Article

Anomaly Detection in Particulate Matter Sensor Using Hypothesis Pruning Generative Adversarial Network

YeongHyeon Park [1,*], Won Seok Park [1], and Yeong Beom Kim[1]

[1] SK Planet Co.,Ltd., Seongnam, 13487, Korea
* Correspondence: yeonghyeon@sk.com

Abstract: World Health Organization (WHO) provides the guideline for managing the Particulate Matter (PM) level because when the PM level is higher, it threats the human health. For managing PM level, the procedure for measuring PM value is needed firstly. We use Tapered Element Oscillating Microbalance (TEOM)-based PM measuring sensors because it shows higher cost-effectiveness than Beta Attenuation Monitor (BAM)-based sensor. However, TEOM-based sensor has higher probability of malfunctioning than BAM-based sensor. In this paper, we call the overall malfunction as an anomaly, and we aim to detect anomalies for the maintenance of PM measuring sensors. We propose a novel architecture for solving the above aim that named as Hypothesis Pruning Generative Adversarial Network (HP-GAN). We experimentally compare the several anomaly detection architectures to certify ours performing better.

Keywords: anomaly detection; generative adversarial network; multiple hypothesis; particulate matter

1. Introduction

World Health Organization (WHO) recommends to managing the provides Particulate Matter (PM) level with providing their guideline as shown in Table 1 because it can infiltrate into the deeper site of humans via respiratory organ [1]. For detail, the PM can trigger not only respiratory diseases [2] but also cardiovascular disease [3], lung cancer [4], or some other diseases.

For managing PM level with referring Table 1, the process for measuring the air condition should be preceded. In the Republic of Korea, the Air Korea, operated by Korea Environment Corporation, provides measured values of SO_2, CO, O_3, NO_2, $PM_{2.5}$ and PM_{10} with unit of one hour. For informing the air pollution to the public, the higher spatial resolution may be more effective than lower. However, the resolution provided by them is relatively low because of the cost of maintaining the high-end PM measuring sensor.

Two types of sensors can be used for PM measurement, each based on Beta Attenuation Monitor (BAM) and Tapered Element Oscillating Microbalance (TEOM) methods [5,6]. The characteristic of the BAM-based sensor is higher precision of measurement, but it needs high maintenance costs as in the case of Air Korea. On the other hand, the TEOM-based sensor needs a lower cost than the BAM-based sensor. For the above reason, when using a BAM-based sensor, increasing the spatial resolution it is difficult but it can be eased via using a cost-effective TEOM-based sensor.

The two coefficients, one of them is Pearson's correlation coefficient and the other is a coefficient of determination for 1-hour averages, are used as proof for the cost-effectiveness of TEOM-based sensor. Those coefficients are already measured as 0.91 and 0.81 respectively [5].

The TEOM-based sensor can alternate the BAM-based sensor to monitor the PM level. Thus, We have installed TEOM-based sensors relatively densely in several regions as a trial. However, the limitation of the TEOM-based sensor is that it has a more probability malfunctioning in the measuring

Preprints (www.preprints.org) | NOT PEER-REVIEWED | Posted: 14 February 2020 doi:10.20944/preprints202001.0028.v2

Table 1. Air Quality Guideline (AQG) for Particulate Matter (PM) level management from World Health Organization (WHO) [1]. The unit of each value is $\mu g/m^3$. When the diameter of PM is $2.5\mu g/m^3$ it called $PM_{2.5}$, and when it between $2.5\mu g/m^3$ and $10\mu g/m^3$, called PM_{10}.

	Annual mean	**24-hour mean**
$PM_{2.5}$	10	25
PM_{10}	20	50

process than the BAM-based sensor because it more affected by the external environment; we call overall malfunctions as an anomaly. Thus, we propose the novel architecture for anomaly detection to maintain a TEOM-based sensor. Because, if an anomaly detecting solution is provided with the TEOM-based sensor, the efficiency of the PM level monitoring cost can be made more effective.

The organization of this paper is described as follows. In Section 2, we summarize previous studies that have efforted for anomaly detection. We present the proposed architecture and experimental results in Section 3 and Section 4 respectively, and conclude the whole content in final section. We only deal with anomaly detection task of functioning sensors in this paper. Thus, the task after anomaly detection, can be categorized to correcting the collected values or maintain the sensor via engineer, will be handled in future study.

2. Related Works

Several previous studies have conducted the anomaly detection via classification approach with various methods [7–11]. However, there are some problem for using classification method such as difficulty of collecting diverse abnormal cases and labeling cost from collected data to specific category.

For detecting anomaly, the above problems can be eased by regression-based method such as One-Class SVM [12], Auto-Encoder (AE) [13,14], or Long-Short Term Memory (LSTM) [15,16]. The idea of regression-based method is such simple. Also, the cost of data preparing to use regression-based machine learning or deep learning-based anomaly detection algorithm is not high. Because we need only the normal case (healthy) data for training, and it does not need the categorization process. Moreover, the labeling task is needed for measuring performance quantitatively, but it is quite simple in this case because each sample needs only checked whether normal or not.

Recently, there are several neural network-based anomaly detection models are published [17–23]. The generative neural network which trained with variational bound can make the user desired data from the random noise [17,24]. However, generating data from noise as same as they do is not essential in anomaly detection. Moreover, because of using variational bound with distribution assumption, the above deep learning architectures may generate the blurred data.

The Generative Adversarial Network (GAN)-based anomaly detection model is published as named as AnoGAN [18]. The AnoGAN can generate more sharped data than the variational bound-based model. However, it needs to find the most close generated sample from noise with input data for determining abnormal or not. Because of the above procedure, the throughput of AnoGAN is lower than the typical AE-based architecture.

Some of the recent research, such as BiGAN [19] and GANomaly [20], eased the limitation of AnoGAN. These models directly generate the sample via input data and use them for anomaly detection. BiGAN [19] has the limitation of generating high-resolution samples because encoder and decoder share the parameters. However, GANomaly [20] eases the above limitation via simply using separated parameters for encoder and decoder respectively.

One other research tried to generate data consistently with avoiding to generate the blurred sample [23]. For achieving their purpose, they apply the multiple hypothesis to the last layer of the generator. Their neural network generates multiple samples and the best sample that close to input data among those is selected. However, one of the unsatisfied things is that they still use the variational

Preprints (www.preprints.org) | NOT PEER-REVIEWED | Posted: 14 February 2020 doi:10.20944/preprints202001.0028.v2

bound as same as Variational Auto-Encoder (VAE) [24] for training. Thus it can be regarded that has still restricted for generating clear, not blurred, data.

The preprocessing is one of the additional considerations to construct an anomaly detection system for reducing the computational cost. For example, data dimension reduction via preprocessing can be highly reduce the computational cost [25–29]. However, our PM sensor collects the value at every one hour, so the preprocessing is not needed to reduce computational cost such as time.

3. Proposed Architecture

In this section, we present the proposed architecture for anomaly detection in the PM sensor. We compare the property of two kinds of convolution method firstly. Then, we present the structure of the neural network that we propose. Finally, we describe the anomaly detection procedure.

3.1. Reason for Using 1D Convolution

When using the image data as the input, the 2 dimensional (2D) convolutional layer can be used for constructing the neural network typically. However, we use the 1 dimensional (1D) convolutional layer because our data is 1D signal data.

The signal data can have the multiple channel. For example, $PM_{2.5}$ and PM_{10} can represent first and second channel respectively. The 2D convolution can be used for processing the multi-channel signal data, but it is not efficient as shown in Figure 1.

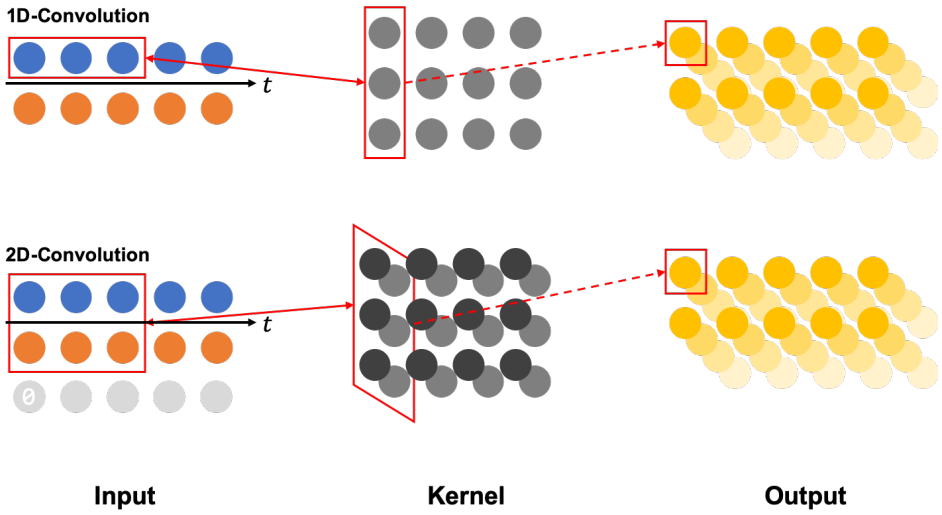

Figure 1. The examples of generating the output via 1D and 2D convolution. In 2D convolution case, the zero-padding method is needed for aggregating time information with maintaining feature dimension.

In order to minimize the information loss for each channel, our purpose is only aggregating the time information via a convolutional layer like a recurrent neural network [30]. In the case of 1D convolution, time information can be summarized while maintaining channel information in natural. However, if the neural network constructed with 2D convolution, the channel information of the input data will be reduced unintentionally. For avoiding this problem the zero-padding can be used as shown in Figure 2, for maintaining the channel dimension, but the last channel of the generated output probably has less information than the front channels. Thus, we adopt the 1D convolutional layer for constructing the neural network.

3.2. Multiple Hypothesis-based Architecture

We have already summarized related works in Section 2. UUsing the variational bound with distribution assumption causes generating blurred the sample. Also, the AnoGAN-like architecture

Preprints (www.preprints.org) | NOT PEER-REVIEWED | Posted: 14 February 2020 doi:10.20944/preprints202001.0028.v2

has lower throughput because it needs a matching procedure for finding the most closer generated sample to input data. The helpful thing is multiple hypothesis that can produce more relevant results and can work more robustly [31–33].

We construct the neural network with reference to the concept of GANomaly as a backbone architecture because it can ease the several mentioned limitations of previous anomaly detection models. We also apply the multiple hypothesis method in the last layer of the generator. Because the multiple hypothesis can utilize to maintain the quality of the output consistently via selecting the best output. We name the procedure of selecting the best output as hypothesis pruning.

The multiple hypothesis generates several samples from the input data directly and generates one additional sample from the random noise. Each generated sample is used as a hypothesis respectively but the samples that are not in the best case will be used as a regularization term. The overall architecture of the neural network that we proposed is shown in Figure 2.

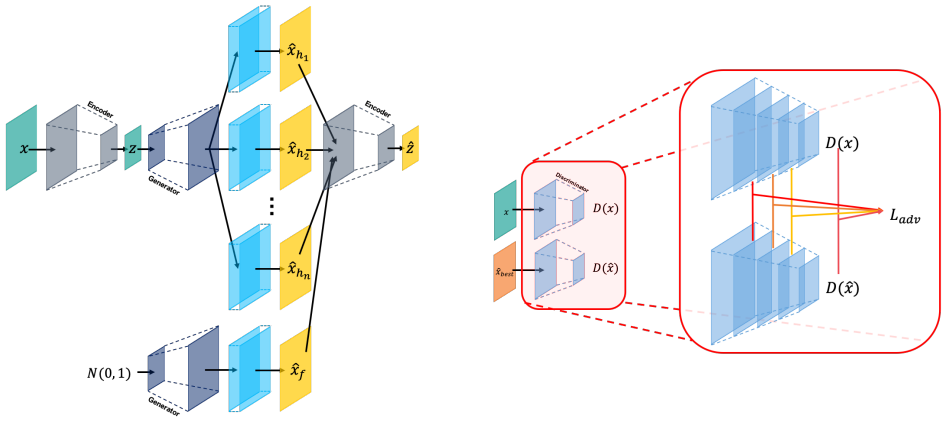

Figure 2. The architecture of the Hypothesis Pruning GAN (HP-GAN). We use a single encoder, generator, and discriminator but the last layer of the generator is constructed with the multiple hypothesis that colored with cyan. Pruning is conducted after generating the branches by multiple hypothesis.

The loss functions for training above neural network are shown in Equation 1 to Equation 8. The symbol E, G, and D are representing encoder, generator, and discriminator respectively. Also, x, z, and f_l mean input data, encoded latent vector, and feature map of l-th layer. The three loss functions $\mathcal{L}_{enc}$, $\mathcal{L}_{gen}$, and $\mathcal{L}_{adv}$ are discripted for encoder, generator, and discriminator respectively; shown in Equation 1, 2, and 7. Equation 1 and 2, are constructed with Winner-Take-All (WTA) theory [34]. The adversarial losses, from Equation 3 to 6, are aggregated in Equation 7.

$$\mathcal{L}_{enc} = ||E(x) - E(G(E(x))_{best}||_2 = ||z - \hat{z}_{best}||_2 \tag{1}$$

$$\mathcal{L}_{gen} = |x - G(E(x))_{best}|_1 = |x - \hat{x}_{best}|_1 \tag{2}$$

$$\mathcal{L}_{adv_{noise}} = ||D(x) - D(\hat{x}_{noise})||_2 \tag{3}$$

$$\mathcal{L}_{adv_{best}} = ||D(x) - D(\hat{x}_{best})||_2 \tag{4}$$

$$\mathcal{L}_{adv_{others}} = ||D(x) - D(\hat{x}_{others})||_2 / number\ of\ others \tag{5}$$

Preprints (www.preprints.org) | NOT PEER-REVIEWED | Posted: 14 February 2020 doi:10.20944/preprints202001.0028.v2

$$\mathcal{L}_{adv_{feature}} = \sum_{l=0}^{L} ||f_l(x) - f_l(\hat{x}_{others})||_2 \tag{6}$$

$$\mathcal{L}_{adv} = \mathcal{L}_{adv_{noise}} + \mathcal{L}_{adv_{best}} + \mathcal{L}_{adv_{others}} + \mathcal{L}_{adv_{feature}} \tag{7}$$

For optimizing the parameters at once, we summarize the three losses $\mathcal{L}_{enc}$, $\mathcal{L}_{gen}$, and $\mathcal{L}_{adv}$ with each weighting coefficient w_{enc}, w_{gen}, and w_{adv}. The weighting coefficients are set with values 1, 50, and 1 respectively. These coefficients are referenced from GANomaly [20] and they can be modified by the hyperparameter tuning process. We conduct the hyperparameter tuning in Section 4.4, but finding the best weighting coefficient is not covered in this paper; we only finding the best kernel size, number of the convolutional block, and learning rate.

$$\mathcal{L} = w_{enc}\mathcal{L}_{enc} + w_{gen}\mathcal{L}_{gen} + w_{adv}\mathcal{L}_{adv} \tag{8}$$

We train the HP-GAN via Algorithm 1. We use the Xavier initializer [35] for initializing the neural network, and use Adam optimizer [36] for optimizing the parameters.

Algorithm 1 Training algorithm.

Input: Measured PM values x, and random noise z_{noise}

Output: Generated PM values $\hat{X}$ with multiple hypothesis

Initialize parameters of neural network by Xavier initializer [35]
while the loss has not converged **do**
  Get $\hat{X}$ by forward propagation
  Prune $\hat{X}$ as $\hat{x}_{best}$ or others
  Compute losses $\mathcal{L}_{enc}$, $\mathcal{L}_{gen}$, and $\mathcal{L}_{adv}$
  Summarizing losses as a total loss $\mathcal{L}$
  Update parameters by backward propagation with Adam optimizer [36]
end while

3.3. Anomaly Detection Method

We describe the anomaly detection procedure in this section. The abnormality decision method is simple as shown in Algorithm 2. For using the above algorithm, input data must be measured via PM sensor firstly with containing two channels; $PM_{2.5}$ and PM_{10}. Then, the generation procedure is conducted via input the data to the neural network. We set the decision boundary θ using μ and σ of the training data as shown in Equation 9. The μ and σ represent the mean and standard deviation of the mean square error between the input and the best hypothesis of the generated samples. For reference, if the user wants to change the sensitivity or specificity of the anomaly detection, θ can be adjusted.

$$\theta = \mu + (1.5 * \sigma) \tag{9}$$

4. Experiments

In this section, we present our PM dataset and show experimental results for various neural network architectures. For assessing each model, we use Area Under the Receiver Operating characteristics Curve (AUROC) [37], Area Under the Precision-Recall Curve (AUPRC) [38], and Mean Square Error (MSE) as the performance indicators.

Preprints (www.preprints.org) | NOT PEER-REVIEWED | Posted: 14 February 2020 doi:10.20944/preprints202001.0028.v2

Algorithm 2 Abnormality decision algorithm.

Input: Measured PM values x

Output: Selected best hypothesis $\hat{x}_{best}$ among the generated samples

$\theta \leftarrow$ set threshold based on training data
if $||\hat{x} - \hat{x}_{best}||_2 \geqslant \theta$ **then**
  x is abnormal
else
  x is normal
end if

4.1. Dataset

We have collected the PM dataset from the 12 locations in the Daegu Metropolitan Jungang Library, Korea. For collecting data, we use the TEOM-based sensor as shown in Figure 3 and the collected dataset is shown in Table 2.

Figure 3. TEOM-based PM sensor.

Table 2. The collected $PM_{2.5}$ and PM_{10} values via TEOM-based sensors. The each sample $x \in \mathbb{R}^{24x2}$ represents two values ($PM_{2.5}$ and PM_{10}) for 24-hour.

	Normal	**Abnormal**	**Total**
Number of Sample	249	73	322

Each sample $x \in \mathbb{R}^{24x2}$ contains the information of $PM_{2.5}$ and PM_{10} for one day with 1-hour unit. The label of normal or abnormal is determined by meteorologists. We use 80% of the normal for training, and the other normal samples and whole abnormal samples are used for assessing the performance.

4.2. Experiment with Published Architecture

First of all, we conduct experiments for confirming which architecture among the previous studies is effective for anomaly detection. We adopt five known architectures and reconstruct them using 1D convolutional layer [20–24]. Each network such as encoder or generator uses three convolutional block, and each convolutional block consists two convolutional layer with elu activation [39]. The max-pooling is applied at the last of each convolutional block. We use two fully connected layers for the rear of the encoder and discriminator. Also, the generator includes two fully connected layers at the front of them.

We measure the performance and summarize them in Table 3. We also present the best-generated samples selected from multiple hypothesis in Figure 4 for qualitively analysis.

Preprints (www.preprints.org) | NOT PEER-REVIEWED | Posted: 14 February 2020 doi:10.20944/preprints202001.0028.v2

Table 3. Measured performance of experiment with published architecture. The purpose of this experiment is confirming which style of the architecture can detect anomaly better.

Aarchitecture	$PM_{2.5}$	PM_{10}	AUROC	AUPRC	MSE	Tr-Time	Te-Time
VAE [24]	O	O	0.92151	0.95629	0.01170	00:16:43	0.41122
	O	-	0.91699	0.95617	0.01424	**00:14:26**	0.30716
	-	O	0.92630	0.95643	0.01772	00:17:14	0.36180
GANomaly [20]	O	O	**0.93397**	**0.96485**	0.01080	00:20:10	0.34627
	O	-	0.92753	0.96209	0.00914	00:21:43	0.31621
	-	O	0.92301	0.95882	**0.01020**	00:24:21	**0.30096**
adVAE [21]	O	O	0.92849	0.95704	0.01716	00:23:42	0.40858
	O	-	0.92096	0.94832	0.02646	00:19:32	0.31841
	-	O	0.94233	0.95760	0.02264	00:19:28	0.32030
LVEAD [22]	O	O	0.88384	0.91724	0.01944	11:16:55	1.03436
	O	-	0.85370	0.87660	0.03764	10:37:55	1.03876
	-	O	0.91575	0.94738	0.01838	11:57:14	1.01579
ConAD [23]	O	O	0.92452	0.95638	0.02667	00:20:22	0.34142
	O	-	0.92342	0.95606	0.01524	00:30:05	0.31843
	-	O	0.91616	0.94154	0.01799	00:30:02	0.30392

Time consumption for training and test procedure of four architectures other than LVEAD are similar to each other but LVEAD needs much longer time. Thus, LVEAD is not efficient in the time consumption viewpoint.

The GANomaly shows the highest AUROC and AUPRC. Also, it has the lowest MSE among the above five architectures, so if who wants to use the known neural network without developing novel architecture GANomaly can be recommended.

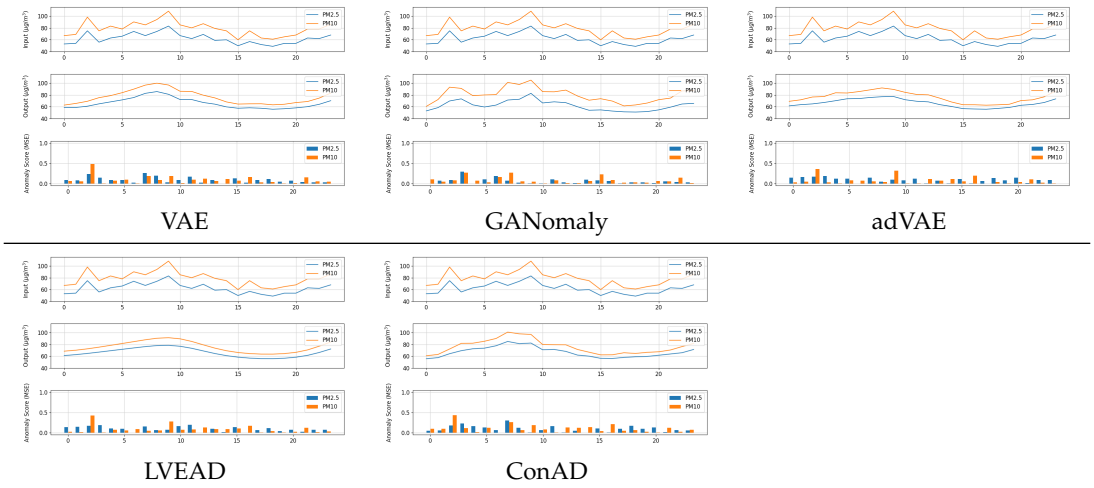

Figure 4. The generated sample from the five architectures for qualitative analysis. The x-axis of each subfigure represents an hour of the day and the label of the y-axis is noted in each subfigure.

For analyzing qualitative results, GANomaly generates the most similar sample to the input. The ConAD and VAE follow after GANomaly. The common method of VAE and ConAD is variational bound. The above method can be regarded as a reason for generating smoothed output. Thus, we conduct an ablation study to find and confirm the cause of performance degradation in the next section.

4.3. Ablation Experiment

We construct HP-GAN by referring to previous studies. HP-GAN, the novel unproven architecture, needs to confirm that it can work better than other architectures. We compose the

Preprints (www.preprints.org) | NOT PEER-REVIEWED | Posted: 14 February 2020 doi:10.20944/preprints202001.0028.v2

experiments to verify the ability of HP-GAN with our dataset. The several kinds of architectures other than HP-GAN are constructed for ablation experiments. The ablation experiment is used to find the cause of the performance impediment [40,41].

The six ablation architectures are constructed that contain the GANomaly and HP-GAN. Latent vector Matching (LM), Variational Bound (VB), and Multiple Hypothesis (MH) are used as the conditions for construct these architectures. The purpose of using LM is minimizing the Euclidean distance between latent vector z (from x) and $\hat{z}$ (from $\hat{x}$). The VB is used for minimizing the Kullback–Leibler divergence between latent vector z and normal distribution. The MH, last condition, is used to generate consistent output based on (WTA) theory.

The mini-batch size, number of the epoch, the dimension of the latent vector z, and the learning rate are used equally for all architectures. The above hyperparameters are set as 32, 1000, and 0.0001 respectively. The experimental result is summarized in Table 4.

Table 4. The performance of the six ablation (combination) architectures. We use three condition Latent vector Matching (LM), Variational Bound (VB), and Multiple Hypothesis (MH) for ablation study.

Architecture	LM	VB	MH	$PM_{2.5}$	PM_{10}	AUROC	AUPRC	MSE
Ablation-1 (LM-GAN; GANomaly)	O	-	-	O	O	0.93397	0.96485	0.01080
				O	-	0.92753	0.96209	0.00914
				-	O	0.92301	0.95882	**0.01020**
Ablation-2 (VB-GAN)	-	O	-	O	O	0.90548	0.93130	0.02148
				O	-	0.94192	0.96536	0.01924
				-	O	0.86767	0.91166	0.04300
Ablation-3 (LMVB-GAN)	O	O	-	O	O	0.89521	0.93441	0.02510
				O	-	0.86096	0.90572	0.04022
				-	O	0.89205	0.93045	0.02974
Ablation-4 (LMMH-GAN; Ours)	O	-	O	O	O	0.91699	0.95303	0.02550
				O	-	0.93219	0.96312	0.01534
				-	O	0.92616	0.95424	0.01230
Ablation-5 (VBMH-GAN)	-	O	O	O	O	**0.94644**	**0.96462**	0.03578
				O	-	0.93027	0.96153	0.01486
				-	O	0.90068	0.94807	0.01404
Ablation-6 (LMVBMH-GAN)	O	O	O	O	O	0.92603	0.94032	0.03019
				O	-	0.93123	0.96179	0.01531
				-	O	0.92712	0.95730	0.01922

The Table 4 quantitatively shows that VBMH-GAN has better AUROC, AUPRC than the other. However, as shown in Figure 5, the qualitative result, the generated sample via LM-GAN (GANomaly) and LMMH-GAN (Ours; HP-GAN) show a better result than VBMH-GAN. Moreover, other VB-based architectures commonly show the blurred result.

In this experiment, we confirm that the quantitative result represents that the VBMH-GAN is the best architecture, but the qualitative result of them is not good. Thus, we need to conduct more experiment for verifying the above architectures using varied hyperparameters, because each model may need the specific hyperparameter for performing much better.

4.4. Experiment with Various Hyperparameter

In this section, we assess four of neural networks with various hyperparameter set. For the experiment, we use the VAE as a baseline architecture because it is the first VB-based architecture. We also use GANomaly, based on LM, as one other baseline architecture. The other two architectures are our HP-GAN (LMMH-GAN) and VBMH-GAN.

We use the kernel size, number of convolutional blocks, and learning rate as the hyperparameter. Each convolutional block consists two convolutional layer with elu activation and one max pooling

Preprints (www.preprints.org) | NOT PEER-REVIEWED | Posted: 14 February 2020 doi:10.20944/preprints202001.0028.v2

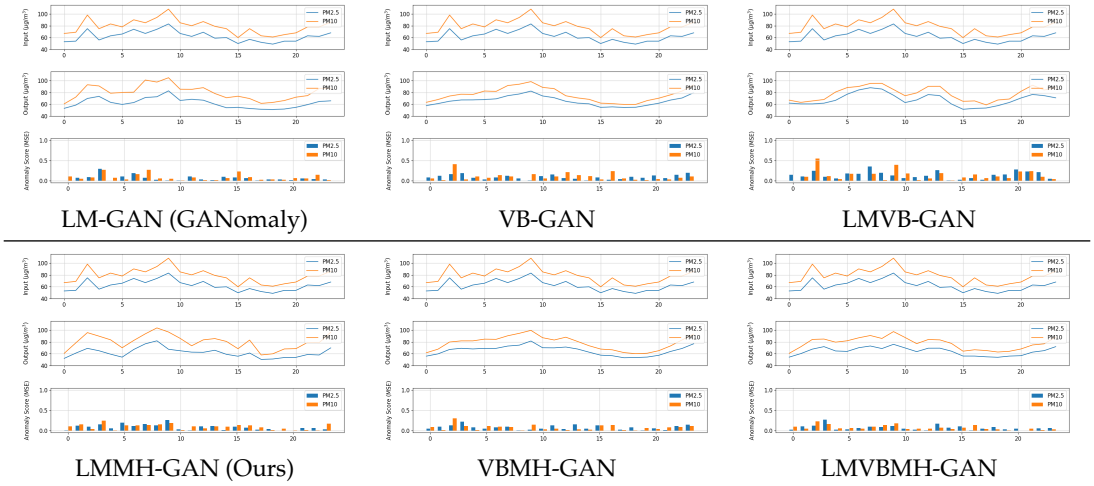

Figure 5. The generated sample from the six ablation architectures for qualitative analysis.

layer same as commented in Section 4.2. We compose the 108 kinds of the hyperparameter set for experiment via combining three hyperparameters as shown in Table 5.

Table 5. The hyperparameters for the experiment. We use the grid search method for hyperparameter tuning [**?**]. The number of convolutional blocks is abbreviated as # of conv-block.

Hyperparameter	Values
Kernel size	3, 5, 7, 9, 11, and 13
# of conv-block	2, 3, and 4
Learning rate	5e-3, 1e-3, 5e-4, 1e-4, 5e-5, and 1e-5

When the case of large kernel sizes such as 13, the time axis dimension of the feature vector can be smaller than the kernel of the last layer. However, the amount of the feature information and balance between feature vectors is not changed by the above situation, so we also use the large size kernels.

We present the measured performance for each hyperparameter with surface form in Figure 6 and 7. The height of the surface represents the anomaly detection ability; the higher height means higher performance. The flatten surface indicates that the neural network stably responds to hyperparameter changes.

When the number of the convolutional block is two or three, the surfaces for each architecture look like similar and relatively flatten then four-block cases. However, HP-GAN shows the most high and flatten surface in the four convolutional block case.

We also confirm that HP-GAN shows generally flatten surface in the AUPRC surfaces. Thus, we can conclude that HP-GAN may perform stably for anomaly detection than other architectures. However, we additionally conduct the experiment for confirming the average performance with the best hyperparameter of each model. The best hyperparameter set summarized in Table 6.

Table 6. The best hyperparameter set selected from AUROC and AUPRC surface.

Architecture	Kernel size	# of conv-block	Learning rate
VAE	7	3 (6 convolution)	5e-4
GANomaly	9	4 (8 convolution)	5e-4
HP-GAN	7	3 (6 convolution)	1e-5
VBMH-GAN	7	3 (6 convolution)	5e-5

Preprints (www.preprints.org) | NOT PEER-REVIEWED | Posted: 14 February 2020 doi:10.20944/preprints202001.0028.v2

We repeat the experiment 30 times with the randomly shuffled dataset for Monte Carlo estimation [42]. The performances are summarized with a mean ± standard deviation as shown in Table 7.

Table 7. The measured AUROC, AUPRC, and MSE are provided with a mean ± standard deviation form. The experiment is conducted with the best hyperparameter set for each architecture.

Architecture	AUROC	AUPRC	MSE
VAE	0.918 ± 0.033	0.942 ± 0.028	0.025 ± 0.009
GANomaly	0.911 ± 0.091	0.937 ± 0.062	0.027 ± 0.049
HP-GAN	**0.948 ± 0.010**	**0.967 ± 0.008**	**0.023 ± 0.016**
VBMH-GAN	0.944 ± 0.021	0.959 ± 0.020	0.033 ± 0.057

In Table 7, the higher the mean represents better performance. On the other hand, the lower standard deviation means higher stability. The HP-GAN that we proposed in this paper shows the higher performance at every indicator. Moreover, HP-GAN has higher stability (lower standard deviation) at every indicator.

5. Conclusion

We experimentally show the cutting-edge performance at anomaly detection of our HP-GAN in the TEOM-based PM sensor. The HP-GAN is trained by latent vector matching with multiple hypothesis based on WTA theory. Our neural network, HP-GAN, can generate the output more clearly and consistently with avoiding blurring effect than other VB-based models when the input data is in the normal category. The mean of AUROC and AUPRC of HP-GAN are 0.037 0.080 higher than the second performance model VBMH-GAN. Also, the mean of MSE is best (lowest) among the whole architectures. Thus, we finally conclude our HP-GAN as a cutting-edge architecture for anomaly detection.

Funding: This research received no external funding.

Acknowledgments: Thank you to all the team and corporation members. They have supported this research via not only data collection but also equipment for the experiment. Furthermore, we also thank to all the director including Dong Chan Lim for supporting this study.

Conflicts of Interest: The authors declare no conflict of interest.

Preprints (www.preprints.org) | NOT PEER-REVIEWED | Posted: 14 February 2020 doi:10.20944/preprints202001.0028.v2

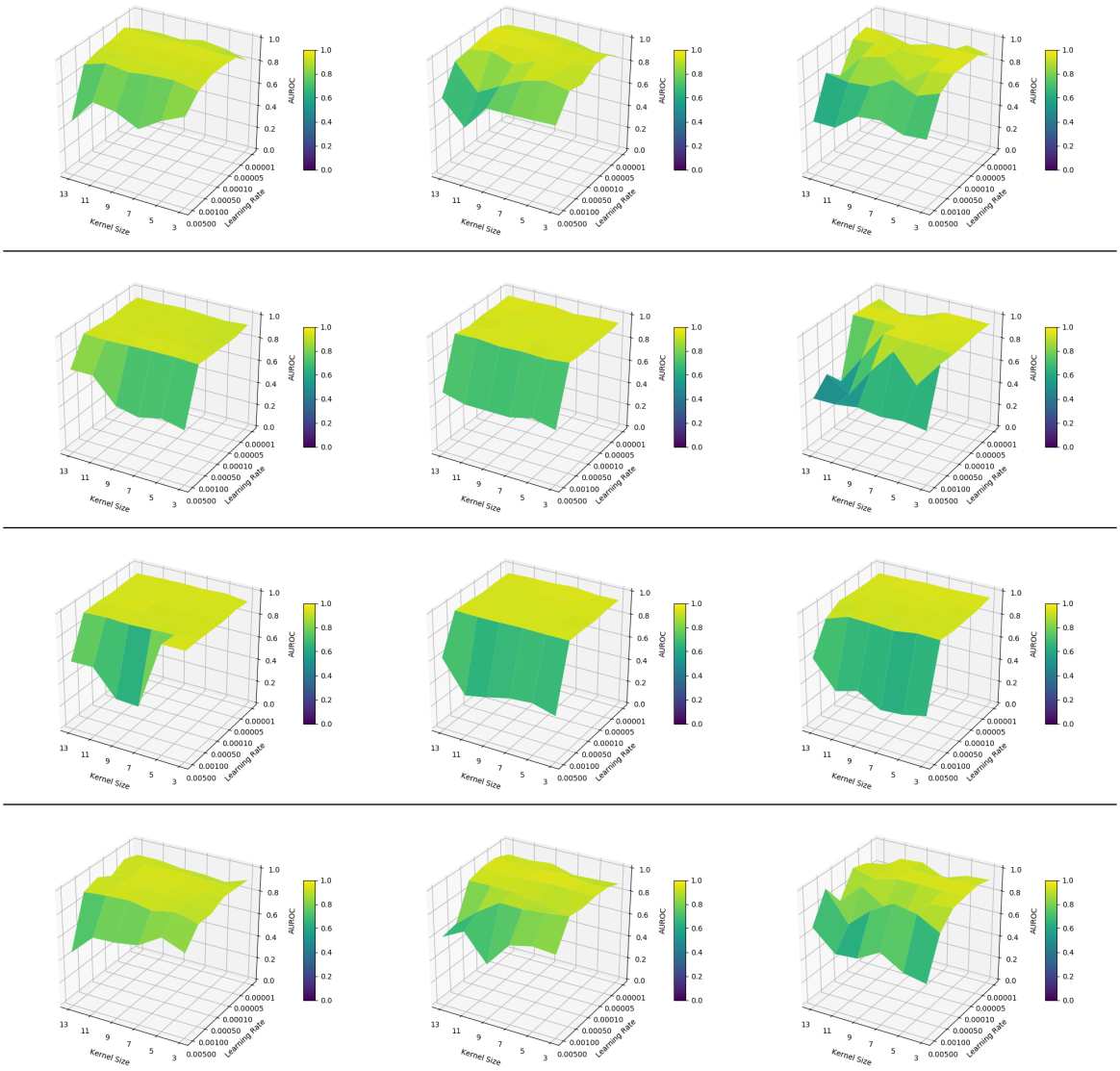

Figure 6. The surface of AUROC with various hyperparameters. The AUROC of VAE, GANomaly, and HP-GAN (LMMH-GAN), and VBMH-GAN are shown in each row sequentially. Each column shows the result of two, three, and four convolutional blocks from the left to right.

Preprints (www.preprints.org) | NOT PEER-REVIEWED | Posted: 14 February 2020 doi:10.20944/preprints202001.0028.v2

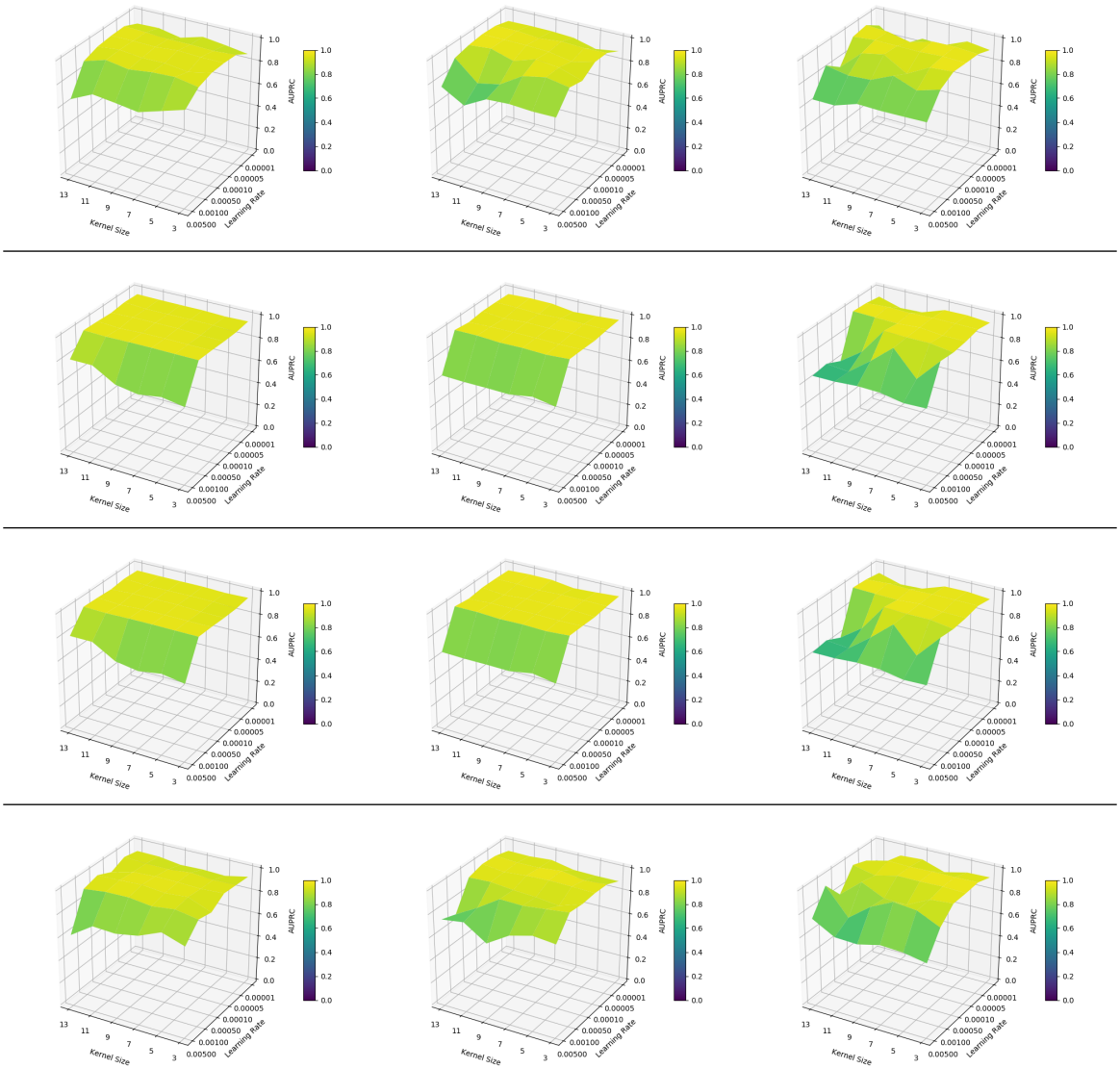

Figure 7. The surface of AUPRC with various hyperparameters. The order of the shown contents is the same as Figure 6.

 doi:10.20944/preprints202001.0028.v2

References

1. Occupational, W.H.O.; Team, E.H. WHO Air quality guidelines for particulate matter, ozone, nitrogen dioxide and sulfur dioxide : global update 2005 : summary of risk assessment, 2006.
2. Faustini, A.; Stafoggia, M.; Berti, G.; Bisanti, L.; Chiusolo, M.; Cernigliaro, A.; Mallone, S.; Primerano, R.; Scarnato, C.; Simonato, L.; Vigotti, M.; Forastiere, F. The relationship between ambient particulate matter and respiratory mortality: a multi-city study in Italy. *European Respiratory Journal* **2011**, *38*, 538–547.
3. Du, Y.; Xu, X.; Chu, M.; Guo, Y.; Wang, J. Air particulate matter and cardiovascular disease: the epidemiological, biomedical and clinical evidence. *Journal of thoracic disease* **2016**, *8*, E8—E19.
4. Hamra, G.B.; Guha, N.; Cohen, A.; Laden, F.; Raaschou-Nielsen, O.; Samet, J.M.; Vineis, P.; Forastiere, F.; Saldiva, P.; Yorifuji, T.; Loomis, D. Outdoor Particulate Matter Exposure and Lung Cancer: A Systematic Review and Meta-Analysis. *Environmental Health Perspectives* **2014**, *122*, 906–911.
5. Badura, M.; Batog, P.; Drzeniecka-Osiadacz, A.; Modzel, P. Evaluation of Low-Cost Sensors for Ambient PM2. 5 Monitoring. *Journal of Sensors* **2018**, *2018*.
6. Bulot, F.M.; Johnston, S.J.; Basford, P.J.; Easton, N.H.; Apetroaie-Cristea, M.; Foster, G.L.; Morris, A.K.; Cox, S.J.; Loxham, M. Long-term field comparison of multiple low-cost particulate matter sensors in an outdoor urban environment. *Scientific reports* **2019**, *9*, 7497.
7. Himmelblau, D.; Barker, R.; Suewatanakul, W. Fault Classification with The Aid of Artificial Neural Networks. In *Fault Detection, Supervision and Safety for Technical Processes 1991*; ISERMANN, A.; FREYERMUTH, B., Eds.; IFAC Symposia Series, Pergamon: Oxford, 1992; pp. 541 – 545.
8. Tellenbach, B.; Burkhart, M.; Schatzmann, D.; Gugelmann, D.; Sornette, D. Accurate network anomaly classification with generalized entropy metrics. *Computer Networks* **2011**, *55*, 3485 – 3502.
9. de la Hoz, E.; Ortiz, A.; Ortega, J.; de la Hoz, E. Network Anomaly Classification by Support Vector Classifiers Ensemble and Non-linear Projection Techniques. Hybrid Artificial Intelligent Systems; Pan, J.S.; Polycarpou, M.M.; Woźniak, M.; de Carvalho, A.C.P.L.F.; Quintián, H.; Corchado, E., Eds.; Springer Berlin Heidelberg: Berlin, Heidelberg, 2013; pp. 103–111.
10. Jan, S.U.; Lee, Y.; Shin, J.; Koo, I. Sensor Fault Classification Based on Support Vector Machine and Statistical Time-Domain Features. *IEEE Access* **2017**, *5*, 8682–8690.
11. Lee, K.B.; Cheon, S.; Kim, C.O. A Convolutional Neural Network for Fault Classification and Diagnosis in Semiconductor Manufacturing Processes. *IEEE Transactions on Semiconductor Manufacturing* **2017**, *30*, 135–142.
12. Steinwart, I.; Hush, D.; Scovel, C. A classification framework for anomaly detection. *Journal of Machine Learning Research* **2005**, *6*, 211–232.
13. Sakurada, M.; Yairi, T. Anomaly Detection Using Autoencoders with Nonlinear Dimensionality Reduction. Proceedings of the MLSDA 2014 2Nd Workshop on Machine Learning for Sensory Data Analysis; ACM: New York, NY, USA, 2014; MLSDA'14, pp. 4:4–4:11.
14. Zhou, C.; Paffenroth, R.C. Anomaly Detection with Robust Deep Autoencoders. Proceedings of the 23rd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining; ACM: New York, NY, USA, 2017; KDD '17, pp. 665–674.
15. Malhotra, P.; Vig, L.; Shroff, G.; Agarwal, P. Long short term memory networks for anomaly detection in time series. Proceedings. Presses universitaires de Louvain, 2015, p. 89.
16. Chauhan, S.; Vig, L. Anomaly detection in ECG time signals via deep long short-term memory networks. 2015 IEEE International Conference on Data Science and Advanced Analytics (DSAA), 2015, pp. 1–7. doi:10.1109/DSAA.2015.7344872.
17. Xu, H.; Chen, W.; Zhao, N.; Li, Z.; Bu, J.; Li, Z.; Liu, Y.; Zhao, Y.; Pei, D.; Feng, Y.; Chen, J.; Wang, Z.; Qiao, H. Unsupervised Anomaly Detection via Variational Auto-Encoder for Seasonal KPIs in Web Applications. Proceedings of the 2018 World Wide Web Conference; International World Wide Web Conferences Steering Committee: Republic and Canton of Geneva, Switzerland, 2018; WWW '18, pp. 187–196.
18. Schlegl, T.; Seeböck, P.; Waldstein, S.M.; Schmidt-Erfurth, U.; Langs, G. Unsupervised Anomaly Detection with Generative Adversarial Networks to Guide Marker Discovery. Information Processing in Medical Imaging; Niethammer, M.; Styner, M.; Aylward, S.; Zhu, H.; Oguz, I.; Yap, P.T.; Shen, D., Eds.; Springer International Publishing: Cham, 2017; pp. 146–157.

19. Donahue, J.; Krähenbühl, P.; Darrell, T. Adversarial Feature Learning. *CoRR* **2016**, *abs/1605.09782*, [1605.09782].
20. Akcay, S.; Atapour-Abarghouei, A.; Breckon, T.P. GANomaly: Semi-supervised Anomaly Detection via Adversarial Training. Computer Vision – ACCV 2018; Jawahar, C.V.; Li, H.; Mori, G.; Schindler, K., Eds.; Springer International Publishing: Cham, 2019.
21. Wang, X.; Du, Y.; Lin, S.; Cui, P.; Yang, Y. Self-adversarial Variational Autoencoder with Gaussian Anomaly Prior Distribution for Anomaly Detection. *CoRR* **2019**, *abs/1903.00904*, [1903.00904].
22. Zhang, C.; Chen, Y. Time Series Anomaly Detection with Variational Autoencoders. *CoRR* **2019**, *abs/1907.01702*, [1907.01702].
23. Nguyen, D.T.; Lou, Z.; Klar, M.; Brox, T. Anomaly Detection With Multiple-Hypotheses Predictions. Proceedings of the 36th International Conference on Machine Learning; Chaudhuri, K.; Salakhutdinov, R., Eds.; PMLR: Long Beach, California, USA, 2019; Vol. 97, *Proceedings of Machine Learning Research*, pp. 4800–4809.
24. Kingma, D.P.; Welling, M. Auto-Encoding Variational Bayes. 2nd International Conference on Learning Representations, ICLR 2014, Banff, AB, Canada, April 14-16, 2014, Conference Track Proceedings, 2014.
25. Serpico, S.B.; D'Inca, M.; Melgani, F.; Moser, G. Comparison of feature reduction techniques for classification of hyperspectral remote-sensing data. Image and Signal Processing for Remote Sensing VIII; Serpico, S.B., Ed. International Society for Optics and Photonics, SPIE, 2003, Vol. 4885, pp. 347 – 358.
26. Jun Yan.; Benyu Zhang.; Ning Liu.; Shuicheng Yan.; Qiansheng Cheng.; Fan, W.; Qiang Yang.; Xi, W.; Zheng Chen. Effective and efficient dimensionality reduction for large-scale and streaming data preprocessing. *IEEE Transactions on Knowledge and Data Engineering* **2006**, *18*, 320–333.
27. Thangavel, K.; Pethalakshmi, A. Dimensionality reduction based on rough set theory: A review. *Applied Soft Computing* **2009**, *9*, 1 – 12.
28. Balzanella, A.; Irpino, A.; Verde, R. Dimensionality Reduction Techniques for Streaming Time Series: A New Symbolic Approach. Classification as a Tool for Research; Locarek-Junge, H.; Weihs, C., Eds.; Springer Berlin Heidelberg: Berlin, Heidelberg, 2010; pp. 381–389.
29. Park, Y.; Yun, I.D. Fast Adaptive RNN Encoder–Decoder for Anomaly Detection in SMD Assembly Machine. *Sensors* **2018**, *18*.
30. Mikolov, T.; Karafiát, M.; Burget, L.; Černockỳ, J.; Khudanpur, S. Recurrent neural network based language model. Eleventh annual conference of the international speech communication association, 2010.
31. Blackman, S.S. Multiple hypothesis tracking for multiple target tracking. *IEEE Aerospace and Electronic Systems Magazine* **2004**, *19*, 5–18.
32. Streller, D.; Dietmayer, K. Object tracking and classification using a multiple hypothesis approach. IEEE Intelligent Vehicles Symposium, 2004, 2004, pp. 808–812.
33. Rupprecht, C.; Laina, I.; DiPietro, R.; Baust, M.; Tombari, F.; Navab, N.; Hager, G.D. Learning in an Uncertain World: Representing Ambiguity Through Multiple Hypotheses. The IEEE International Conference on Computer Vision (ICCV), 2017.
34. Maass, W. On the Computational Power of Winner-Take-All. *Neural Computation* **2000**, *12*, 2519–2535.
35. Glorot, X.; Bengio, Y. Understanding the difficulty of training deep feedforward neural networks. Proceedings of the thirteenth international conference on artificial intelligence and statistics, 2010, pp. 249–256.
36. Kingma, D.P.; Ba, J. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980* **2014**.
37. Fawcett, T. An introduction to ROC analysis. *Pattern Recognition Letters* **2006**, *27*, 861 – 874.
38. Saito, T.; Rehmsmeier, M. The Precision-Recall Plot Is More Informative than the ROC Plot When Evaluating Binary Classifiers on Imbalanced Datasets. *PLOS ONE* **2015**, *10*, 1–21.
39. Clevert, D.A.; Unterthiner, T.; Hochreiter, S. Fast and accurate deep network learning by exponential linear units (elus). *arXiv preprint arXiv:1511.07289* **2015**.
40. Anwar, S.; Hwang, K.; Sung, W. Structured Pruning of Deep Convolutional Neural Networks. *CoRR* **2015**, *abs/1512.08571*, [1512.08571].
41. Meyes, R.; Lu, M.; de Puiseau, C.W.; Meisen, T. Ablation Studies in Artificial Neural Networks. *CoRR* **2019**, *abs/1901.08644*, [1901.08644].
42. Kroese, D.P.; Brereton, T.; Taimre, T.; Botev, Z.I. Why the Monte Carlo method is so important today. *Wiley Interdisciplinary Reviews: Computational Statistics* **2014**, *6*, 386–392.