Detecting Anomalous 3D Point Clouds Using Pre-Trained Feature Extractors

Dario Mantegazza[a] and Alessandro Giusti[b]
*Dalle Molle Institute for Artificial Intelligence (IDSIA),
USI-SUPSI, Lugano, Switzerland*

Keywords: Anomaly Detection, 3D Point Clouds, Deep Learning.

Abstract: In this paper we explore the status of the research effort for the task of 3D visual anomaly detection; in particular, we investigate whether it is possible to find anomalies on 3D point clouds using off-the-shelf feature extractors, similar to what is already feasible on images, without the requirement of an ad-hoc one. Our work uses a model composed of two parts: a feature extraction module and an anomaly detection head. The latter is fixed and works on the embeddings from the feature extraction module. Using the MVTec-3D dataset, we contribute a comparison between a 3D point cloud features extractor, a 2D image features extractor, a combination of the two, and three baselines. We also compare our work with other models on the dataset's DETECTION-AUROC benchmark. The experiment results demonstrate that, while our proposed approach surpasses the baselines and some other approaches, our best-performing model cannot beat purposely developed ones. We conclude that a combination of dataset size and 3D data complexity is the culprit to a lack of off-the-shelf feature extractors for solving complex 3D vision tasks.

1 INTRODUCTION

Off-the-shelf pre-trained image feature extractors are increasingly being used in academic and industrial research for building deep-learning models to solve computer vision tasks. Recently, Vision Transformer (Dosovitskiy et al., 2021)(ViT) paved a new road for researchers to build even more complex and performing models for computer-vision tasks. A notable example of ViT-based models is CLIP from OpenAI (Radford et al., 2021) a very large and complex computer vision model trained on an enormous corpus of captioned images to solve any kind of vision task. Due to their size and reliance on large and complex datasets, models such as CLIP can be only developed and trained by a limited set of companies and research labs. However, most of these large models share an open-source nature with pre-trained models available online [1]. By removing the need to train an ad-hoc feature extractor, researchers can focus on solving the task at hand using the extracted feature embeddings; regularly smaller than images which contain low-level semantic information, the embeddings encode visual data into high-level semantic features allowing researchers to train computer vision models with fewer samples or smaller models (excluding the extractor).

In a similar fashion, we are seeing a wider use, in both academia and industry, of 3D data through depth cameras, LIDARs, photogrammetry representation or Neural Radiance fields for solving computer vision tasks. Nonetheless, one 3D task that is still understudied (Frittoli, 2022) is Anomaly Detection on 3D Data. The task of 3D Anomaly Detection has potential applications in many fields such as health care, industrial product inspection, industrial asset maintenance, site surveillance and robotics; currently, all of these fields only rely on 2D Anomaly Detection.

A few recent works (Horwitz and Hoshen, 2023; Rudolph et al., 2023; Wang et al., 2023; Chu et al., 2023; Masuda et al., 2021; Floris et al., 2022) approached the task of Anomaly Detection on 3D data, with most (Horwitz and Hoshen, 2023; Rudolph et al., 2023; Wang et al., 2023; Chu et al., 2023) focusing on Segmentation of Anomalies.

In this paper, we ask ourselves if, with the current state of the art in deep learning models for 3D Point Clouds, it is possible to solve the task of anomaly detection on 3D point clouds without training an ad-hoc

[a] https://orcid.org/0000-0001-9088-0897
[b] https://orcid.org/0000-0003-1240-0768
[1] https://github.com/openai/CLIP

Mantegazza, D. and Giusti, A.
Detecting Anomalous 3D Point Clouds Using Pre-Trained Feature Extractors.
DOI: 10.5220/0012465600003660

In *Proceedings of the 19th International Joint Conference on Computer Vision, Imaging and Computer Graphics Theory and Applications (VISIGRAPP 2024) - Volume 4: VISAPP*, pages 373-380
ISBN: 978-989-758-679-8; ISSN: 2184-4321

features extractor, similarly to what we achieve in our previous work (Mantegazza et al., 2023). The use of pre-trained 3D feature extractors would remove the need for a difficult-to-develop and train 3d features extractor, lowering the entrance barrier to 3D visual data analysis.

2 RELATED WORK

2.1 Dataset

For this work, we use the MVTec-3D dataset (Bergmann. et al., 2022). To the best of our knowledge, this is the only existing open-access dataset for the task of 3D Anomaly Detection, and more specifically, 3D Anomaly Segmentation.

The MVTec-3D dataset is built for studying the task of 3D anomaly segmentation in the context of industrial mass production; the dataset is composed of more than 4000 high-resolution point clouds and RGB images of 10 different objects with 10 different anomalies, captured using an industrial 3D sensor. The dataset is already subdivided into training, validation and testing sets; all sets contain normal samples, classified into different object categories, but only the testing set contains anomalous samples.

For each anomalous test sample, a precisely annotated ground truth is provided; in Figure 1 a selection of the dataset is shown.

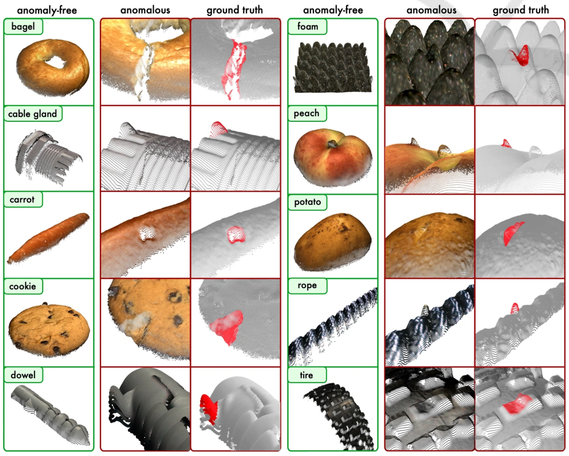

Figure 1: Examples of samples from the Mvtec3D dataset.

2.2 Models and Approaches

2.2.1 Image Anomaly Detection

Anomaly Detection is a widely researched topic (Chandola et al., 2009) applied to a variety of fields (Ruff et al., 2021). As for other Machine Learning branches, even Anomaly Detection has been extensively researched with images as the information medium. As explained in their review (Ruff et al., 2021) by Ruff et al., when operating on images, the task of anomaly detection requires an understanding of normal data in order to find *high-level* semantics anomalies. Thankfully, deep Learning methods have been used in different fields to extract high-level information from complex data, and this applies also to anomaly detection. Deep learning-based anomaly detection has been studied for many applications, from security purposes (Birnbaum et al., 2015) to healthcare (Schlegl et al., 2017), or industrial settings (Scime and Beuth, 2018; Haselmann et al., 2018; Christiansen et al., 2016) and as explained in the chapters before, robotics (Khalastchi et al., 2015; Wellhausen et al., 2020).

2.2.2 3D Anomaly Detection

While Image Anomaly Detection has been studied in different settings, 3D Anomaly Detection, due to the limitation to only MVTec 3D as the only representative dataset, is focused only on the topic of anomaly detection of industrial products.

One of the earlier studies on 3D anomaly detection, after the release of MVTec-3D, is from Horwitz et al (Horwitz and Hoshen, 2023). In their work, the authors study the task of 3D anomaly detection and segmentation (3DAD&S) and compare some non-purposely made models with their proposed approach on the MVTec dataset. Their objective is to better understand if, for this task, the 3D data is useful or not; from their results, it's clear that while 2D approaches still beat the 3D purposely built ones, at least for the latter the 3D data is essential. Then they provide an analysis of the key properties for successful 3DAD&S representation, leading to their proposed approach called BTF; while their model achieves very good segmentation performances, the authors recognize the model limitation on the image level accuracy, the same task we set to analyse in this work.

Rudolph et al. (Rudolph et al., 2023) propose an Asymmetric Student Teacher network to solve the Anomaly segmentation task on both the MVTec-3D and the original image-only MVTec dataset (Bergmann et al., 2021). Different to other student-teacher networks, their approach uses normalizing flow models for the teacher and conventional feed-forward convolution blocks for the students creating a discrepancy in the student prediction outside of the normal data on which the student network is trained on.

In their paper, Wang et al (Wang et al., 2023) use the MVTec-3D dataset to propose a multimodal approach to 3D anomaly detection called Multi-3D-Memory (M3DM). With M3DM the authors combine features extracted from both 3D point clouds and images; first, patches of point clouds are produced using the farthest point sampling, then the points in each patch are encoded using Point Transformer (Zhao et al., 2021) and the resulting features are remapped onto a 2D plane with the same size of the RGB picture and are then averaged into patch-wise features; at the same time patch features are extracted from the RGB image. Given the sets of patch features for both RGB and point cloud, the authors propose two new learnable modules called Unsupervised Feature Fusion and Decision Layer Fusion that are used, respectively, to learn the interaction between multimodal features and to deal with possible information loss that happens during the information fusion; the latter module uses multimodal memory banks during inference to produce the final anomaly and segmentation predictions.

Chu et al. (Chu et al., 2023), differently from others, propose a shape-guided approach for integrating the information from both RGB and point clouds. Their approach uses neural implicit functions to represent local areas of the point clouds. Similarly to others, they first split the point cloud in 3D patches, then these patches are passed to a PointNet network and the resulting features are used by the Neural Implicit Function module, to extract components that are used to define signed distance functions that implicitly encode normal local representations; these are then combined with ResNet extracted RGB features and used to define segmentation maps of the anomalies.

2.2.3 3D Feature Extractor

Image Based. A logical approach to 3D feature extraction is to use approaches well-tested on 2D data and adapt them to the additional dimension. In their paper (Zhang et al., 2022b), the authors propose to bridge the gap between a pre-trained CLIP (Radford et al., 2021) vision transformer and the point clouds from ModelNet (Wu et al., 2015) and ScanObjectNN (Uy et al., 2019) using point-projection images of different views of a single object. For each object, several views are generated and the resulting set of images is used to extract features. In their work, the authors note that using a zero-shot approach leads to poor performances, but with an additional trainable component after a few-shot training, the performance on the classification task increases.

Dong et al. (Dong et al., 2023) use pre-trained image vision transformers as part of the teacher encoder to then train a point cloud only student encoder module. Their approach, called ACT, uses only *x,y,z* information and achieves the best performance on both point cloud classification and semantic segmentation.

Point Cloud Based. In contrast to the aforementioned approaches, Zhang et al. (Zhang et al., 2022a) propose a point cloud only approach that does not use images or image pre-trained models as part of the pipeline. With Point-M2AE, the authors define and train a multi-scale masked autoencoder with the objective of using it as a zero-shot point cloud encoder. The model is composed of an encoder and decoder with skip connections; the encoder is fed with differently scaled masked point clouds from the same sample. As for images, the masked autoencoder is trained using a proxy reconstruction task. In the original paper, the approach achieves promising results across different tasks; finally, the authors provide both code and pre-trained models.

In this work, we set to use a pre-trained version of the encoder, used in combination with an SVM to solve a Linear classification task. The major drawback for Point-M2AE is the limited input size; the point clouds used for training the encoder are limited to 1024 points while the MVTec-3D point clouds contain hundreds of thousands of points per sample. For this reason, part of the code provided by Zhang et al. has been adapted and a point sampler has been introduced to reduce the MVTec-3D point cloud to the correct size.

3 EXPERIMENTAL SETUP

While the MVTec 3D dataset is built for 3D anomaly segmentation, in this work we will limit ourselves to the binary classification task of anomaly detection; for each point cloud our model will predict if it contains an anomaly or not. One of the metrics that is used in the dataset benchmark is the DETECTION-AUROC (in some cases called Image-AUROC); this metric indicates the AUC for detecting anomalies in the samples, without considering the segmentation. We will compare our results to those of the DETECTION-AUROC benchmark, available on the dataset page of *papers with code* [2].

In this work we use a two-part model, a feature extractor and an anomaly detection head; the latter receives as input the feature embedding from the extractor and produces an anomaly score for each embedding. The detection head is a Real-NVP model and - excluding adaptation to different embedding sizes - it is not changed throughout the experiments; thus the only changing part will be the feature extractor.

3.1 Real-NVP

This model has been already used in multiple recent papers (Wellhausen et al., 2020; Mantegazza et al., 2022; Mantegazza et al., 2023) as an anomaly detector based on latent embeddings; the Real-NVP (Dinh et al., 2016) is a kind of Normalizing Flow model, a deep learning model that learns a mapping between different spaces; in our approach, this is the only component that is trained. The features extracted are passed to a Real-NVP model; this model learns a mapping between the features embedding and a multivariate Gaussian distribution with identity matrix as covariance, 0 as mean, and the same number of dimensions as the embedding. We explore the Real-NVP hyper-parameters using an empirical process, ultimately landing on a similar setup to those used in the previous work (Mantegazza et al., 2022; Mantegazza et al., 2023); the model is composed of four coupling layers with a single hidden layer with the same size as the input vector with *odds* input masking, this applies for both the translationa and scaling modules. For more details on the specific components please refer to the original paper (Dinh et al., 2016). The only differences between these experiments and previous works are the internal size of the layers and input size; these are input-dependent. Note that during training or hyper-parameter search, the Real-NVP always converged with the mapping, excluding its influence on the experiment results. All Real-NVPs are trained for 100 epochs with early stopping, randomly initialized weights, a starting learning rate of 0.001 and Adam (Kingma and Ba, 2014) optimizer.

3.2 Baselines

We define two baselines, *Random* and *Ones*. The first substitutes the features extractor component with a random signal sampled from a Normal distribution; the latter produces a 1s feature vector as input for the Real-NVP. We use two different baselines to demonstrate that the Real-NVP component is not relevant to our experiment.

Handcrafted Baseline. We also define a set of handcrafted feature extractors to serve as an additional baseline. These basic features are heuristics chosen to be easy to compute and informative enough to detect macroscopic anomalies (e.g. a large piece of an object missing). The 11 features are the following:

- ft1: number of points in the point cloud
- ft2 to ft5: number of points in 4 quadrants (i.e. split the point cloud into 4 quadrants from a top view)
- ft6 and ft7: maximum and minimum z value for any point cloud's points
- ft8 and ft9: maximum and minimum x value for any point cloud's points
- ft10 and ft11: maximum and minimum y value for any point cloud's points

3.3 XYZ Model

The first, non-baseline, approach proposed uses Point-M2AE as a feature extractor. This XYZ (i.e. point cloud data) encoder, uses positional information from the entire point cloud to produce a feature vector. The Point-M2AE encoder takes as input a 1024pts point cloud and produces 384 features; these are passed to the Real-NVP that maps them to a latent space where the ”normality” probability can be extracted.

3.4 RGB Model

Since the MVTec-3D dataset provides RGB images of the sample scans, we took the CLIP+Real-NVP model from our previous work (Mantegazza et al., 2023) on image anomaly detection, and we used it to identify anomalies in the dataset. Notice that, while the RGB images are more informative for some specific anomalies (see the anomaly color for the object foam), the overall information provided to the Real-NVP is more limited than total information in a point cloud.

This setup uses the Vision Transformer(ViT) module of CLIP (Radford et al., 2021); the ViT takes the RGB image as input and produces a 512-sized feature vector. As for the previous approaches, the vector is then passed to the Real-NVP component.

[2] https://paperswithcode.com/sota/rgb-3d-anomaly-detection-and-segmentation-on

3.5 RGB+XYZ Model

Finally, we test an RGB+XYZ model by simply concatenating the 512 RGB-derived features and the 384 XYZ-derived features in a single vector for each sample.

Even in this case with an 896 size vector, the Real-NVP correctly converged and learned a mapping.

4 RESULTS

We report all the AUC results of the 46 runs (excluding the hyper-parameter searches) in Tables 3,4,5,6 and 2. The results are color-coded, any value of AUC equal to or lower than 0.5 is colored red; higher values shift from red to yellow and towards green, which is the color for values near or equal to 1.

In each table, we report the performances split by anomaly type and object class, with the addition of the AUC considering the whole test set as a binary problem, and the averaged AUC, built by averaging the AUC of each object class per se.

As detailed by the tables, for all models except *random* and *ones*, we also consider the AUC for the models trained and tested on samples from a single object class; for example, all lines with *bagle* as object class, represent models that during training, validation and testing, only saw bagels. To the best of our knowledge, we are the first to introduce this kind of experiment for this dataset; our motivation is to study the effects of each object class's characteristics (shape and color) on each specific model (and thus features extractor) performance.

The best-performing model is the RGB, CLIP-based one followed closely by the RGB+XYZ one. The RGB model, using all objects, achieved an AUC of 0.69 for the test set. We acknowledge that this performance is surpassed by other, more complex, models benchmarked on the MVTec-3D dataset, but are nonetheless better than the baselines and handcrafted. Moreover, both the RGB and RGB+XYZ beat the performance of the purposely built approaches proposed in (Bergmann. et al., 2022), namely Voxel VM, Voxel AE and Voxel GAN.

5 CONCLUSION

In this work, we compared different off-the-shelf pre-trained feature extractors combined with a Real-NVP model to solve the task of 3D anomaly detection on the MVTec-3D dataset.

Table 1: Comparing our approaches to the MVTec-3D benchmark.

Shape-Guided (Chu et al., 2023)	0.95
M3DM (Wang et al., 2023)	0.95
AST (Rudolph et al., 2023)	0.94
Back To Feat. (Horwitz and Hoshen, 2023)	0.87
RGB (Ours)	*0.69*
RGB+XYZ (Ours)	*0.67*
Voxel VM (Bergmann. et al., 2022)	0.61
Voxel AE (Bergmann. et al., 2022)	0.54
XYZ (Ours)	*0.53*
Voxel GAN (Bergmann. et al., 2022)	0.52

From our experiments, it is clear that all approaches tested while better than the baselines are not sufficient for an anomaly detection task. We impute the limited performances to the features extractors; while performing excellently on their original tasks, the models available are still too limited to solve this task; for example, the XYZ Point-M2AE extractor is strongly limited by the number of points it accepts in input and thus losing important local details that might help to detect small anomalies. Future work in this direction would be to either this Point-M2AE on 3D patches of the point cloud or to find an alternative model that can accept more points than Point-M2AE. In addition, we think that the dataset is too small; this implies that Real-NVP, while correctly converging, fails to learn a mapping of the normal samples to correctly identify anomalies.

With our results, we demonstrate that a combination of a dataset limitation and additional complexity when dealing with point clouds versus images leads to a lack of *off-the-shelf* models for solving complex 3D vision tasks. This is remarked by the striking difference in performance when comparing our approach to the ad-hoc solutions.

We think that the research work for 3D anomaly detection is just at the beginning; more models and datasets are needed in order to achieve a performance and ease of use comparable to the one for 2D vision. Finally, while we understand the complexity of collecting and labelling a dataset for 3D Anomaly Detection, we strongly encourage future works in this direction.

REFERENCES

Bergmann, P. et al. (2021). The mvtec anomaly detection dataset: a comprehensive real-world dataset for unsupervised anomaly detection. *Int. Journal of Computer Vision*, 129:1038–1059.

Bergmann., P., Jin., X., Sattlegger., D., and Steger., C. (2022). The mvtec 3d-ad dataset for unsupervised 3d

Table 2: AUC of the Baseline models.

Baseline	Obj seen	Bent	Color	Comb.	Contam.	Crack	Cut	Hole	Open	Thread	Test	Test Avg
random	**all**	**0.49**	**0.53**	**0.48**	**0.50**	**0.50**	**0.51**	**0.51**	**0.55**	**0.56**	**0.50**	**0.51**
ones	**all**	**0.50**	**0.50**	**0.50**	**0.50**	**0.50**	**0.50**	**0.50**	**0.50**	**0.50**	**0.50**	**0.50**

Table 3: AUC of the XYZ model.

	Bent	Color	Comb.	Contam.	Crack	Cut	Hole	Open	Thread	Test	Test Avg
all	**0.32**	**0.96**	**0.51**	**0.51**	**0.58**	**0.52**	**0.55**	**0.61**	**0.68**	**0.53**	**0.58**
bagel			0.71	0.67	0.71		0.68			0.69	0.69
cable gland	0.65					0.44	0.56		0.56	0.55	0.55
carrot			0.56	0.53	0.65	0.65	0.59			0.60	0.60
cookie			0.70	0.47	0.58		0.48			0.56	0.56
dowel	0.48		0.44	0.45		0.50				0.47	0.47
foam		0.48	0.65	0.49		0.54				0.54	0.54
peach			0.46	0.49		0.41	0.46			0.46	0.46
potato			0.43	0.47		0.33	0.40			0.41	0.41
rope				0.63		0.66		0.96		0.72	0.75
tire			0.30	0.49		0.50	0.46			0.48	0.44

Table 4: AUC of the RGB model.

	Bent	Color	Comb.	Contam.	Crack	Cut	Hole	Open	Thread	Test	Test Avg
all	**0.63**	**0.99**	**0.78**	**0.64**	**0.83**	**0.59**	**0.72**	**0.53**	**0.61**	**0.69**	**0.70**
bagel			0.92	0.63	1.00		0.79			0.84	0.83
cable gland	0.65					0.73	0.68		0.62	0.67	0.67
carrot			0.77	0.68	0.71	0.59	0.72			0.69	0.69
cookie			0.62	0.56	0.77		0.52			0.62	0.62
dowel	0.87		0.91	0.77		0.72				0.82	0.82
foam		1.00	0.82	0.70		0.75				0.82	0.82
peach			0.60	0.56		0.66	0.49			0.57	0.58
potato			0.60	0.59		0.43	0.41			0.51	0.51
rope				0.63		0.63		0.87		0.69	0.71
tire			0.48	0.54		0.52	0.59			0.55	0.53

Table 5: AUC of the XYZ+RGB model.

	Bent	Color	Comb.	Contam.	Crack	Cut	Hole	Open	Thread	Test	Test Avg
all	**0.53**	**1.00**	**0.75**	**0.63**	**0.81**	**0.60**	**0.66**	**0.62**	**0.61**	**0.67**	**0.69**
bagel			0.93	0.67	0.99		0.81			0.85	0.85
cable gland	0.74					0.69	0.73		0.69	0.71	0.72
carrot			0.76	0.72	0.70	0.62	0.73			0.70	0.70
cookie			0.67	0.58	0.84		0.58			0.67	0.67
dowel	0.83		0.89	0.76		0.63				0.78	0.78
foam		1.00	0.84	0.67		0.63				0.78	0.78
peach			0.57	0.59		0.60	0.49			0.56	0.56
potato			0.60	0.55		0.43	0.37			0.49	0.49
rope				0.62		0.67		0.87		0.70	0.72
tire			0.30	0.54		0.53	0.55			0.52	0.48

Table 6: AUC of the Handcrafted features model.

	Bent	Color	Comb.	Contam.	Crack	Cut	Hole	Open	Thread	Test	Test
											Avg
all	**0.35**	**0.78**	**0.53**	**0.51**	**0.58**	**0.48**	**0.48**	**0.30**	**0.13**	**0.49**	**0.46**
bagel			0.52	0.44	0.48		0.48			0.48	0.48
cable gland	0.47					0.49	0.64		0.43	0.51	0.51
carrot			0.51	0.42	0.46	0.49	0.35			0.45	0.45
cookie			0.61	0.58	0.67		0.73			0.65	0.65
dowel	0.59		0.52	0.64		0.51				0.57	0.56
foam		0.67	0.60	0.65		0.69				0.65	0.65
peach			0.52	0.60		0.48	0.48			0.52	0.52
potato			0.00	0.00		0.00	0.00			0.00	0.00
rope				0.60		0.59		0.69		0.62	0.62
tire			0.60	0.48		0.58	0.77			0.61	0.61

anomaly detection and localization. In *Proceedings of the 17th International Joint Conference on Computer Vision, Imaging and Computer Graphics Theory and Applications - Volume 5: VISAPP,*, pages 202–213. INSTICC, SciTePress.

Birnbaum, Z. et al. (2015). Unmanned aerial vehicle security using behavioral profiling. In *2015 International Conference on Unmanned Aircraft Systems (ICUAS)*, pages 1310–1319.

Chandola, V., Banerjee, A., and Kumar, V. (2009). Anomaly detection: A survey. *ACM Comput. Surv.*, 41(3).

Christiansen, P. et al. (2016). Deepanomaly: Combining background subtraction and deep learning for detecting obstacles and anomalies in an agricultural field. *Sensors*, 16(11):1904.

Chu, Y.-M., Liu, C., Hsieh, T.-I., Chen, H.-T., and Liu, T.-L. (2023). Shape-guided dual-memory learning for 3D anomaly detection. In Krause, A., Brunskill, E., Cho, K., Engelhardt, B., Sabato, S., and Scarlett, J., editors, *Proceedings of the 40th International Conference on Machine Learning*, volume 202 of *Proceedings of Machine Learning Research*, pages 6185–6194. PMLR.

Dinh, L., Sohl-Dickstein, J., and Bengio, S. (2016). Density estimation using real nvp.

Dong, R., Qi, Z., Zhang, L., Zhang, J., Sun, J., Ge, Z., Yi, L., and Ma, K. (2023). Autoencoders as cross-modal teachers: Can pretrained 2d image transformers help 3d representation learning? In *The Eleventh International Conference on Learning Representations*.

Dosovitskiy, A., Beyer, L., Kolesnikov, A., Weissenborn, D., Zhai, X., Unterthiner, T., Dehghani, M., Minderer, M., Heigold, G., Gelly, S., Uszkoreit, J., and Houlsby, N. (2021). An Image is Worth 16x16 Words: Transformers for Image Recognition at Scale. arXiv:2010.11929 [cs].

Floris, A., Frittoli, L., Carrera, D., and Boracchi, G. (2022). Composite layers for deep anomaly detection on 3d point clouds.

Frittoli, L. (2022). *ADVANCED LEARNING METHODS FOR ANOMALY DETECTION IN MULTIVARIATE DATASTREAMS AND POINT CLOUDS*. PhD thesis, Politecnico Milano.

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

Horwitz, E. and Hoshen, Y. (2023). Back to the Feature: Classical 3D Features Are (Almost) All You Need for 3D Anomaly Detection. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 2967–2976.

Khalastchi, E., Kalech, M., Kaminka, G. A., and Lin, R. (2015). Online data-driven anomaly detection in autonomous robots. *Knowledge and Information Systems*, 43(3):657–688.

Kingma, D. P. and Ba, J. (2014). Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*.

Mantegazza, D., Giusti, A., Gambardella, L. M., and Guzzi, J. (2022). An outlier exposure approach to improve visual anomaly detection performance for mobile robots. *IEEE Robotics and Automation Letters*, pages 1–8.

Mantegazza, D., Xhyra, A., Giusti, A., and Guzzi, J. (2023). Active Anomaly Detection for Autonomous Robots: A Benchmark. In Iida, F., Maiolino, P., Abdulali, A., and Wang, M., editors, *Towards Autonomous Robotic Systems*, Lecture Notes in Computer Science, pages 315–327, Cham. Springer Nature Switzerland.

Masuda, M., Hachiuma, R., Fujii, R., Saito, H., and Sekikawa, Y. (2021). Toward unsupervised 3d point cloud anomaly detection using variational autoencoder. In *2021 IEEE International Conference on Image Processing (ICIP)*, pages 3118–3122.

Radford, A., Kim, J. W., Hallacy, C., Ramesh, A., Goh, G., Agarwal, S., Sastry, G., Askell, A., Mishkin, P., Clark, J., Krueger, G., and Sutskever, I. (2021). Learning Transferable Visual Models From Natural Language Supervision. arXiv:2103.00020 [cs].

Rudolph, M., Wehrbein, T., Rosenhahn, B., and Wandt, B. (2023). Asymmetric Student-Teacher Networks for Industrial Anomaly Detection. In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision (WACV)*, pages 2592–2602.

Ruff, L. et al. (2021). A unifying review of deep and shallow anomaly detection. *Proceedings of the IEEE*, 109(5):756–795.

Schlegl, T. et al. (2017). Unsupervised anomaly detection with generative adversarial networks to guide marker discovery. In *Information Processing in Medical Imaging*, pages 146–157, Cham.

Scime, L. and Beuth, J. (2018). A multi-scale convolutional neural network for autonomous anomaly detection and classification in a laser powder bed fusion additive manufacturing process. *Additive Manufacturing*, 24:273–286.

Uy, M. A., Pham, Q.-H., Hua, B.-S., Nguyen, T., and Yeung, S.-K. (2019). Revisiting point cloud classification: A new benchmark dataset and classification model on real-world data. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*.

Wang, Y., Peng, J., Zhang, J., Yi, R., Wang, Y., and Wang, C. (2023). Multimodal Industrial Anomaly Detection via Hybrid Fusion. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 8032–8041.

Wellhausen, L., Ranftl, R., and Hutter, M. (2020). Safe robot navigation via multi-modal anomaly detection. *IEEE Robotics and Automation Letters*, 5(2):1326–1333.

Wu, Z., Song, S., Khosla, A., Yu, F., Zhang, L., Tang, X., and Xiao, J. (2015). 3d shapenets: A deep representation for volumetric shapes. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 1912–1920.

Zhang, R., Guo, Z., Gao, P., Fang, R., Zhao, B., Wang, D., Qiao, Y., and Li, H. (2022a). Point-M2AE: Multi-scale Masked Autoencoders for Hierarchical Point Cloud Pre-training. In Koyejo, S., Mohamed, S., Agarwal, A., Belgrave, D., Cho, K., and Oh, A., editors, *Advances in Neural Information Processing Systems*, volume 35, pages 27061–27074. Curran Associates, Inc.

Zhang, R., Guo, Z., Zhang, W., Li, K., Miao, X., Cui, B., Qiao, Y., Gao, P., and Li, H. (2022b). Pointclip: Point cloud understanding by clip. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 8552–8562.

Zhao, H., Jiang, L., Jia, J., Torr, P., and Koltun, V. (2021). Point Transformer. arXiv:2012.09164 [cs].