mode collapses, which is why many improvements have been proposed, e.g., in GANomaly (Akçay et al. (2018)), Skip-GANomaly (Akçay et al. (2019)) and DAGAN (Tang et al. (2020)). Other works like the one from Han et al. (2021) focus on adapting the training procedure by training multiple generator and discriminator networks concurrently. Our work builds on the architecture proposed in the DAGAN paper and on the ensemble methods proposed by Han et al. (2021). We introduce a novel ensemble way of training multiple U-Nets of the same architecture in a competitive way. As these networks are trained to reconstruct healthy examples very well while performing worse on unhealthy examples, we call our method Competitive Reconstruction Networks (CRN). Our method surpasses the other approaches and delivers state-of-the-art results on common anomaly detection datasets. We also show that our approach produces superior results on the dataset of generated road images generated from 3D point clouds and on a separate dataset of panorama images taken from roads in the city of Essen, Germany.

2 Related Work

In this section, we present a comparison of various road quality evaluation techniques, including those that utilize algorithmic approaches and those that utilize neural networks or other machine learning procedures. We also compare anomaly detection techniques that utilize GAN architectures.

2.1 Road Quality Evaluation

The diverse range of damages found on roads, such as road rutting, cracks, and potholes, necessitates the use of multiple sensors to accurately monitor road condition. 2D imagery alone is not capable of capturing all possible deformations found on roads. Ragnoli et al. (2018) argue that 2D imagery is often influenced by weather and lightning conditions, shadow casts and camera quality, which makes it hard to find an evaluation method that suits all different capture methods. However, cracks or other damages that are visually easily detectable, can be detected and analyzed by modern object detection approaches very well. 3D LiDAR data, captured by UAVs or MMVs, is less influenced by these conditions. However, environment conditions like puddle formation after heavy rain can still reflect the laser beams sent out by LiDAR systems and thus alter the captured 3D points. Damages like road rutting, basically invisible to 2D images, leave an easily detectable bump in the 3D point surface that is captured in the LiDAR data.

Some evaluation methods use 3D LiDAR data to detect pavement distress. Gézero and Antunes (2019) use an algorithmic approach to measure road rutting depth using LiDAR data, but they used a low precision LiDAR scan, scanning one point every 5 mm. Their results were also fine-tuned on the road segment they examined. Li et al. (2019) also use LiDAR generated 3D point clouds collected with UAVs. They train a random forest classifier on the road points to identify patches with road distress. They select 48 characteristics from their data and achieve an overall accuracy of 95.86% on their validation subset.

Seichter et al. (2018) focus on utilizing 2D imagery and introduce an improved method for image labeling by humans, specifically for road images. They propose using an uncertainty measure to select subregions of images the model is the least familiar with and would most benefit from human labeling. They achieve better model performance while only annotating a fraction of their road image data.

Wu et al. (2019) try to combine 2D and 3D data to exploit the advantages of 2D and 3D data sources for pothole detection and analysis on roads. They use 2D images to detect candidate pothole instances and 3D data to extract pothole points and analyze their features like depth or diameter. However, they still use hand labeled data for their training of networks to find candidate potholes.

All these methods need some sort of human labeled data, which makes applying these approaches on a large scale a difficult task due to changing road and data capturing conditions.

2.2 Anomaly Detection Using GANs

Anomaly detection is a task of great importance in many areas, e.g., in the biomedical (Schlegl et al. (2017)), the financial (Ahmed et al. (2016)) or the fraud detection (Abdallah et al. (2016)) sector. Many approaches rely on reconstruction based anomaly detection, the techniques utilizing GANs fall in this category.

Using Generative Adversarial Networks (Goodfellow et al. (2020)) for detecting anomalous images was initially introduced by Schlegl et al. (2017) with their AnoGAN architecture. They use a generator network to create realistic looking medical scan images of the human eye from a random latent vector z. The discriminator learns to distinguish these generated images from real ones. The authors train their anomaly detection framework on healthy scan images. To find anomalous images, they find the latent representation z' that creates an image closest to the unseen image i by using an optimization algorithm. The anomaly score is then computed using the difference between the generated and the real image and using the discriminator loss on the real image.

As this approach of finding the matching latent representation for each unseen image is rather inefficient, Akçay et al. (2018) introduced GANomaly, using an auto encoder based image generator which no longer generates realistically looking images from a random input instead of reconstructing input images. This way, the reconstruction error can be directly used for anomaly score computation. They achieve higher AUC scores than the initial AnoGAN model while reducing the runtime significantly.

GANomaly, however, still suffered from regular mode collapses and learning instabilities. Akçay et al. (2019) propose using a U-Net auto encoder as the generator, adding skip connections bypassing the bottleneck. This way, the reconstruction abilities of the generator network can be vastly improved. Their Skip-GANomaly architecture yielded better AUC scores on the CIFAR-10 dataset.

Tang et al. (2020) proposed the DAGAN architecture, using the Skip-GANomaly architecture and added ideas from the BEGAN architecture and learning method proposed by Berthelot et al. (2017). For BEGAN, an auto encoder replaces the discriminator network. The discriminator no longer tries to produce a single output value, instead it reconstructs the real images precisely as possible and the generated ones imprecisely. This approach emerges from interpreting the discriminator part of the GAN architecture as an energy based model, which was initially proposed by Zhao et al. (2017). The authors of BEGAN show that this auto encoder discriminator network enables the GAN to be much more stable during training. They also introduce a special balancing technique for the two competing targets of the discriminator (reconstructing real images vs. reconstructing generated ones imprecisely) which further boosts its performance. The DAGAN architecture combines the advantages of an auto encoder generator with skip connections and an auto encoder discriminator. They, however, did not make use of the target balancing technique proposed for BEGAN.

Multiple works consider using more than one network for the generator or discriminator networks (Durugkar et al. (2017); Chavdarova and Fleuret (2018); Choi and Han (2021)). Especially, Hoang et al. (2018) and Arora et al. (2017) show that GAN ensembles can generally outperform single networks in generation tasks. Han et al. (2021)

introduced the idea to train multiple generators with multiple discriminators for anomaly detection. They use multiple different base models, namely f-Anogan (Schlegl et al. (2019)), EGBAD (Zenati et al. (2018)), GANomaly (Akçay et al. (2018)) and Skip-GANomaly (Akçay et al. (2019)) and train ensembles of multiple generators and discriminators. Their results surpass the single base models in several anomaly detection tasks. From the DAGAN architecture (Tang et al. (2020)) and the training procedure of GAN ensembles we derive the concept of Competitive Reconstruction Networks (CRNs) which are described in Chapter 3.

3 Competitive Reconstruction Networks

In this section, we introduce Competitive Reconstruction Networks. An overview is given in Fig. 1. CRNs combine the principles of DAGAN introduced by Tang et al. (2020), where similar architectures were used for both the generator (a U-Net proposed by Ronneberger et al. (2015)) and the discriminator networks (convolutional autoencoder), and of GAN ensembles for anomaly detection proposed by Han et al. (2021). As the generator and the discriminator train to capture the same target distribution (the generator to draw samples from it, the discriminator to distinguish between real and generated samples) they both learn similar information.

The DAGAN (Tang et al. (2020)) architecture consists of a U-Net generator and a convolutional autoencoder as discriminator. Early experiments have shown that replacing the discriminator with a U-net leads to an improvement in its reconstruction capabilities and better discrimination abilities. The reason for this is that the discriminator can focus more on discriminating between real and generated samples and less on the reconstruction part of its loss calculation. This results in better feedback for the generator and an overall improved anomaly detection performance.

These results indicate that even better results could be achieved by further blurring the differences between generator and discriminator. With Competitive Reconstruction Networks (CRN) we propose an approach which completely removes this difference and trains multiple models in a competitive fashion. Competitive training means the models are trained in a ”all vs all” fashion, as opposed to the traditional ”generator vs discriminator” setting in GANs (Goodfellow et al. (2020)).

3.1 CRN Architecture

CRNs consist of a pool of Competitive Units (CUs). Each unit has the same architecture, which is depicted in Fig. 2. The architecture of the CUs is a U-Net (Ronneberger et al. (2015)) like the one used in the Skip-GANomaly architecture (Akçay et al. (2019)). Using a U-Net improves the reconstruction ability of the generator network by adding skip connections, which boosted the detection performance of Skip-GANomaly and was thus adapted by the authors of DAGAN (Tang et al. (2020)).

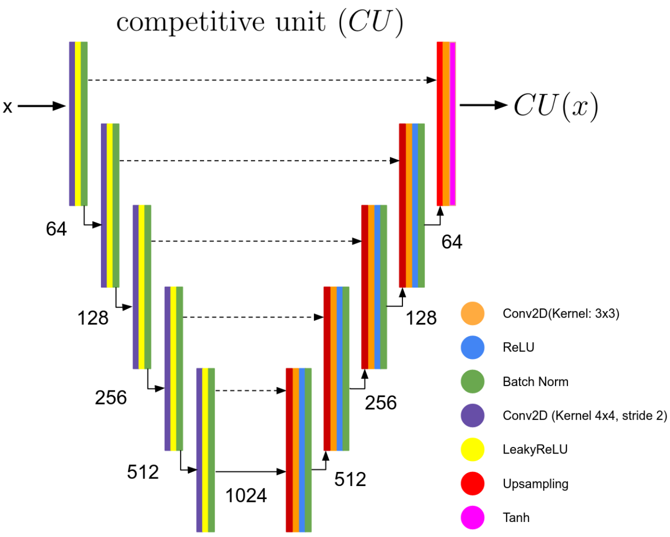

Figure 2. The architecture of a single competitive unit. We use a U-Net architecture with different upscaling than used in Skip-GANomaly from Akçay et al. (2019).

The original U-Net implementation uses transposed convolutional layers for the upsampling part of the network. However, we replaced these with simple upscaling layers using nearest neighbor interpolation followed by a convolution layer with kernel size 3. This approach was proposed by Odena et al. (2016) to avoid checkerboard like artifacts in the upsampled images and further enhance the model’s reconstruction abilities.

3.2 Training Procedure

The CRNs are trained in a semi supervised fashion, i.e., the training data contains only examples without anomalies. The validation and test data then contains samples with and without anomalies. During the training the CUs learn to reconstruct healthy images very precisely and when confronted with anomalous examples from the validation split the reconstruction should be significantly worse.

At each training step, two CUs are picked randomly from the pool, following the training procedure of the GAN ensembles from Han et al. (2021) where for each iteration the generator and discriminator are also chosen randomly. One of these units will act as a generator (G) and the other one will act as a discriminator (D). Updating the units uniformly at random follows the principle of stochastic optimization (Han et al. (2021)) and does not influence the training process negatively. Let n be the number of CUs used in the CRN. Then after n training steps, each unit is expected to be updated two times, one time as a generator, one time as a discriminator.

Let x be an input batch from the train dataset containing only non-anomalous images. The loss computation is very similar to the training of DAGAN (Tang et al. (2020)). Let

$\mathcal{L}_{r_G}$ be the reconstruction loss of the competitive unit G acting as generator, which is computed as follows:

$$\mathcal{L}_{r_G}(x) = ||x - G(x)||_1 \tag{1}$$

We use $||\cdot||_1$ to denote the use of L1 norm, which is defined as $||x||_1 = \sum |x_i|$. Let $\mathcal{L}_{a_{G,D}}$ be the adversarial loss of two units G and D, with G acting as generator and D acting as discriminator. The generator tries to minimize this loss by creating precise reconstructions of the input image, while the discriminator tries to maximize it. It is computed as follows:

$$\mathcal{L}_{a_{G,D}}(x) = ||G(x) - D(G(x))||_1 \tag{2}$$

Equation 3 shows the final loss computation for the generator $\mathcal{L}_G$ and for the discriminator $\mathcal{L}_D$.

$$\mathcal{L}_G(x) = \lambda_r * \mathcal{L}_{r_G}(x) + \lambda_f * \mathcal{L}_{a_{G,D}}(x) \tag{3}$$

$$\mathcal{L}_D(x) = -\lambda_d * \mathcal{L}_{a_{G,D}}(x) \tag{4}$$

λ_r is the weight for the reconstruction loss, λ_f the weight of the adversarial loss applied to the generator and λ_d the weight of the adversarial loss applied to the discriminator. These values are hyperparameters and have to be optimized manually or with the help of dedicated hyperparameter optimization frameworks.

We also tried including the discriminator reconstruction loss L_{r_D} into the discriminator loss computation as it is implemented in DAGAN (i.e., $\mathcal{L}_D(x) = \lambda_r L_{r_D}(x) - \lambda_a \mathcal{L}_{a_{G,D}}(x)$). However, our experiments showed that this actually decreases the overall anomaly detection performance, as the reconstruction loss is then over-represented during training. Since the discriminator will also at some point be trained as a generator, it still learns to reconstruct input images as precise as possible.

In summary, during training, each competitive unit CU tries to minimize the following term:

$$\begin{aligned}\mathcal{L}_{CU}(x) = \quad & \lambda_r \mathcal{L}_{r_{CU}}(x) \\ & + avg\{\lambda_f \mathcal{L}_{a_{CU,D}}(x) \; \forall D \in CP\} \\ & - avg\{\lambda_d \mathcal{L}_{a_{D,CU}}(x) \; \forall D \in CP\}\end{aligned}$$

With x being the input batch and CP being the entire population of competitive units. Currently, this also includes the CU itself, i.e., during training it also learns to produce images that itself cannot distinguish from the input dataset distribution. How this affects the overall learning process has yet to be researched and is left for future work.

3.3 Anomaly Score Calculation

For computing the anomaly score, the input images are reconstructed, and the reconstruction error is measured. The GAN ensembles proposed by Han et al. (2021) relied on averaging the anomaly scores from all used generators and discriminators to even out the influence of eventual under performing networks. For CRNs, we also observed that combining the output of multiple units improved the prediction performance.

However, we observed that some units are trapped in local minima early on in the training process and thus produce very unreliable results for the anomaly detection, which would negatively affect an averaged anomaly score computation. Thus, we select the two most promising units from the population and once again assign the roles of generator G and discriminator D to them. The unit with the smallest reconstruction error is chosen as the generator and the unit with the largest difference between its reconstruction and discrimination loss as the discriminator. That way we make sure, the generator is the most specialized unit to reconstruct non-anomalous samples the most precise and the discriminator is the most specialized unit in identifying samples that are not part of the original sample distribution. This combination of two units proved to consistently outperform other randomly selected pairs of units.

When the input image x is anomalous, then the reconstruction error of $G(x)$ will be higher than if it was non-anomalous. This higher reconstruction error is then amplified by applying the discriminator, which results in an overall better detection performance. The anomaly score A for an input image x is then computed as follows:

$$A(x) = ||G(x) - D(G(x))||_1 \tag{5}$$

In our experiments we also tried chaining more units, feeding the output of the discriminator to another unit and so on, so that the reconstruction error amplification if further increased. However, the best results were achieved by just using two units.

The detection performance is measured by computing the AUC ROC score of the anomaly detection on the validation subset. The AUC ROC score is not influenced by highly imbalanced data, which is why it is commonly used as a measure of binary classification quality. As we can not make any assumption about the distribution of anomalous and non-anomalous data in the validation dataset, the AUC ROC offers the best measure to identify the currently best performing units in the pool.

4 Road Quality Evaluation Datasets

We evaluate the performance of CRNs on two datasets related to road quality assurance. Both were supplied by the Department for Geoinformation, Surveying and Cadastre of Essen. The first dataset consists of 2D camera imagery of roads, the second one consists of 2D image renderings of a 3D LIDAR scan from multiple roads.

4.1 Panorama Images

The dataset consists of 1756 images captured by a panorama 360° camera. As these images include a lot of additional information about road surroundings and vehicles, the images were cropped to only include the roads. The final images have a resolution of 1610×954 pixels. Some examples of damaged and undamaged images are shown in Fig. 3.

The images were annotated by experts from the Department for Geoinformation, Surveying and Cadastre of Essen. The annotations include bounding boxes around all damages found in the image. However, for the task of anomaly detection, the only relevant information is if the image contains any damages whatsoever. The dataset included 386 images without damages and 1370 with at least one road damage. The train split consists of 270 images without damages. The validation dataset consists of the remaining images, i.e., 1370 images with damages and 116 without.

The dataset has a mean of $(0.3546, 0.3349, 0.3145)$ and a standard deviation of $(0.1782, 0.1729, 0.1490)$ for the r, b and g channels, respectively.

4.2 Images Generated from 3D Point Clouds

The 2D road image dataset generation from 3D point clouds is divided into four sub steps: Road extraction, tiling, rasterization and data labeling. These steps will be briefly explained in the following sections. We used thirteen 3D point clouds supplied by the Department for Geoinformation, Surveying and Cadastre of Essen. They include intensity data as well as x, y and z coordinate information for each point.

4.2.1 Road Extraction

To extract the roads points from 3D point clouds, several approaches are possible, e.g., via region growing algorithms (Liu et al. (2018)) or segment analysis (Boyko and Funkhouser). We use a very simple road approximation by using the trajectory of the MMV which scanned the road segment. We then decide if a point p belongs to the road if it fulfills the following criteria:

$$||t_{xy} - p_{xy}||_2 < \alpha \tag{6}$$

t is the closest point of the trajectory to p. For the distance calculation, only the Euclidean distance on the XY-Plane is considered. α is a hand tuned hyperparameter defining the desired road width. To compute these closest point pairs efficiently, the trajectory and point cloud points are structured as a KD tree (Ooi (1987)). Then a ground detection algorithm (Yadav et al. (2017)) is run on the initial point cloud, which generates an approximate ground elevation model. With this model, we compute a distance measure for each point to this approximate ground level. An additional hyperparameter β is chosen to filter out the road points which are too far from the ground.

4.2.2 Road Tiling

The resulting 3D point clouds containing only road points are divided into tiles along the initial MMV trajectory. For this, a tile width $\mathcal{L}_w$ and a tile length $\mathcal{L}_l$ have to be specified. We choose two trajectory points t_1 and t_2 with $|\overrightarrow{t_1 t_2}| \approx l_l$. Let $\boldsymbol{v}$ be defined as follows:

$$\boldsymbol{v} = orth\left(\frac{\overrightarrow{t_1 t_2}}{|\overrightarrow{t_1 t_2}|} * \frac{l_w}{2}\right) \tag{7}$$

$orth(\boldsymbol{a})$ returns the orthogonal vector of a vector $\boldsymbol{a}$.

Then we determine the bounding box for each tile by adding and subtracting $\boldsymbol{v}$ from t_1 and t_2 respectively.

The road points within this rectangle are extracted, translated to the origin and rotated so that all points of this tile lie between $(0, 0)$ and (l_w, l_l).

4.2.3 Tile Rasterization

We compute values for each pixel using the extracted tiles. For this, we use a grid with $r_w \times r_l$ cells. We then split the tile into mini tiles, one for each pixel of the grid. If there are multiple points in a mini tile, we compute the average of the point values (height and intensity). If there are empty cells, we first apply trilinear interpolation to fill these cells. Since this method might leave some cells without values, we additionally use nearest-neighbor interpolation. After this, grid cell values are converted into two PNG images, one containing the height and one containing the intensity values.

4.2.4 Data Labeling

From the thirteen 3D point clouds, we extracted roads of 4 meters width. From these, we generated 17,906 2D images of 3x3 meter road segments with an overlap of 0.667 on the X axis (orthogonal to MMV trajectory) and 0.8 on the Y axis (parallel to MMV trajectory). The images have a resolution of 256×256 pixels.

To test CRNs and other methods on this dataset, we labeled 996 road images, yielding 390 images without damages and 579 images with at least one damage. The training dataset contains 273 images without damages, and the validation contains 579 images with and 117 images without damages. Fig. 4 shows some examples from the healthy and damaged road image portions.

The dataset has a mean of $(0.6124, 0.4959)$ and a standard deviation of $(0.2744, 0.2284)$ for the height and intensity channels, respectively.

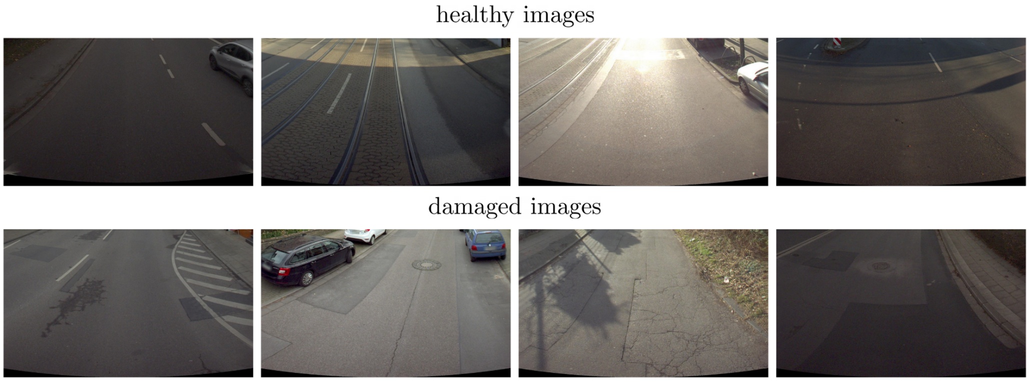

Figure 3. Some examples of the road images obtained from cropped panorama images. These images were annotated by experts at the Department for Geoinformation, Surveying and Cadastre of Essen

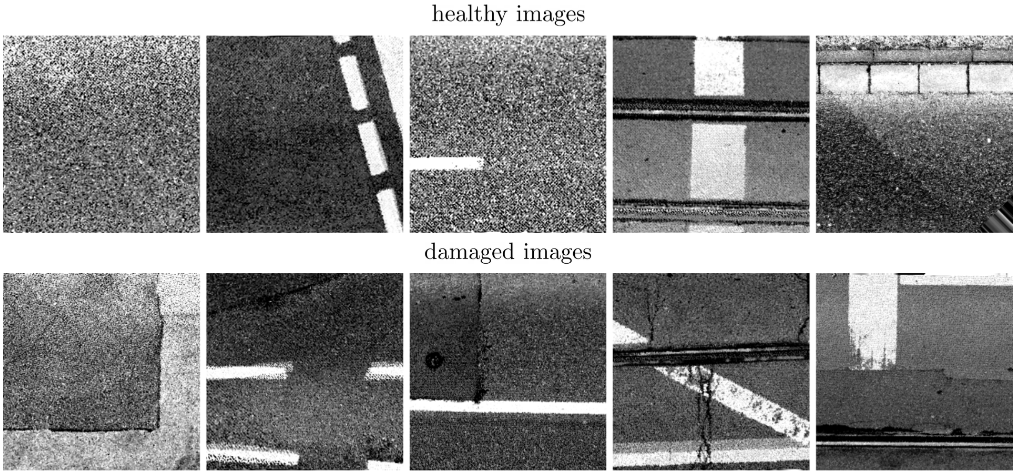

Figure 4. Some examples of the generated ortho-projected images from 3D LiDAR point clouds using our data preparation pipeline.

5 MVTec Anomaly Detection Dataset

To compare our CRNs with other approaches, we test them on a more commonly used benchmark dataset. The MVTec Anomaly Detection Dataset (MVTecAD) introduced by Bergmann et al. (2021) contains 5354 anomalous and non-anomalous high resolution images from ten categories of industrial products, such as cables, bottles or transistors, and from five categories of textures, such as tiles, carpets, or wood. The number of provided training examples is relatively small, ranging from 60 to 391 non-anomalous images in the train split. As the products vary in shape, degree of details and color, the dataset is often used to demonstrate the robustness and performance of anomaly detection methods (Tang et al. (2020); Liang et al. (2022); Zavrtanik et al. (2021); Li et al. (2021); Rudolph et al. (2021)). Fig. 5 shows some examples.

6 Evaluation

In this chapter, we compare the performances of relevant prior works with the current state of the art unsupervised and semi supervised methods with our CRNs.

6.1 Compared Models

We compare the anomaly detection performance of CRNs for MVTecAD with the previous works we are building upon, namely AnoGAN (Schlegl et al. (2017)), GANomaly (Akçay et al. (2018)), Skip-GANomaly (Akçay et al. (2019)), DAGAN (Tang et al. (2020)) and the GAN ensembles (Han et al. (2021)). For the GAN ensembles we had to compute the performance scores ourselves as the authors did not test their approach on MVTecAD. Also, we include the performances of Puzzle-ae (Salehi et al. (2020)) and DGAD (Xia et al. (2021)), two different reconstruction based anomaly detection methods published in 2021. Puzzle-ae, proposed by Salehi et al. (2020), tries to detect anomalies by de- and reconstructing the input images as jigsaw puzzles. DGAD, proposed by Xia et al. (2021), introduces a self supervised framework to train a generator and discriminator jointly using a pretext task to further enhance anomaly detection abilities.

For comparing these approaches, we use the ROC AUC score because this metric is robust against unbalanced datasets and is thus often used as a benchmark for anomaly detection approaches.

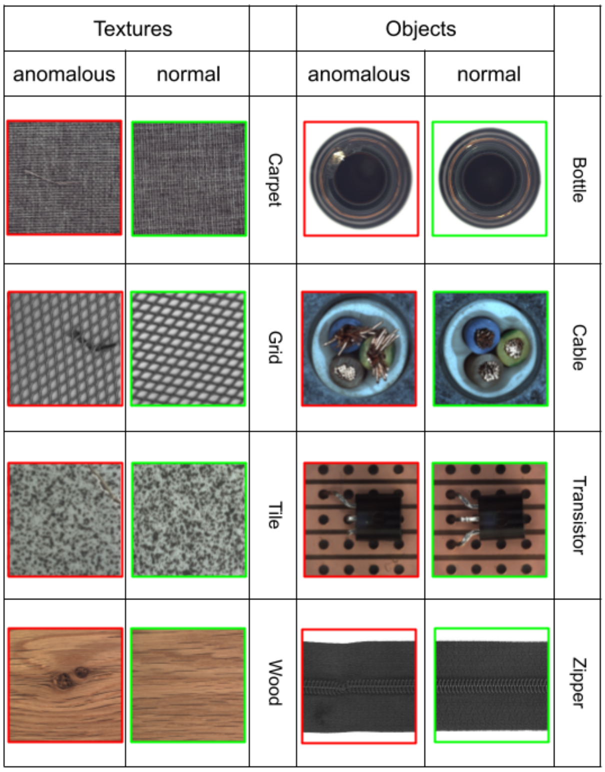

Figure 5. Some examples from different categories of the MVTecAD dataset. In the top row some object categories are displayed, in the bottom row some texture categories. Each category comes with a few hundred examples of anomalous and non-anomalous images.

6.2 Data and Software Availability

The implementation of CRNs and the code for reproducing our results is available in the following repository: https://github.com/Snagnar/CompetitiveReconstructionNetworks and is accessible under the following DOI: https://doi.org/10.5281/zenodo.7682032. The provided code base also includes download links to the three datasets we used: MVTecAD (Bergmann et al. (2021)) and subsets of the panorama and rendered road image datasets. We decided to curate subsets to eliminate any private information contained in these datasets.

To find the set of hyperparameters described in Section 6.3, we used the hyperparameter sweeps of the online logging platform *Weights & Biases* (Biewald (2020)), applying a Bayes optimization algorithm for roughly a hundred iterations. The sweep configuration files are provided in the repository mentioned above. We also used *Weights & Biases* to log all metrics taken during training and evaluation.

The networks were trained on a machine equipped of 8 CPU cores and an NVIDIA A100 GPU.

6.3 Hyperparameters

We trained the CRNs using 12 competitive units. The tradeoff to be considered when choosing an appropriate number of competitive units is discussed in Section 6.7. RAdam, proposed by Liu et al., was used as an optimizer with a learning rate of 0.0001 and a batch size of 64. We chose $\lambda_r = 0.5, lambda_f = 1.0$ and $\lambda_d = 2.0$ as parameters for the loss equation 3. The CRNs train for 20,000 training steps for each category.

6.4 MVTecAD

The results we obtained from training our CRNs on the MVTecAD dataset are shown in Table 1. As there were no numbers reported for the GAN ensembles (Han et al. (2021)) trained on MVTecAD, we adapted their code and trained an ensemble of five Skip-GANomaly (Akçay et al. (2019)) generators and five discriminators on the MVTecAD images.

To demonstrate the superiority of our competitive training approach, we also trained a CRN with only two competitive units on the MVTecAD dataset. This way, we use the same number of networks as other standard GAN-based methods (like DAGAN (Tang et al. (2020))) with the only major difference, that the two networks frequently change the roles of generator and discriminator. The results are shown in the 2-CRN column of Table 1.

The ROC AUC scores obtained from training CRN on the different anomaly detection categories is the highest among all compared models in most cases. In the cases where other models performed better, our score is always very close to the best result, leading to an overall superior average performance across the texture and object categories.

Our approach far outperforms DAGAN and the GAN ensembles, the two main works from which we derived the CRN architecture. The reported average ROC AUC score for the texture categories are significantly worse than the proposed CRN model.

The 2-CRN average scores are also better than any other compared approach, using the same amount of networks as a traditional GAN procedure. This indicates that, in fact, the competitive training paradigm is superior to adversarial training of two networks with fixed roles. With more units, we can further leverage the benefits of competitive training and achieve state-of-the-art results on the MVTecAD benchmarking dataset.

6.5 Panorama Images

We trained our CRN on the road panorama image dataset. We used the same hyperparameters as used for the MVTec dataset. Better results might be achievable by running a dedicated hyperparameter search for this dataset.

Table 1. ROC AUC scores of various anomaly detection approaches on the MVTec anomaly detection dataset. We report two sets of scores for our proposed CRN approach: the CRN column reports the results obtained with the hyperparameters specified in Section 6.3 and the 2-CRN column reports the results from experiments using only two competitive units. The best scores in the specific categories are shown in bold font.

		Previous Work					Reconstruction Based			
	odels Year of Publication	Ano-GAN '17	GANomaly '18	Skip-GANomaly '19	DAGAN '20	GAN Ensemble (Skip) '20	Puzzle-ae '21	DGAD '21	2-CRN (Ours)	CRN (Ours)
Textures	Carpet	49.0	82.1	79.5	90.3	53.1	65.7	52.0	87.0	**97.5**
	Grid	51.0	74.3	65.7	86.7	72.9	75.4	67.0	98.9	**100.0**
	Leather	52.0	80.8	90.8	94.4	76.8	72.9	94.0	96.6	**97.5**
	Tile	51.0	72.0	85.0	96.1	90.3	65.5	83.0	79.9	**98.6**
	Wood	68.0	92.0	91.9	97.9	85.0	89.5	72.0	99.6	**100.0**
	Average	54.2	80.24	82.6	93.1	75.6	73.8	73.6	92.4	**98.7**
Objects	Bottle	69.0	79.4	93.7	**98.3**	41.3	94.2	97.0	85.6	97.1
	Cable	53.0	71.1	67.4	66.5	59.9	87.9	**90.0**	79.8	87.5
	Capsule	58.0	72.1	71.8	68.7	59.7	66.9	60.0	76.9	**92.8**
	Hazelnut	50.0	87.4	90.6	**100.0**	98.0	91.2	80.0	94.4	98.1
	Metal Nut	50.0	69.4	79.0	81.5	42.8	66.3	**95.0**	86.0	93.2
	Pill	62.0	67.1	75.8	76.8	37.6	71.6	76.0	81.7	**89.1**
	Screw	35.0	**100.0**	**100.0**	**100.0**	**100.0**	57.8	67.0	**100.0**	**100.0**
	Toothbrush	57.0	70.0	68.9	95.0	53.3	97.8	93.0	91.9	**100.0**
	Transistor	67.0	80.8	81.4	79.4	45.8	86.0	88.0	84.3	**92.8**
	Zipper	59.0	74.4	66.3	78.1	33.4	75.7	82.0	80.1	**86.4**
	Average	56.0	77.17	79.5	84.4	57.2	79.5	82.8	86.1	**94.5**
	Overall	55.0	78.19	80.5	87.3	63.3	77.6	80.0	88.2	**95.9**

We also trained DAGAN (Tang et al. (2020)), Skip-GANomaly (Akçay et al. (2019)) and the GAN ensembles using Skip-GANomaly (Han et al. (2021)) models on this dataset to compare the different model performances. In case of Skip-GANomaly and the GAN ensembles, we used the publicly available reference implementations. As there is no public implementation of DAGAN we reimplemented their model from the paper and used the results of the MVTecAD dataset to assure the correctness of our implementation. For the GAN ensembles, we used 5 generators and 5 discriminators, as this comes closest to our approach using 10 competitive units. The results are shown in Table 2.

Table 2. The ROC AUC scores of Skip-GANomaly, DAGAN, GAN Ensembles and our CRNs on the road panorama image dataset.

Model	ROC AUC
Skip-GANomaly	87.4
DAGAN	90.5
GAN Ensembles (Skip)	67.6
CRN (Ours)	**93.0**

Our CRN model achieves significantly better results than the other models from previous works by a significant margin. Interestingly, the ensembles using Skip-GANomaly perform much worse than using a single Skip-GANomaly generator and discriminator. This indicates that the training procedure used by the GAN ensembles is less capable of capturing more complex image contexts. The panorama image dataset contains a lot of skewing information like shadows, cars or vegetation, that have no influence on the road condition and have to be correctly identified as such by the models.

6.6 Images Generated from 3D Point Clouds

We also trained our CRN on the labeled road images rendered from 3D point clouds. We used the same hyperparameters as for the MVTecAD dataset as well. For comparison, we also trained the DAGAN model (Tang et al. (2020)), Skip-GANomaly (Akçay et al. (2019)) and a GAN ensemble (Han et al. (2021)) with 5 Skip-GANomaly generators and discriminators each. The results are depicted in Table 3.

Table 3. The ROC AUC scores of Skip-GANomaly, DAGAN, GAN ensembles and our CRNs on the road image dataset rendered from 3D LIDAR point clouds.

Model	ROC AUC
Skip-GANomaly	60.4
DAGAN	77.1
GAN Ensembles (Skip)	57.3
CRN (Ours)	**89.6**

Our approach once again outperforms the previous approaches now with a way larger gap in ROC AUC score.

One explanation could be the different upscaling technique used in the U-Net architecture. When training DAGAN,

GAN ensembles or Skip-GANomaly on the road image dataset, the variance of the discrimination image often collapses to 0, producing gray output images with the same value for each pixel. One reason for this could be the very monotonous appearance of healthy road images, which often just display a mostly gray image with very fine details (see Fig. 4). The CRNs, however, are able to maintaining a non-zero variance in the reconstruction images throughout the training process due to the improved U-Net architecture which was described in Section 3.1. An algorithmic upscaling layer without learned parameters seems to help to preserve fine details in the reconstructing image, and thus prevents total feature collapse in the discriminating unit.

6.6.1 Anomaly Score Remap

In order to assess the efficacy of our CRNs in detecting anomalies in unseen 3D point clouds, we select the pair of CUs that demonstrated the highest performance on the dataset containing rendered road images from 3D point clouds. These CUs were used to generate anomaly scores for all 3D road point clouds. The results are depicted in Fig. 6.

In order to facilitate the qualitative assessment, we colored the 3D point clouds yellow where the anomaly score assigned to a point exceeded a threshold of 0.5. As depicted in Subfig. 6a), our CRNs were able to accurately identify anomalous road parts (i.e., road damages) in 3D point clouds that were not included in the initial training data. In addition, our models demonstrate accuracy even in heavily damaged areas. Subfig. 6c) illustrates that the models did not confuse road markings for anomalies, even in cases of complex shapes (such as the large arrow on the road). This demonstrates the robustness of our approach.

However, the anomaly scores in close vicinity around the large arrow are very high, while the road seems not to contain any damages in that area. This shows that in some cases, especially around complex forms of road markings, the CRNs fail to reconstruct the surroundings meaningfully. This can be seen in many examples throughout the data we processed in our experiments. One possible explanation could be, that the models learned during training that reconstructing the complex shape of road markings is more important in these particular cases than reconstructing the other parts of the image, since errors in reconstructing (e.g., the borders of road markings) lead to a large increase in reconstruction loss.

The 3D point cloud visualization in Subfig. 6d) appears to lack any visible damage. However, the height landscape reveals the presence of heavy rutting. The corresponding anomaly score visualization correctly identifies the rutting lanes as anomalous, demonstrating the ability of our CRNs to robustly identify damages that are limited to only one channel of information.

Processing all generated 17,906 images (Section 4.2.4) took approximately 11 minutes on the machine used to train the CRNs (described in Section 6.2). The 13 point clouds we extracted the road image dataset from covered a length of approximately 5.4 kilometers, which results in a total processing speed of $\approx$ 30 kilometers per hour. Right now, this number excludes the process of rendering the images and remapping the anomaly scores.

6.7 Number of Units vs ROC AUC

We tested how adding more competitive units affects the overall detection performance. For this, we trained CRNs with a varying number of units on the cable, leather, metal nut, pill, and tile categories of the MVTecAD anomaly detection dataset. Fig. 7 shows the ROC AUC score averaged over the five categories of the MVTecAD dataset for the specified number of units. This assures, that the findings are consistent across different training data distributions. It can be observed that more units indeed improve the anomaly detection performance of CRNs. This is consistent with other ensemble methods (not only the mentioned GAN ensembles by Han et al. (2021)) where larger ensembles often yield better results than ensembles with lesser units, as shown in Section 2.2.

Fig. 7 shows, that up until nine units, the performance increased approximately linearly with each added unit. We trained each run with the same number of training steps, which lead to fewer training steps per unit the more units were used in the CRN. This explains the increasing value variance in the runs with more units, since the single units inside the pool are trained far less than in the runs with fewer units. It also explains the eventual average decline in performance of CRNs with more than 16 units. As described in Section 6.3, for training on the benchmarking datasets, we decided for 12 competitive units, as there seems no further improvement achievable by adding more units after this point.

7 Conclusion

In this study, we propose a novel method called Competitive Reconstruction Network (CRN) for the task of anomaly detection. CRN involves training multiple networks in a competitive ”all-vs-all” manner, with the aim of correctly reconstructing non-anomalous samples while producing larger reconstruction errors for samples containing anomalies. We evaluate the performance of CRNs on the MVTecAD (Bergmann et al. (2021)) dataset, a widely used benchmark for anomaly detection. Our results demonstrate that CRN outperforms other ensemble and reconstruction-based methods, achieving state-of-the-art performance. We also show that in many cases the competitive training paradigm is superior to the traditional adversarial training of GANs, even when using only two units.

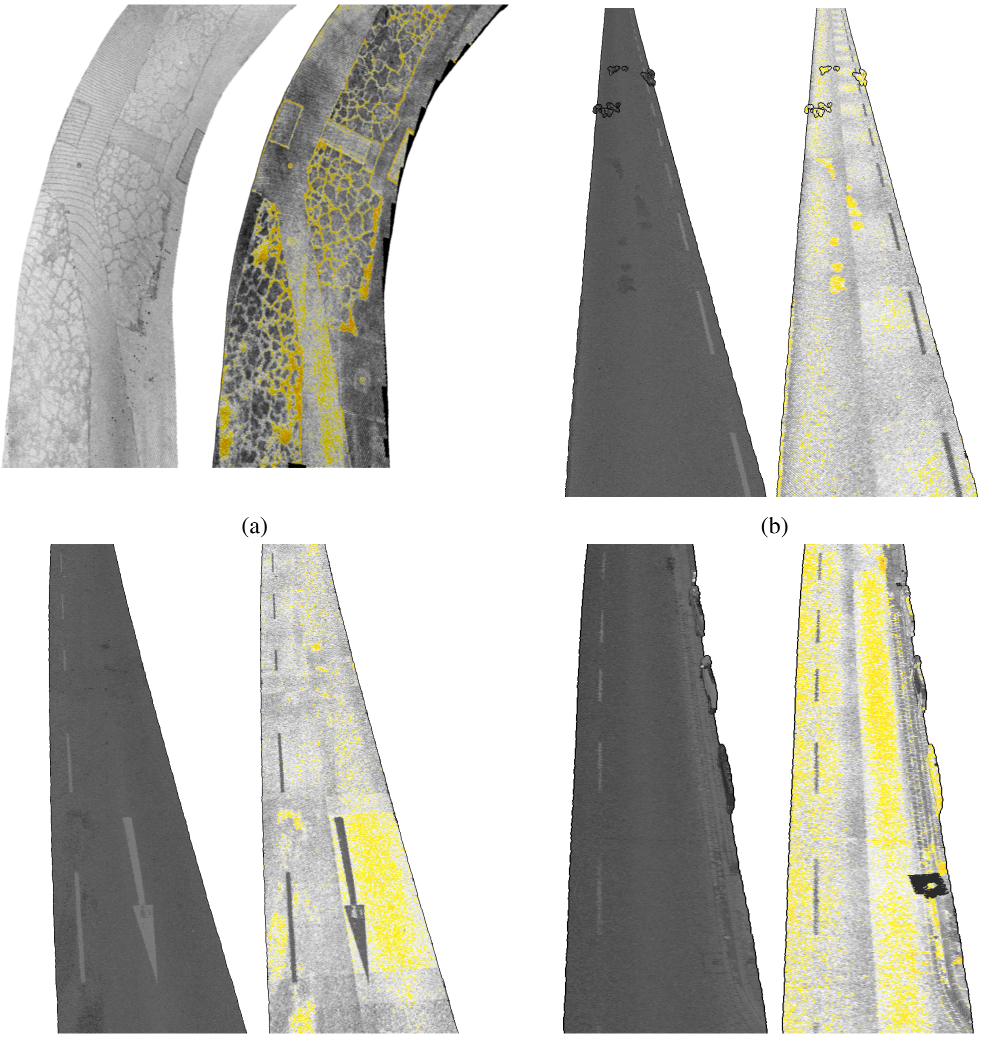
(a)
(b)
(c)
(d)

Figure 6. Different examples of anomaly scores remapped into the original 3D point clouds. The images on the left of each subfigure are the visualization of the intensity values of the 3D road point clouds. The images on the right are the visualization of the anomaly scores assigned to each point. Road parts that are classified as an anomaly are colored yellow, non-anomalous parts are colored gray. The division into anomalous and non-anomalous is done via a threshold.

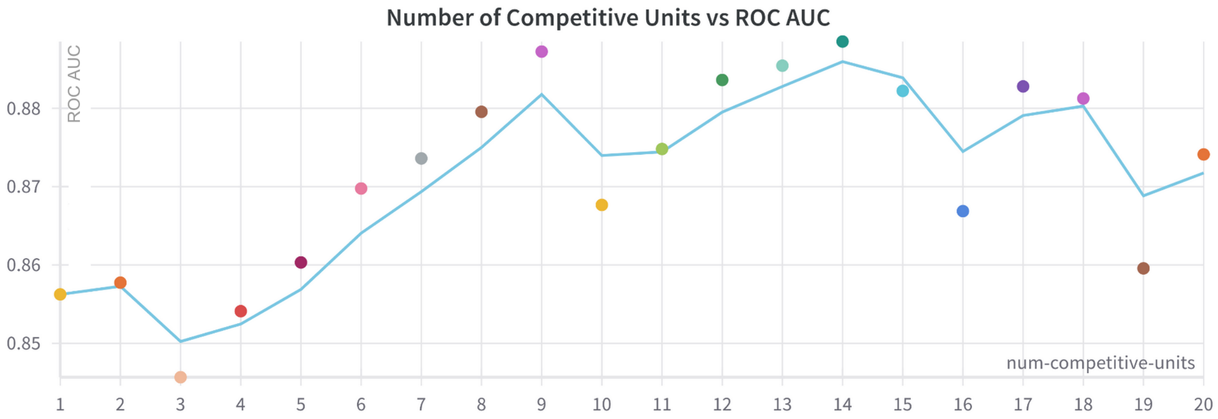

Figure 7. The influence of a different number of units affects the ROC AUC score, with an approximating graph added for better readability. All data points are the average of five runs with three different categories of the MVTecAD dataset: cable, leather, metal nut, pill, and tile. This assures, that the findings are consistent across different training data distributions.

Additionally, we introduce two datasets consisting of road imagery for the purpose of anomaly detection. One dataset includes camera images of both damaged and undamaged road sections, while the other consists of images rendered from 3D point clouds of roads. Using CRNs, we demonstrate their ability to reliably identify damaged road sections. An interactive visualization of the results provides a qualitative assessment of the method's effectiveness.

8 Future Work

In order to further evaluate the performance of our CRN architecture, additional testing on various anomaly detection benchmark datasets is necessary. For example, previous research has used datasets such as Cifar10 and MNIST to evaluate the effectiveness of various approaches, including GAN ensembles. In a clinical setting, the OCT dataset (Kermany et al. (2018)) has often been utilized to assess model performance using high resolution biomedical images. Also, a direct comparison between the anomaly detection performed by the CRNs and images annotated by experts would give further insight into the quality of the obtained results. Section 6.6.1 refers to some cases of false positives, however a detailed evaluation of when these false positives occur remains to be done.

In the current setup, we have trained two competitive units against each other at each step. However, it is possible that a different training method, such as a ”classroom” setting where a single unit receives feedback from multiple or all other units, may yield improved results. This approach involves one unit reconstructing the input and the one or all other units computing a discriminative loss for this reconstruction, which may lead to a faster and more stable conversion. However, due to the potential increase in computational complexity, we did not test this method, and it is left for future work.

Additionally, a comprehensive theoretical analysis of the competitive training paradigm has yet to be conducted. The ”all-vs-all” training introduces significant modifications to the originally proposed adversarial learning strategy, and the game theory behind a group of autonomous models competing with one another is a complex topic that requires further investigation. This is also an area for future research.

Acknowledgements. We thank the anonymous reviewers for their valuable feedback and the Department for Geoinformation, Surveying and Cadastre of Essen, Germany for providing mobile mapping data. This work was partially funded by the Federal Ministry of Education and Research, Germany through grant 01IS22062 "AI research group FFS-AI" and the Federal Ministry for Digital and Transport through grant 19F2210 "TWIN4ROAD".

References

Abdallah, A., Maarof, M. A., and Zainal, A.: Fraud detection system: A survey, Journal of Network and Computer Applications, 68, 90–113, https://doi.org/10.1016/j.jnca.2016.04.007, 2016.

Ahmed, M., Mahmood, A. N., and Islam, M. R.: A survey of anomaly detection techniques in financial domain, Future Generation Computer Systems, 55, 278–288, https://doi.org/10.1016/j.future.2015.01.001, 2016.

Akçay, S., Atapour-Abarghouei, A., and Breckon, T. P.: Skip-ganomaly: Skip connected and adversarially trained encoder-decoder anomaly detection, in: International Joint Conference on Neural Networks (IJCNN), pp. 1–8, IEEE, https://doi.org/10.1109/IJCNN.2019.8851808, 2019.

Akçay, S., Atapour-Abarghouei, A., and Breckon, T. P.: Ganomaly: Semi-supervised anomaly detection via adversarial training, in: Asian conference on computer vision, pp. 622–637, Springer, https://doi.org/10.1007/978-3-030-20893-6_39, 2018.

Arora, S., Ge, R., Liang, Y., Ma, T., and Zhang, Y.: Generalization and equilibrium in generative adversarial nets (gans), in: International Conference on Machine Learning, pp. 224–232, PMLR, https://doi.org/10.1145/3188745.3232194, 2017.

Bergmann, P., Batzner, K., Fauser, M., Sattlegger, D., and Steger, C.: The MVTec anomaly detection dataset: a comprehensive real-world dataset for unsupervised anomaly detection, International Journal of Computer Vision, 129, 1038–1059, https://doi.org/10.1007/s11263-020-01400-4, 2021.

Berthelot, D., Schumm, T., and Metz, L.: BEGAN: Boundary Equilibrium Generative Adversarial Networks, CoRR, abs/1703.10717, http://arxiv.org/abs/1703.10717, 2017.

Biewald, L.: Experiment Tracking with Weights and Biases, https://www.wandb.com/, software available from wandb.com, 2020.

Boyko, A. and Funkhouser, T.: Extracting roads from dense point clouds in large scale urban environment, ISPRS Journal of Photogrammetry and Remote Sensing, 66, 6, 2–12, https://doi.org/10.1016/j.isprsjprs.2011.09.009.

Chavdarova, T. and Fleuret, F.: Sgan: An alternative training of generative adversarial networks, in: IEEE conference on computer vision and pattern recognition, pp. 9407–9415, https://doi.org/10.1109/CVPR.2018.00980, 2018.

Choi, J. and Han, B.: MCL-GAN: Generative Adversarial Networks with Multiple Specialized Discriminators, CoRR, abs/2107.07260, https://arxiv.org/abs/2107.07260, 2021.

Durugkar, I. P., Gemp, I., and Mahadevan, S.: Generative Multi-Adversarial Networks, in: 5th International Conference on Learning Representations, ICLR, OpenReview.net, https://doi.org/10.48550/arXiv.1611.01673, 2017.

Gézero, L. and Antunes, C.: Road Rutting Measurement Using Mobile LiDAR Systems Point Cloud, ISPRS International Journal of Geo-Information, 8, 404, https://doi.org/10.3390/ijgi8090404, 2019.

Goodfellow, I., Pouget-Abadie, J., Mirza, M., Xu, B., Warde-Farley, D., Ozair, S., Courville, A., and Bengio, Y.: Generative adversarial networks, Communications of the ACM, 63, 139–144, https://doi.org/10.1145/3422622, 2020.

Han, X., Chen, X., and Liu, L.-P.: Gan ensemble for anomaly detection, in: Proceedings of the AAAI Conference on Artificial Intelligence, vol. 35, pp. 4090–4097, https://doi.org/10.1609/aaai.v35i5.16530, 2021.

Hoang, Q., Nguyen, T. D., Le, T., and Phung, D.: MGAN: Training generative adversarial nets with multiple generators, in: International conference on learning representations, 2018.

Kermany, D. S., Goldbaum, M., Cai, W., Valentim, C. C., Liang, H., Baxter, S. L., McKeown, A., Yang, G., Wu, X., Yan, F., et al.: Identifying medical diagnoses and treatable diseases by image-based deep learning, Cell, 172, 1122–1131, https://doi.org/10.1016/j.cell.2018.02.010, 2018.

Li, C.-L., Sohn, K., Yoon, J., and Pfister, T.: Cutpaste: Self-supervised learning for anomaly detection and localization, in: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp. 9664–9674, https://doi.org/10.1109/CVPR46437.2021.00954, 2021.

Li, Z., Cheng, C., Kwan, M.-P., Tong, X., and Tian, S.: Identifying asphalt pavement distress using UAV LiDAR point cloud data and random forest classification, ISPRS International Journal of Geo-Information, 8, 39, https://doi.org/10.3390/ijgi8010039, 2019.

Liang, Y., Zhang, J., Zhao, S., Wu, R., Liu, Y., and Pan, S.: Omni-frequency Channel-selection Representations for Unsupervised Anomaly Detection, CoRR, abs/2203.00259, https://doi.org/10.48550/arXiv.2203.00259, 2022.

Liu, L., Jiang, H., He, P., Chen, W., Liu, X., Gao, J., and Han, J.: On the Variance of the Adaptive Learning Rate and Beyond, in: 8th International Conference on Learning Representations, ICLR 2020.

Liu, Y., Ma, C., Li, L., Xing, X., Zhang, Y., Wang, Z., and Xu, J.: A Road Extraction Method Based on Region Growing and Mathematical Morphology from Remote Sensing Images, Journal of Computer and Communications, 6, 91–97, https://doi.org/10.4236/jcc.2018.611008, 2018.

Odena, A., Dumoulin, V., and Olah, C.: Deconvolution and checkerboard artifacts, Distill, 1, e3, https://doi.org/10.23915/distill.00003, 2016.

Ooi, B. C.: Spatial kd-tree: A data structure for geographic database, pp. 247–258, https://doi.org/10.1007/978-3-642-72617-0_17, 1987.

Ragnoli, A., De Blasiis, M. R., and Di Benedetto, A.: Pavement distress detection methods: A review, Infrastructures, 3, 58, https://doi.org/10.3390/infrastructures3040058, 2018.

Richter, R. and Döllner, J.: Concepts and techniques for integration, analysis and visualization of massive 3D point clouds, Computers, Environment and Urban Systems, 45, 114–124, https://doi.org/10.1016/j.compenvurbsys.2013.07.004, 2014.

Ronneberger, O., Fischer, P., and Brox, T.: U-net: Convolutional networks for biomedical image segmentation, in: International Conference on Medical image computing and computer-assisted intervention, pp. 234–241, Springer, https://doi.org/10.1007/978-3-319-24574-4_28, 2015.

Rudolph, M., Wandt, B., and Rosenhahn, B.: Same same but differnet: Semi-supervised defect detection with normalizing flows, in: Proceedings of the IEEE/CVF winter conference on applications of computer vision, pp. 1907–1916, https://doi.org/10.1109/WACV48630.2021.00195, 2021.

Salehi, M., Eftekhar, A., Sadjadi, N., Rohban, M. H., and Rabiee, H. R.: Puzzle-AE: Novelty Detection in Images through Solving Puzzles, CoRR, abs/2008.12959, https://arxiv.org/abs/2008.12959, 2020.

Schlegl, T., Seeböck, P., Waldstein, S. M., Schmidt-Erfurth, U., and Langs, G.: Unsupervised anomaly detection with generative adversarial networks to guide marker discovery, in: International conference on information processing in medical imaging, pp. 146–157, Springer, https://doi.org/10.1007/978-3-319-59050-9_12, 2017.

Schlegl, T., Seeböck, P., Waldstein, S. M., Langs, G., and Schmidt-Erfurth, U.: f-AnoGAN: Fast unsupervised anomaly detection with generative adversarial networks, Medical image analysis, 54, 30–44, https://doi.org/10.1016/j.media.2019.01.010, 2019.

Seichter, D., Eisenbach, M., Stricker, R., and Gross, H.-M.: How to improve deep learning based pavement distress detection while minimizing human effort, in: 2018 IEEE 14th International Conference on Automation Science and Engineering (CASE), pp. 63–70, IEEE, https://doi.org/10.1109/COASE.2018.8560372, 2018.

Tang, T.-W., Kuo, W.-H., Lan, J.-H., Ding, C.-F., Hsu, H., and Young, H.-T.: Anomaly detection neural network with dual auto-encoders GAN and its industrial inspection applications, Sensors, 20, 3336, https://doi.org/10.3390/s20123336, 2020.

Wolf, J., Richter, R., and Döllner, J.: Techniques for Automated Classification and Segregation of Mobile Mapping 3D Point Clouds, in: 14th International Joint Conference on Computer Vision, Imaging and Computer Graphics Theory and Applications - GRAPP, pp. 201–208, INSTICC, SciTePress, https://doi.org/10.5220/0007308802010208, 2019.

Wu, H., Yao, L., Xu, Z., Li, Y., Ao, X., Chen, Q., Li, Z., and Meng, B.: Road pothole extraction and safety evaluation by integration of point cloud and images derived from mobile mapping sensors, Advanced Engineering Informatics, 42, 100 936, https://doi.org/10.1016/j.aei.2019.100936, 2019.

Xia, X., Pan, X., He, X., Zhang, J., Ding, N., and Ma, L.: Discriminative-Generative Representation Learning for One-Class Anomaly Detection, CoRR, abs/2107.12753, https://arxiv.org/abs/2107.12753, 2021.

Yadav, M., Singh, A. K., and Lohani, B.: Extraction of road surface from mobile LiDAR data of complex road environment, International Journal of Remote Sensing, 38, 4655–4682, https://doi.org/10.1080/01431161.2017.1320451, 2017.

Zavrtanik, V., Kristan, M., and Skočaj, D.: Draem – a discriminatively trained reconstruction embedding for surface anomaly detection, in: Proceedings of the IEEE/CVF International Conference on Computer Vision, pp. 8330–8339, https://doi.org/10.1109/ICCV48922.2021.00822, 2021.

Zenati, H., Foo, C. S., Lecouat, B., Manek, G., and Chandrasekhar, V. R.: Efficient GAN-Based Anomaly Detection, CoRR, abs/1802.06222, http://arxiv.org/abs/1802.06222, 2018.

Zhao, J. J., Mathieu, M., and LeCun, Y.: Energy-based Generative Adversarial Networks, in: 5th International Conference on Learning Representations, ICLR 2017, OpenReview.net, 2017.