ANOMALY DETECTION USING SPECTROGRAM RECONSTRUCTION ERRORS WITH U-NET

Technical Report

David Hauser, Tobias Katsch, Sara Moosbauer

Students at Johannes Kepler University, Linz, Austria

ABSTRACT

In this report we describe our submission to the DCASE 2023 Task 2: First-Shot Unsupervised Anomalous Sound Detection Challenge, which has the goal of detecting malfunctioning machines by analyzing a machine's sound recording. [1]. We applied the U-Net architecture [2], trained to reconstruct partially masked spectrograms generated from the machine sound recordings. The task turned out to be challenging, beating the baseline on one out of seven machines during evaluation.

Index Terms— anomalous sound detection, machine sound classification, audio signal processing

1. INTRODUCTION

The project covered by this technical report was conducted during a Machine Learning and Audio class at which Johannes Kepler University our team participated. In the following, we describe our system for DCASE 2023 Task 2: First-Shot Unsupervised Anomalous Sound Detection targeted at identifying malfunctioning machines from audio recordings.

One of the challenges of anomaly detection is that anomalous data is often scarce or unavailable, making it difficult to train supervised models that can generalize well to previously unseen anomalies. Therefore, unsupervised methods, such as learning a data distribution from normal sounds, are often preferred, as they do not require labeled data for training. However, unsupervised classifiers often suffer from low true positive or high false positive rates. This is especially the case when the model has to deal with noisy data, making this a very challenging task.

One major difference from DCASE 2022 Task 2 is that the set of machine types is completely different between the development dataset and the evaluation dataset. Hence, the model architecture and hyper-parameters cannot be tailored to a specific machine type, making the task even more challenging than in previous years.

2. APPROACH

Our approach is inspired by Yamashita et al. [3], who previously showed promising results for a similar task with a U-Net [2] architecture. U-Net consists of CNN encoder and decoder layers with skip connections. We use 64x64 log mel-spectrograms of provided audio samples as input. During training, random patches in the input spectrogram are masked, and the model learns to reconstruct those missing parts. To derive an anomaly score at inference time, we average the reconstruction error over 256 randomly generated masks. Since the model is only trained on normal audio, the reconstruction error for anomalous samples is higher. Hence we use it as an anomaly score.

Our approach is based on the assumption that normal sounds can be reconstructed more accurately than anomalous sounds by a model trained only on normal sounds. We use a U-Net architecture because it has been shown to be effective for image inpainting tasks, where parts of an image are missing or corrupted. We apply a similar technique to audio spectrograms, where we randomly mask patches of the input spectrogram and feed it to the U-Net model to reconstruct the masked parts of a spectrogram.

2.1. Training

During development, we trained the model on the normal audio samples from the provided development dataset, which are divided into seven machine types: fan, gearbox, bearing, slide rail, toy car, toy train, and valve. We used the training setup described in Table-1 to train one model for each machine type and evaluated them on the provided evaluation data set, which includes labeled anomalous and normal audio samples.

We convert each audio sample into a 64 x 64 log mel-spectrogram. We then randomly mask 48 out of 64 patches of size 8x8 in the spectrogram and use it as the input to the U-Net model. The model tries to reconstruct the masked areas of the original spectrogram by minimizing the mean squared error (MSE) loss. At each epoch, we used fresh, randomly generated masks.

Table 1: Training Setup

Parameter	**Value**
Learning rate	0.0001
Optimizer	Adam
Epochs	100
Batch size	64

2.2. Evaluation

To test the model, we use the same masking procedure as in training and give the masked spectrogram as input to the model. We then average the output spectrograms over 256 inference steps to obtain a more stable reconstruction. We compute the MSE between the masked areas of the reconstructed spectrogram and the masked areas of the original spectrogram. The MSE acts as our anomaly score, as a high MSE is hypothesized to indicate an increased likelihood of a sample being anomalous. The challenge uses area under the curve (AUC) and partial area under the curve (pAUC) as evaluation metrics to measure how well the model can distinguish between normal and anomalous sounds. The pAUC focuses on a specific range of low false-positive rates (FPR) from 0 to a given threshold value p (0.1 in our case). The pAUC is introduced to address practical concerns, as a reliable ASD system should have a high true-positive rate while keeping the FPR low. This ensures that the system effectively detects anomalies without generating frequent false alarms. For more detailed descriptions of AUC and pAUC we refer to [1]

3. DATASET

3.1. Raw Data

We use the development data set provided by the DCASE challenge organizers. It consists of seven different machine types: Fan, Gearbox, Bearing, Slide rail, Toy car, Toy train, and Valve. The data set is split into training data consisting of 990 normal clips from the source domain and 10 normal clips from the target domain for each machine. The source and target domain differ in operating speed, machine load, viscosity, heating temperature, type of environmental noise, signal-to-noise ratio, etc. The test data for each machine consists of 50 normal clips and 50 anomalous clips. Each clip has a duration varying between 6 to 18 seconds.

3.2. Preprocessing

Before training, we transform each audio sample to a log mel-spectrogram with a shape of (64,64). Log mel-spectrograms are preferred over log spectrograms because they balance the signal strength across different frequency ranges. Log spectrograms, on the other hand, have weak signals in the mid to low frequencies and only strong signals in the very high frequencies. We suspect that differences in the low to mid-level frequencies between normal and anomalous samples are higher when using the Mel scale, making it more difficult for the model to reconstruct anomalous samples after only being trained on normal samples and thus leading to an improved anomaly score.

The parameters used to calculate the spectrograms are given in Table-2. They are chosen such that the spectrograms of the audio samples have a shape of (64, 64),

Table 2: Preprocessing Parameters

Parameter	Value
Hop Length	256
Window length	124
FFT lenght	400
Mel bins	80
Spectrogram Shape	64 x 64
Patch size	8 x 8
Patches masked	48

4. MODEL ARCHITECTURE

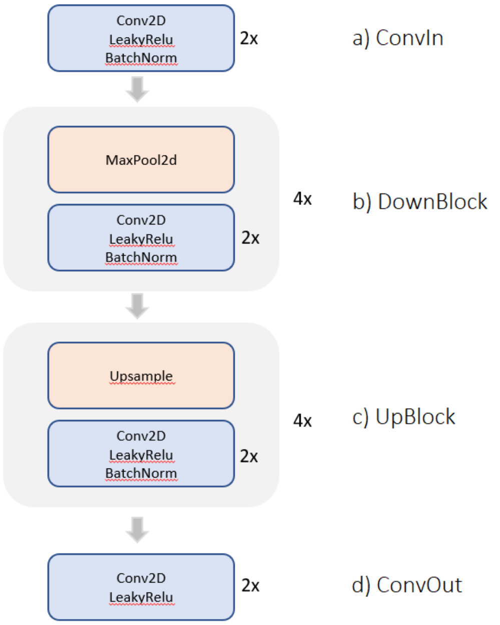

Figure 1: U-Net Model architecture for our submission

We use a U-Net architecture for our model. The U-Net architecture is a convolutional neural network (CNN) that was originally designed for biomedical image segmentation. It has since been used in many other applications, including audio processing. In particular, Yamashita et al. [3] used the U-Net architecture in their submission for DCASE 2022 Task 2 and we decided to take a similar approach.

Our implementation of the U-Net architecture consists of an input layer followed by four down-sampling blocks and four up-sampling blocks, as described in 1. Each down-sampling block consists of two convolutional layers with batch normalization and ReLU activation functions, followed by max pooling. This results in a compressed latent space size of (4,4) with 512 channels, as shown in Table-3. Each up-sampling block consists of an up-sampling layer with scale factor 2, followed by two convolutional layers with batch normalization and ReLU activation functions. The output layer consists of two 2D convolutional layers with Leaky ReLU activation functions. For all convolutional layers, we use a kernel size of 3 and padding of 1.

Table 3: A U-Net architecture

Input	Module	Kernel size	N	Output
(1,64,64)	ConvIn	3	2	(64,64,64)
(64,64,64)	DownBlock	3	4	(512,4,4)
(512,4,4)	UpBlock	3	4	(64,64,64)
(64,64,64)	ConvOut	3	2	(1,64,64)

5. RESULTS

Table 4: Performance Metrics Comparison: Baseline vs. U-net

Machine	Baseline		U-net	
	AUC	pAUC	AUC	pAUC
Slider	**70.31%**	**56.37%**	52.08%	51.57%
ToyCar	**70.1%**	**52.47%**	46.18%	48.91%
Gearbox	**60.31%**	**53.22%**	43.10%	49.76%
Valve	55.35%	51.18%	**64.38%**	**58.85%**
ToyTrain	**57.93%**	48.57%	49.33%	**49.53%**
Bearing	**65.92%**	**50.42%**	40.02%	49.34%
Fan	**80.19%**	**59.04%**	48.18%	58.11%

Table-4 shows a comparison based on the U-net approach with the parameters from tables 1, 2, 3 and an auto-encoder baseline from the challenge organizers [1]. Our results suggest that for all experimental setups, the reconstruction loss of the U-net architecture does not provide a reliable estimation for a given machine's normality (or anomaly, respectively). Our method beats the baseline only on one model (valve) for both AUC and pAUC.

6. FURTHER EXPERIMENTS

We conducted various experiments with varying hyperparameters and reconstruction schemes to investigate the reasons for our poor results.

- **Learning Rate:** Gradual steps between 1e-2 to 1e-5
- **Training Batch Sizes:** Ranging from 32 to 256
- **Epochs:** 10 to 300
- **Number of Masked Patches:** Varying from masking only a few patches to masking almost all patches
- **Patch Sizes:** Sizes ranging from 2 to 16
- **Frequency Scaling of Spectrogram Generation:** Log and log-mel
- **Reconstruction Mode:**
  a) Reconstructing the full spectrogram
  b) Reconstructing only the masked spectrogram areas

Varying the learning rate, batch size, number of masked patches, or patch size does not yield reliable improvement but the suggestion from our supervisor to use reconstruction approach b) instead of a) lead to a small but consistent improvement, as the following table shows. Here, all other parameters are kept the same, as reported in tables 1, 2, 3.

Table 5: Performance Metrics Comparison: a) vs. b)

Machine	a) Reconst. full		b) Reconst. masked	
	AUC	pAUC	AUC	pAUC
Slider	50.38%	51.25%	**52.08%**	**51.57%**
ToyCar	45.21%	**49.12%**	**46.18%**	48.91%
Gearbox	42.54%	49.49%	**43.10%**	**49.76%**
Valve	62.55%	57.26%	**64.38%**	**58.85%**
ToyTrain	48.46%	**49.81%**	**49.33%**	49.53%
Bearing	39.45%	49.29%	**40.02%**	**49.34%**
Fan	51.44%	59.12%	**52.18%**	**58.11%**

7. CONCLUSION

The task of Anomalous Sound Detection for DCASE 2023, employing the U-Net architecture, was challenging. Our method based on the U-net architecture surpassed the baseline only for the valve machine type. Considering the goal of domain generalization, i.e., training a model for a completely new machine with only normal sounds available, these results are particularly concerning. For some machines, e.g., for fan, the model even turns out significantly worse than average. As our method yields sub-average AUC scores for 5/7 machine types, we assume that the difficulty of reconstructing normal vs. reconstructing anomalous sounds using our U-net reconstruction approach varies greatly from machine type to machine type. Therefore our method can not be recommended for training a model on a completely new machine but may be useful solely for potentially detecting malfunctioning valves. Although a lot of effort was made to uncover potential reasons for these unsatisfactory results, we can not present definite answers. Despite the challenging nature of the task and many revisions of the experimental setup, we have to seriously consider potential mistakes in our preprocessing, training, or evaluation setup as a potential explanation. Overall, working on this project was still a great opportunity to learn about the challenges of dealing with unsupervised learning and to handle setbacks during projects.

8. REFERENCES

[1] [Online]. Available: https://dcase.community/challenge2023/task-first-shot-unsupervised-anomalous-sound-detection-for-machine-condition-monitoring

[2] O. Ronneberger, P. Fischer, and T. Brox, "U-net: Convolutional networks for biomedical image segmentation," 2015, cite arxiv:1505.04597Comment: conditionally accepted at MICCAI 2015. [Online]. Available: http://arxiv.org/abs/1505.04597

[3] J. Yamashita, R. Tanaka, K. Ikeda, S. A. S. Hayamizu, and S. Tamura, "Anomaly detection using autoencoder, idnn and u-net using ensemble, in detection and classification of acoustic scenes and events 2022 challenge," 2022.