GAN Enhanced Anomaly State Detection in Autonomous Systems

***Abstract*—Cyber-Physical system is a field integrate communication, controlling and computing with multiple sensors and actuators connected to the physical world. Attacks from invader and complex environment could lead to abnormal state of the system. Many previous works designed various anomaly detection algorithms to handle this problem. Recently, machine learning methods has been applied in many works but anomaly samples are extreme rare which makes the performance not good. An efficiency way is data augmentation, a few works applied Generative neural networks to achieve better performance, but none of them explain how the process related to physical world and did interpret it from a knowledge-based view. Our work proposed a novel perspective on how to understand how Generative Neural Network helped exploiting the knowledge of our collected data with different anomaly detection algorithm. Additionally, we built testbeds to simulate autonomous vehicles to see how the augmented data related to physical world.**

***Index Terms*—Autonomous system, anomaly detection, data augmentation, GAN**

I. Introduction

Autonomous systems enable new functionality and services, e.g., self-driving and unmanned delivery, that promise enormous societal and economic benefits [1]. Meanwhile, easily exploitable vulnerabilities arise with the increasing functionality and network connectivity in such systems due to the transition to open and interoperable architectures [2]–[4]. Exploiting them, malicious attackers can spoof the system to perform dangerous actions and further cause catastrophic consequences [5]–[7]. For instance, the authors in [8] show spoofing attacks on GPS sensors to misguide a yacht off course. The authors in [9] present non-invasive attacks on antilock braking systems. The authors in [10] discuss remote attacks on camera and LiDAR that can disable autonomous vehicles and even cause accidents.

Cyber-physical system(CPS) is a framework integrates sensors and actuators linked to the physical world such as industrial control system and autonomous driving car [11]. For safety and secure concern, an anomaly detection system is required to maintain the system working consistently. Anomaly detection is finding the abnormal instance that does not have normal behaviour. To achieve good performance, different anomaly detection systems were designed for various research fields and complex application scenarios such as bank fraud, web intrusion and system fault monitoring. A key problem of designing good anomaly detection system is to capture anomalies as much as possible since anomaly detection system usually has higher tolerance for false positive than false negative. Because an undetected anomaly can be more harmful than a misclassified normal instance in practical. For example, comparing a survival virus process and a mistakenly killed normal process, the former is obviously more dangerous to the system.

It is important to define what is anomaly before designing the system otherwise the detection will be controversial. Generally, there are two branches of anomaly detection system: behaviour-based system and knowledge-based system [12].In a knowledge-based system, there is direct definition of the anomaly behaviour, this is very rare since most time we don't have enough knowledge of the system. In high-dimensional real-world scenario, it is very hard to define what is anomaly in the sensor data space [13].In a behaviour-based system, if some instances has different behaviour from the normal behaviour, then it is an anomaly but there is no direct definition of anomaly in the system. Machine learning has been applied in many anomaly detection algorithms on CPS in recent years [14]–[16]. From machine learning perspective, algorithms for knowledge-based system are supervised learning, algorithms for behaviour-based system are unsupervised learning. Specifically, labeling of anomaly is an expression for knowledge of the system, if we can label an instance as anomaly means knowledge requirement has been satisfied to distinguish an anomaly. We proposed a novel and general view of understanding the anomaly detection in cyber-physical system, and designed several experiments to support our theory.

Even we have the labels of the data, most time the data is extreme imbalanced [17]. Specifically, the number of anomaly are far less than the number of normal instances. Many machine learning algorithm will have bad performance on imbalanced dataset [18], so data augmentation is one of the most popular area of machine learning. But there is no work focused on linking the augmented sensor data to the physical world. How to exploit the acquired knowledge about anomaly and how to interpret the augmented data and improved performance related to physical world in cyber-physical system is our research question.

Unification of the feature space and physical world space is preferred during the exploration of the augmented data. A rational representation of data should link to the knowledge of the system or physical behaviour in real world. Cyber-physical system has actuators take the sensor data as input to perform series of behaviours. Also, the comparison of the original data and augmented data is necessary since they represented knowledge we exploit from the data.

There are 3 main contribution of this work:

- A generative neural network has been trained to get more negative data simulating the knowledge exploitation

process. The augmented data merge with the original data to a new dataset.
- Multiple anomaly detection algorithms were implemented as baselines.
- Comparison between implemented anomaly detection model performance on original data and that on dataset with generated data. We built 4 testbeds with multiple sensors and actuators for our experiment to simulate the autonomous car as a cyber-physical system.

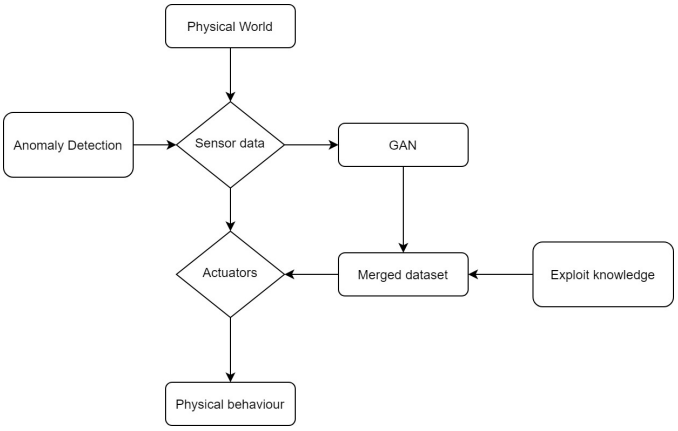

Fig. 1. Flow of design

II. Data Generation Model

A generative neural network was trained to generate some more anomaly data to see the difference.Additionally, another data augmentation method SMOTE was also implemented to compare with the data augmented by GAN.

Synthetic minority oversampling technique, also known as SMOTE is a oversampling data augmentation method dealing with imbalanced dataset. The minority class points were over-sampled from K nearest neighbors of them based on euclidean distance:

$$x_g = x + (x_n - x) * \sigma \tag{1}$$

Generated data were inserted between the minority class and a random point in K nearest neighbor which is the reason the small groups of points in Fig.9. Additionally, the shape of points has changed since new points are merged into dataset, so the projection to 2D Space has changed. The anomalies are not mixed with normal points now, most of them are on the borders of the space which means they are easier to be classified as anomalies. In other words, the generated data helped us to find a better projection from feature space to some other spaces where the features are more indicative.

Compared with the generated data with GAN, the SMOTE data are more sparse on 2D space, most GAN-generated anomaly points grouped together surrounding the original data. Furthermore, there are smaller intersection between GAN-generated anomaly and original data than that between SMOTE-generated anomaly and original data. The difference

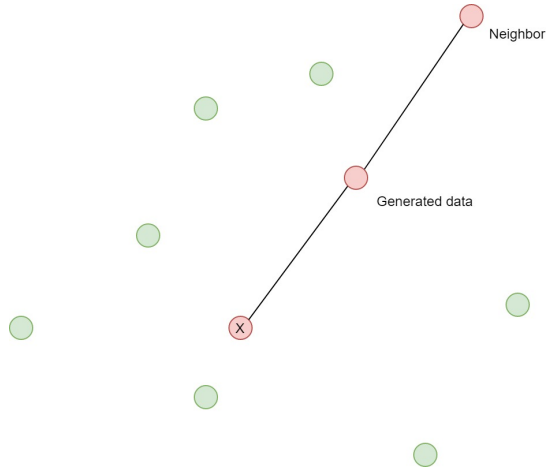

Fig. 2. SMOTE algorithm

between the visualization is due to the augmentation pattern since GAN tried to learn the distribution but SMOTE insert some new points which modified the distribution. Recent years GAN has been applied to data augmentation, especially for image data [19]–[23]. An important reason is the model performance can be tested by human eyes, generated images are acceptable for general problems if they are indistinguishable with feeding samples by human eyes. However, this is a very subjective metrics and cannot be quantified.However, use of parzen density estimation can balance the pixels' value histogram, but it cannot handle high-dimensional dataset due to its blemish.

Generative Neural Network(**GAN**) can generate data from existing data [24], sometimes the performance will be improved with augmented data. There are 2 sectors in a GAN: generator and discriminator.We use $x \in \mathbb{R}^n$ to denote the input batch data vector for single iteration during training,i.e, $x = [x_1, ..., x_n]^T$. $z \in \mathbb{R}^k$ denotes the noise vector as input for the noise variable.

- **Generator**: Generator(G) is a differentiable function which can be formulated as a multi-layer perceptron. A small random noise variable $p_z(z)$ was add on the generated data as randomness where p_z is the noise distribution. The output of the generator under distribution p_g could be expressed as $G(z, \theta_g)$ where θ_g are the parameters of the network. The goal of the generator is to make p_g to the distribution of training data x as close as possible.
- **Discriminator**: Discriminator(D) has similar structure as G except the last layer output a single value $D(x)$ represent the probability of the input comes from the distribution of training data rather than generated data.

According to above description, training a GAN is to find the

solution of a MinMax problem as shown below:

$$\min_G \max_D (D, G) = \mathbb{E}_{x \sim p_x}[logD(x)] + \mathbb{E}_{z \sim p_z}[log(1 - D(G(z)))] \quad (2)$$

The first term of this equation is trying to maximize the probability of correctly classify the training data and generated data which trained D. The second term is trying to minimize the probability of D can recognized the data is generated from G. Since there are only only 2 players(G, D) in this zero-sum problem, therefore, the solution of this problem is same as the Nash equilibrium.

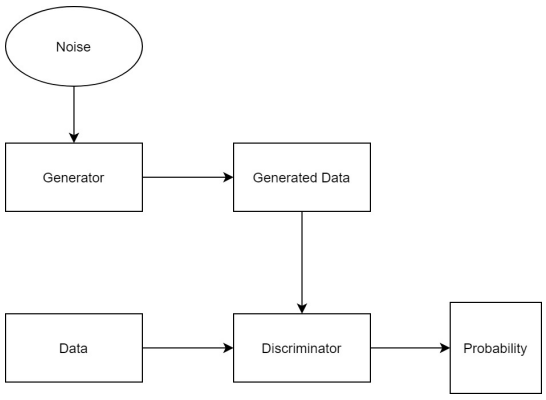

Fig. 3. GAN Structure

A. Training

2 multi-layer perceptrons as described above has been trained with gradient descent and back propagation to update the weights of neural in the network to find the local optimum. Additionally, we applied Adam optimizer to accelerate the training process.

1) Gradient Descent: An algorithm to find local optimum on a first-order differentiable function which is our lost function. Our next step will take the opposite of the current step gradient to reach the local optimum as soon as possible. a_n is the n^{th} step of the algorithm, F is the loss function of the network. As shown below, this algorithm requires the calculation of gradients, and the gradients will be updated for each iteration. However, gradient descent has several problems that makes the calculation messy and computationally expensive.

$$\theta_{n+1} = \theta_n - \gamma \nabla F(\theta_n) \quad (3)$$

γ is the learning rate or step size and θ are parameters of nodes in the network.

2) Back Propagation: Since the loss function is global instead of unique to each nodes, the weights depends on parameters of the previous layers and following layers. To address this problem, back propagation has been introduced based on chain rule and partial derivative.

$$\frac{\partial F}{\partial \theta_{ij}^k} = \frac{\partial F}{\partial a_j^k} \frac{\partial a_j^k}{\partial \theta_{ij}^k} \quad (4)$$

a_j^k is the activation for node j in k^{th} layer, in this experiment, we used ReLu as activation function between layers except the last layer of Discriminator which used Sigmoid to output a scalar. ReLu was chosen due to the sparsity design since the dimension of noise is bigger than that of data, ReLu helped the loss converge faster.

$$ReLu(x) = \begin{cases} x, & \text{if } x \geq 0 \\ 0, & \text{otherwise} \end{cases} \quad (5)$$

$$Sigmoid(x) = \frac{1}{1 + e^{-x}} \quad (6)$$

3) Adam optimizer: This is a variation of gradient descent based an adaptive estimates of lower-order moments [25]. In practical, Adam optimizer could reduce the converge time of training and can deal with noisy/sparse gradients problem for first-order gradient-based optimization of stochastic objective functions. The updated rule of Adam optimizer is similar to gradient descent with a few modification. β_i denote the exponential decay rates for the moment estimates, m_n is the biased first moment estimate, v_n is the second moment estimate. The terms with hat are the bias-corrected moment estimates, g_n denotes the gradient of the n^{th} parameter.

$$\theta_{n+1} = \theta_n - \frac{\gamma \hat{m}_n}{\sqrt{\hat{v}_n} + \epsilon} \quad (7)$$

$$g_n = \nabla F(\theta_n) \quad (8)$$

$$m_n = \beta_1 \cdot m_{n-1} + (1 - \beta_1) \cdot g_n \quad (9)$$

$$v_n = \beta_2 \cdot v_{n-1} + (1 - \beta_2) \cdot g_n^2 \quad (10)$$

$$\hat{m_n} = \frac{m_n}{1 - \beta_1^n} \quad (11)$$

$$\hat{v_n} = \frac{v_n}{1 - \beta_2^n} \quad (12)$$

B. Threat Model

Considering the autonomous car is driving on the highway with a speed limit, speedings will be considered as anomalies.Assume attackers could attack the speed sensors on the car during driving, it is necessary for system to catch it as soon as possible. A constant ϵ was adding to the speed attribute to simulate the speeding behaviour. Considering the delay of other sensors, no modification was applied on other attributes.

$$V_m = V + \epsilon \quad (13)$$

Moreover, the attackers does not have access to other sensors and have no prior knowledge of the correlation between the sensors. In other words, attackers have no capability to attack other sensors to make an uniform attack over the car. Furthermore, the training data and models are not open to the attackers. Therefore, it is impossible for attackers to generate adversarial injection samples to break the model we trained.

III. Implementation of GAN

A. Challenges

There are few challenges during the training since GAN is the one of the most difficult neural network to train. In this section, we will discuss the faced challenges during training and corresponding physical meaning.

1) Powerful Discriminator: A too powerful discriminator is a disaster of training GAN. As shown in Fig.6, the training loss of the discriminator is converged to zero, which means the discriminator is powerful enough to distinguish training data and generated data. On the other hand, this means the generator is too weak to generated good samples which leads to the failure of training. The generator does not learn anything more due to the vanishing of gradients. Corresponding to the physical word, this means the generated anomaly is not the real anomaly under the same distribution. Additionally, at knowledge level, the GAN does not exploit the knowledge from the data.On the contrast, it adding wrong knowledge to the system which results in worse performance. We tried following solutions to deal with this problem:

- Increasing the complexity of the generator to make the generated data harder to be distinguished. This can be implemented by adding different kinds of noise to the generated data.
- Adding fewer noise to the sample is also a solution to this problem since the distance between generated data and training data will be smaller.
- The failure can also due to the learning rate imbalance of the optimizers for discriminator and generator. Decreasing the learning rate of the optimizer for discriminator to an acceptable level will avoid this problem.

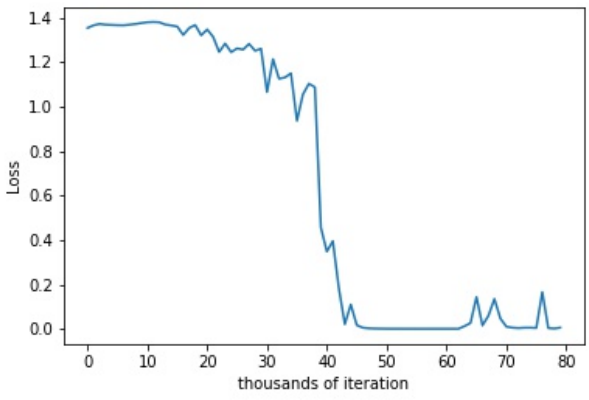

Fig. 4. The Training Loss of Powerful Discriminator

2) When to stop training?: Since our output of generator is not image data which is common, it is very hard to decide when to stop the training process. This is a critical problem which means the missing of scale for knowledge about our system. Visualization of the loss did not help deciding when to stop training except the case describe above: when the loss of discriminator dropped very fast in a few iterations. At that time, retraining is necessary to avoid nonsense results. Training the network for a reasonable long time is a compromised solution for this problem. In this experiment, the GAN was trained in hours for several times and the performance are stable. Furthermore, merging the augmented data and original data will make a new dataset, we should discuss the distribution of the new dataset. Since only negative data(anomaly) were generated, the discussion will focused on the negative data. The GAN trying to generate the data has same distribution as training data. In other word, GAN is try to learn the distribution of the data. Optimally, if a perfect GAN has been trained, the merged dataset will under the same distribution of the original data with greater size and some hidden features will be learned with more samples as so-called knowledge exploitation.

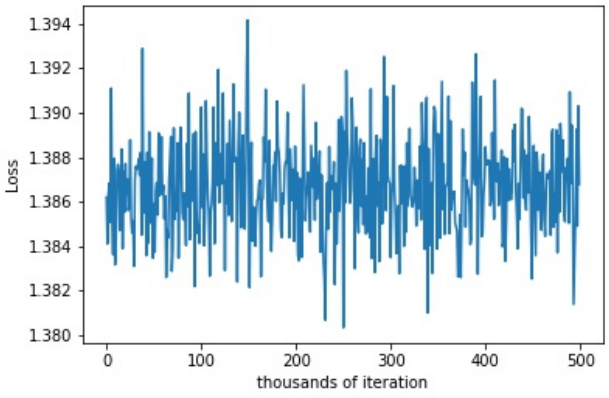

Fig. 5. The Training Loss of appropriate Discriminator

A few works explained the origin loss function of GAN has several problems [26]. Therefore,we modified the loss function a little bit to get stable results. For example, Arjpvsky has introduced a new loss function which replaced the gradient step to train the GAN [27]:

$$\mathbb{E}_{\mathbb{z}\sim\mathbb{p}(\mathbb{z})}[-\nabla_\theta log D(g_\theta(z))|\theta=\theta_0] = \nabla_\theta[KL(\mathbb{P}_{\mathbb{g}_\theta}||\mathbb{P}_\mathbb{r}) - 2JSD(\mathbb{P}_{\mathbb{g}_\theta}||\mathbb{P}_\mathbb{r})]|\theta=\theta_0 \quad (14)$$

where g_θ is a generator parameterized by parameters θ.$\mathbb{P}_{\mathbb{g}_\theta}$ is the distribution of data generated by GAN, $\mathbb{P}_\mathbb{r}$ is the distribution of original anomaly data.
Recall Kullback–Leibler divergence and Shannon Divergence Format [28], [29]:

$$KL(\mathbb{P}||\mathbb{Q}) = \Sigma P(x) log(\frac{P(x)}{Q(x)}) \quad (15)$$

$$JSD(\mathbb{P}||\mathbb{Q}) = \frac{1}{2}KL(\mathbb{P}||\mathbb{M}) + \frac{1}{2}KL(\mathbb{Q}||\mathbb{M}) \quad (16)$$

where $\mathbb{P}$ and $\mathbb{Q}$ denotes the distribution of data with density P and Q. and $\mathbb{M} = \frac{1}{2}(\mathbb{P}||\mathbb{Q})$.
As Goodfellow showed [24], we know that the second term

in equation (9) could be transformed to Jensen Shannon Divergence Format:

$$\mathbb{E}_{\mathbb{z}\sim\mathbb{p}(\mathbb{z})}[-\nabla_\theta log(1-D(g_\theta(z)))|\theta=\theta_0] = 2JSD(\mathbb{P}_{\mathbb{g}_\theta}||\mathbb{P}_{\mathbb{r}})]|\theta=\theta_0 \quad (17)$$

We can prove the gradient replacement as Arjpvsky showed with equation (2) [27]:

$$\begin{aligned} KL(\mathbb{P}_{g_\theta}||\mathbb{P}_r) &= \mathbb{E}_{x\sim p_{g_\theta}}[log\frac{P_{g_\theta}(x)}{P_r(x)}] \\ &= \mathbb{E}_{x\sim p_{g_\theta}}[log\frac{P_{g_{\theta_0}}(x)}{P_r(x)}] - \mathbb{E}_{x\sim p_{g_\theta}}[log\frac{P_{g_\theta}(x)}{P_{g_{\theta_0}}(x)}] \\ &= \mathbb{E}_{x\sim p_{g_\theta}}[log\frac{D(x)}{1-D(x)}] - KL(\mathbb{P}_{g_\theta}||\mathbb{P}_{g_{\theta_0}}) \\ &= \mathbb{E}_{x\sim p_{g_\theta}}[log\frac{D(g_\theta(z))}{1-D(g_\theta(z))}] - KL(\mathbb{P}_{g_\theta}||\mathbb{P}_{g_{\theta_0}}) \end{aligned} \quad (18)$$

Then we calculated the derivative of θ when $\theta=\theta_0$ [27]:

$$\begin{aligned} \nabla_\theta KL(\mathbb{P}_{g_\theta}||\mathbb{P}_r)|_{\theta=\theta_0} &= -\nabla_\theta \mathbb{E}_{x\sim p_{g_\theta}}[log\frac{D(g_\theta(z))}{1-D(g_\theta(z))}]|_{\theta=\theta_0} - \\ KL(\mathbb{P}_{g_\theta}||\mathbb{P}_{g_{\theta_0}})|_{\theta=\theta_0} & \\ &= \mathbb{E}_{x\sim p_{g_\theta}}[-\nabla_\theta log\frac{D(g_\theta(z))}{1-D(g_\theta(z))}]|_{\theta=\theta_0} \end{aligned} \quad (19)$$

Equation (2) subtracted the last line of equation (19) simply get equation (17) as stated.Considering the complex mathematical proof,an explanation in a knowledge view would be more straightforward. KL divergence makes the discriminator very strict since KL divergence is not symmetric. This is pretty intuitive:

$$KL(\mathbb{P}||\mathbb{Q}) = \Sigma P(x)log(\frac{P(x)}{Q(x)}) \quad (20)$$

$$= -\Sigma P(x)log(\frac{Q(x)}{P(x)}) \quad (21)$$

On the other hand, discarding a not too bad generated sample is almost no cost, sometimes the GAN will failed due to this reason. Therefore, JSD was introduced to generalize the imbalance since it is symmetric. That's the reason why GAN is learning the distribution of the data since its objective function is a mixture of KL divergence and JSD who are designed to measure the distance between distribution. There is another interesting observation: If we assume the original data is under Gaussian distribution, and the GAN is trying to learn from the distribution so that the generated data is also under Gaussian distribution. Then JSD here is just comparing the distance between 2 Gaussian distribution and so does the middle term M since the sum of 2 Gaussian distribution is also an Gaussian Distribution.

IV. Testbed and Dataset

A good testbed should satisfied several experiment requirements to support cyber-physical system research. A few main capabilities should be developed to implement a series of application as specified in this work [30], here we made some modification :

- *Cyber-Physical Simulation*: The testbed is supposed to perform as a cyber-physical system integrated security,communication and control. In other words, it should be able to simulate physical world behaviour of a system for further analysis.
- *Data Integration*: This capability guaranteed the accessibility of data from multiple different sources on testbed in each step during simulation. The testbed should be able to fuse data with unavoidable noise in practical and help researchers understand the status of the system.
- *Scalability*: When the system is composed of a great number of agents, scalability of testbeds should be considered. For example, internet of vehicles usually contains multiple cars for analysis and experiment.

4-wheel testbeds have been built to simulate autonomous car. At this stage, there are 4 testbeds running without trails for data collection as shown in Figure.2. A group of accessories

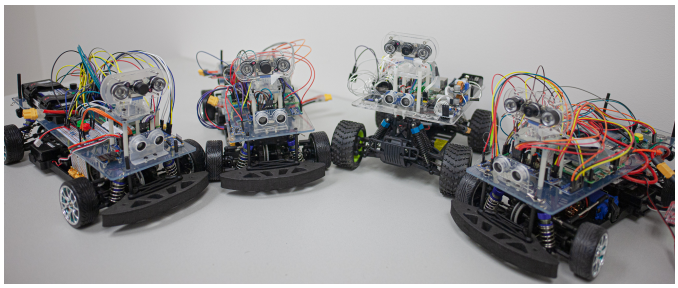

Fig. 6. Testbeds

have been launched on Raspberry Pi board including camera, sonar and lots of sensors. The sensor data mainly comes from a 9 degrees of freedom breakout board as shown in table 1. There are a few adjustment we made on original data:

- **AS5048A** is a magnetic position sensor to collect the rotatory position of the motor shaft. The range of value for this sensor is between 0 to 0x3fff on Hexadecimal, we first transform the value into decimal and find the difference between time states to calculate the speed of the testbed.
- **BNO055** is a 9-DOF sensor which can do the sensor fusion on its own, calibration for system, gyroscope, accelerometer and magnetometer were required before each time the charge on, this is due to the design of the board. This step are responsible for fusing noisy raw sensor data into an accurate orientation reading.
- **Normalization** has been applied to data to avoid the effect of different scaling on anomaly detection algorithm performance, the data will be biased otherwise.

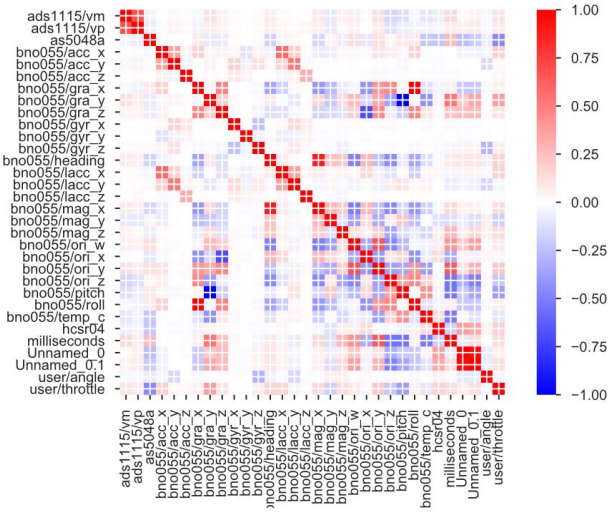

Fig. 7. Correlation Matrix

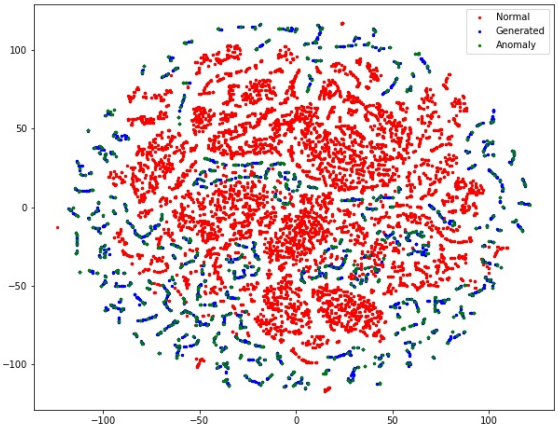

Fig. 9. t-sne visualization of merged data with SMOTE

The dataset has 13013 samples and 27 attributes, only 1000 of them are negative samples which is 7.68% of the dataset. Due to the shortage of GAN, small data whose number of samples are less than number of attributes was not feasible since it is not enough for GAN to learn the distribution. Neither do big data because there is no need for GAN if we already have enough data. As shown in Fig.2, the correlation matrix is almost diagonal, therefore, the naive bayes assumption was reasonable and feasible for our data. In other words, the attributes in our data were almost independent which means the data is free of multicolinearity for multi invariant linear regression models. Contextual data was not collected until now, all the data are collected when testbeds running on ground without slopes and pothole. In further research, complex contexts will be considered to simulate the physical world challenges.

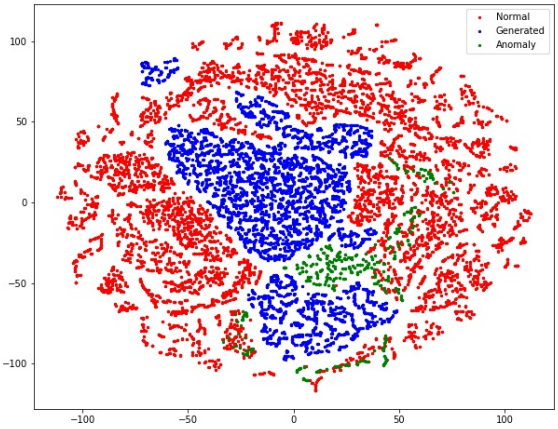

Fig. 10. t-sne visualization of merged data with GAN

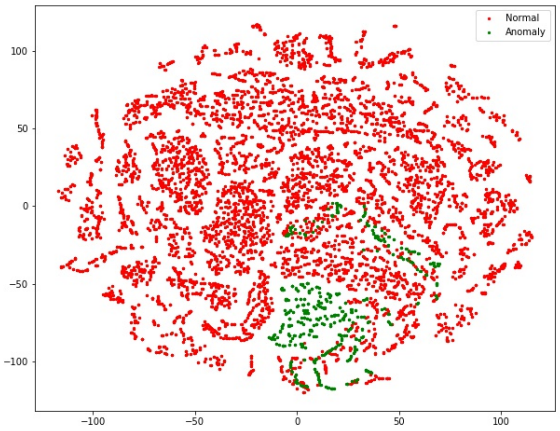

Fig. 8. t-sne visualization of original data

The necessity of augmented data should be discussed at very first. Is there a possibility that some machine learning algorithms can efficiently detect anomaly? The answer should come from the data itself instead of discussion of the strength of algorithms. Visualization of high-dimension data is required to get a sense of their relationship in an understandable space.The visualization of original dataset and merged dataset with generated data were implemeted by t-sne.

T-sne was introduce by Maaten in 2008 to help scientists understand data into 2D or 3D space for visualization [31], the only parameter we changed from default setting is the perplexity due to the data size we have. As shown in Fig.8, the red point are the normal samples, the green points are the anomaly samples. There are non-continuous clusters of them which provides a signal that we may not have enough knowledge to classify them properly with current data. Furthermore,

TABLE I
DATA COLLECTED FROM THE TESTBEDS

Type	Component	Model	Symbol	Data Type	Meaning
Sensor Measurements	Ultrasonic Distance Sensors	HC-SR04	hcsr04	float64	distance from barriers in front of the car in cm
	Analog-to-digital Converter (ADC)	ADS1115	vm, vp	float64	voltage of the battery of the motor and controller in V
	Magnetic Position Sensor	AS5048A	as5048a	int64	rotatory position of the motor shaft (0-0x3fff)
	9-DOF IMU Sensor	BNO055	heading, roll, pitch	float64	three axis orientation data in Euler Angles format
			ori_x, ori_y, ori_z, ori_w	float64	four orientation data in quaternion format
			lacc_x, lacc_y, lacc_z	float64	three axis of linear acceleration data (acceleration minus gravity) in m/s^2
			gra_x, gra_y, gra_z	float64	three axis of gravitational acceleration (minus any movement) in m/s^2
			acc_x, acc_y, acc_z	float64	three axis of acceleration (gravity + linear motion) in m/s^2
			gyr_x, gyr_y, gyr_z	float64	three axis of 'rotation angular velocity' in rad/s
			mag_x, mag_y, mag_z	float64	three axis of magnetic field sensing in micro Tesla (uT)
			temp_c	int64	Ambient temperature in degrees Celsius
	Camera	OV5647	image	JPEG image	images of track
State Estimations	Controller	Raspberry Pi 4	speed	float64	revolving speed of the motor shaft computed from as4048a in r/s
			bias1, bias2, bias3	float64	deviation from target path computed from the top, middle and bottom of the image
Actuations	Electronic Speed Controls (ESC)	Brushed	motor	int64	PWM duty cycle to control motor
	Steering Servo	Standard size high-torque	servo	int64	PWM duty cycle to control servo

merging the augmented data and original data will make a new dataset, we should discuss the distribution of the new dataset. Since only negative data(anomaly) were generated, the discussion will focused on the negative data. The GAN trying to generate the data has same distribution as training data. In other word, GAN is try to learn the distribution of the data. Optimally, if a perfect GAN has been trained, the merged dataset will under the same distribution of the original data with greater size and some hidden features will be learned with more samples as so-called knowledge exploitation.

V. ANOMALY DETECTION ALGORITHMS

Supervised machine learning algorithms showed an excellent performance on public dataset, especially on low-dimensional dataset. This problem can also be transformed to a classification problem. Additionally, without consideration of anomaly type, anomaly detection is a binomial classification problem. Generally, the algorithms can be devided into 3 categrories: distance-based methods, distribution-based methods, deep learning. Since deep learning methods usually lack of enough theoretical support of the great performance, they are not selected as baseline.

A. Central limit Theorem

The Central limit theorem(CLT) is an intuitive common sense about relationship between sample and population. If the sample size is big enough(practically greater than 30), the normalized sum of random variable is under normal distribution even the origin variable is not under normal distribution. CLT provided the theoretical support for our normalization of data without assuming data is under normal distribution. It can be expressed as follow:

$$F(x) = \lim_{x\to\infty} \frac{\Sigma_{i=1}^{n} X_i - n\mu}{\sqrt{n}\sigma} = \frac{1}{\sqrt{2\pi}} \int_{-\infty}^{x} e^{\frac{t^2}{2}}\, dt \qquad (22)$$

where μ and σ are the average and standard deviation of the distribution.

B. Anomaly Detection Algorithms

In this section, all algorithms applied to this work will be introduced and explained briefly. Algorithms has been divided into supervised and unsupervised according to the requirement of labels.

1) Unsupervised methods:

- **Isolation Forest**(IF) was introduced by Liu [32] in 2008, it has been well applied in industry for anomaly detection.

Anomaly are set of points deviated from normal points on feature spaces, therefor, it is easier to separate them out than normal points from other points.This algorithm assumed anomalies are few and heterogeneous compared to normal instances.

- **Local Outlier Factor**(LOF) took care of k nearest neighbors to estimate the local density [33]. The points have low local density will be considered as anomaly(Outlier). And density was measured by LOF score.
- **CBLOF** is a variation of LOF based on cluster density and distance [34]. The data points has been divided into clusters, and points in small clusters which closed to huge cluster are consider as local outlier. The original paper applied squeezer algorithm [35] but any clustering algorithm is feasible as long as the clustering result is good.

- **Auto Encoder**(AE) is an unsupervised artificial neural network to extract features of data, it can also detect outliers in feature space. Auto Encoder was trained to minimize the reconstruction loss to get most representative features.Training Auto Encoder is similar to training other neural networks, the primary goal is searching for appropriate weights to minimize the reconstruction loss using back-propagation. Only normal samples are using for training, if we find a sample has big reconstruct error then it considered as an anomaly.
- **Angle-based Outlier Detection**(ABOD) has an important position when dealing with high-dimensional data for anomaly detection [36].
- **One Class Support Vector Machine**(OCSVM) is a special variation of support vector machine to find anomaly in data. Specifically, this methods assumed we only know how normal data looks like, and if a new data is far from the training data it will be considered as anomaly which is similar to auto encoder. And the problem could be transform to a optimization problem for best decision boundary in a dot product space.

2) Supervised methods:

- **K-nearest neighbors**(KNN) assigned label to a point based on k nearest neighbors' labels closed to it. In this work, we trained a knn classifier to classify data in testset with $k = 20$. knn is a computing-expensive non-parametric algorithm, especially for big dataset in high dimension, the test time complexity is $O(mn)$ where m is the number of features and n is the size of training set. Furthermore, KNN has good performance on imbalanced data since the classification process only involved a few points closed to it.
- **Gaussian Naive Bayes classifier**(GNBC) is a variation of naive Bayesian classifier. Bayes' Theorem is a rule described the conditional probability of 2 random events based on prior knowledge:

$$P(A|B) = \frac{P(B|A)P(A)}{P(B)} \tag{23}$$

TABLE II
RECALL PERFORMANCE FOR ORIGINAL DATA AND GENERATED DATA

	Original Recall	SMOTE Recall	GAN Recall
IF	0.33	0.33	0.32
ABOD	0.53	0.5	0.5
CBLOF	0.5	0.51	0.52
LOF	0.51	0.52	0.51
AE	0.5	0.5	0.5
KNN	0.82	0.94	0.97
GNBC	0.95	0.99	0.98
BNBC	0.92	0.92	0.93
OCSVM	0.52	0.51	0.51

Naive Bayes classier assumed all the features of data are independent, it has stable and good performance on small dataset with small dependency between attributes.Gaussian Naive Bayes Classifier is a special case of Gaussian Bayes classifier which model $P(x|t)$ as a Gaussian distribution encoding the dependent relation into covariance matrix. The covariance matrix Σ of data with naive bayes assumption is an identity matrix whose determinant is 1.

- **Bernoulli Naive Bayes Classifier**(BNBC) is a binary case of Naive Bayes classifier with the assumption of Bernoulli distribution of data. In our dataset, there are 2 classed of data: normal and anomaly, but the feature is not binary, therefore, the performance of BNBC should be worse than GNBC.

As mentioned in the very beginning of this paper, there are different evaluation metrics for different scenarios. Anomaly detection system are expected to detect the malicious behaviour even it is not a real anomaly. Therefore, recall should be one of the best candidates for evaluation.

$$Recall = \frac{TP}{TP + FN} \tag{24}$$

We made a comparison with above algorithms applied on orginal dataset and dataset with generated data. As shown in Table.2, there are few observations interesting. First, the unsupervised methods had worse performance than supervised methods in general since labels was not used. Second, unsupervised methods did not get benefit from augmented training data by the same reason above, some models even get worse performance. Third, all the supervised methods had a better performance on different levels. Specifically, the KNN had big improvement from 0.82 to over 0.94.

VI. RELATED WORKS

Autonomous vehicles are vulnerable to attacks during the communication between sensors and physical environment which is clearly demonstrated by so-called non-invasive sensor attacks. The authors in [9] showed arbitrary measurements has been injected to the analog sensors which can cause dangerous situations. The authors in [37] introduced how to inject fake sensor measurement to fool the failure detector. Intrusion detection/Anomaly detection has been well studied for different cyber-physical system [14]–[16], [38]. Most of

them applied unsupervised methods due to the lack of anomaly data. Sonntag tried to explained the knowledge acquisition process with the system development [38], he only focused on the so-called human-in-the-loop knowledge acquisition as the basis of the data integration. Unfortunately, the exploitation process is not discussed during anomaly detection. Also, the details and steps of knowledge acquisition were not included in that work. Goh has applied deep learning to anomaly detection in cyber-physical system [16], but it does not discussed the anomaly on a distribution level.

Data augmentation is an intuitive solution for imbalance dataset. Generally, there are 2 categories of this technique: oversampling and downsampling. Downsampling reduced the majority class data in the dataset to balance the data ration.However, it is not popular since lots of priceless information lost during the process. Oversampling is sampling more from the minority class points, SMOTE is one of the most classic methods of it [39]. There are also some hybrids methods combining SMOTE and downsampling methods to speed up the balancing process. GAN is a new oversampling method applied deep learning which produces incredible results [24]. There are different variations of GAN to cover the disadvantages of the original design such as CGAN and InfoGAN [40] but they also have their own problems such as difficulty of training and poor interpretability.

There is also a concern about the theoretical explanation of the augmented data. How to interpret the augmented data and how to connect them to the problem itself? Several scientists found that GANs performed as a generator for the model to avoid overfitting. Liang et al. [41] analyzed how GANs learn distribution densities with non-parametric and parametric results based on rate of convergence, and they introduce a new pair regularization. Authors in [42] formulate data augmentation into a kernel expression. Authors in [43] summarized the geometric methods and photometric methods of data augmentation with images. But it only focused on transformation of existed data so it was unavailable for non-image datasets.

VII. Conclusion

In this work, the general problem that cyber-physical faced with anomaly detection was discussed. Data augmentation are sometimes necessary when the dataset is imbalanced. The knowledge exploitation process has also been connected with cyber-physical system to have an better understanding of these techniques. Moreover, our theory was tested on real data collected from the testbeds we built. The improved performance is not due to the new knowledge adding into the data we have. Instead, they come from the unexploited knowledge hidden in the original data. Unfortunately, time-series analysis was not performed which will make the results more convincing. If time-series GAN was implemented, the generated data could be fed to actuators in cyber-physical system to see the physical behaviours. It also provides a new way to evaluate the generated data quality and knowledge exploitation process. In future, the experiments will be replicated on greater sized time-series data. Additionally,this dataset only contains one type of anomaly which could be extend to multi-class problem. In other words, different anomalies would make the results more general and convincing.

References

[1] D. J. Fagnant and K. Kockelman, "Preparing a nation for autonomous vehicles: opportunities, barriers and policy recommendations," *Transportation Research Part A: Policy and Practice*, vol. 77, pp. 167–181, 2015.

[2] S. Chaterji, P. Naghizadeh, M. A. Alam, S. Bagchi, M. Chiang, D. Corman, B. Henz, S. Jana, N. Li, S. Mou *et al.*, "Resilient cyberphysical systems and their application drivers: A technology roadmap," *arXiv preprint arXiv:2001.00090*, 2019.

[3] S. Parkinson, P. Ward, K. Wilson, and J. Miller, "Cyber threats facing autonomous and connected vehicles: Future challenges," *IEEE transactions on intelligent transportation systems*, vol. 18, no. 11, pp. 2898–2915, 2017.

[4] M. Amoozadeh, A. Raghuramu, C.-N. Chuah, D. Ghosal, H. M. Zhang, J. Rowe, and K. Levitt, "Security vulnerabilities of connected vehicle streams and their impact on cooperative driving," *IEEE Communications Magazine*, vol. 53, no. 6, pp. 126–132, 2015.

[5] T. He, L. Zhang, F. Kong, and A. Salekin, "Exploring inherent sensor redundancy for automotive anomaly detection," *57th Design Automation Conference*, 2020.

[6] F. Kong, M. Xu, J. Weimer, O. Sokolsky, and I. Lee, "Cyber-physical system checkpointing and recovery," in *2018 ACM/IEEE 9th International Conference on Cyber-Physical Systems (ICCPS)*. IEEE, 2018, pp. 22–31.

[7] F. Kong, O. Sokolsky, J. Weimer, and I. Lee, "State consistencies for cyber-physical system recovery," 2019.

[8] A. H. Rutkin, "spoofers use fake gps signals to knock a yacht off course," *MIT Technology Review*, 2013.

[9] Y. Shoukry, P. Martin, P. Tabuada, and M. Srivastava, "Non-invasive spoofing attacks for anti-lock braking systems," in *International Workshop on Cryptographic Hardware and Embedded Systems*. Springer, 2013.

[10] J. Petit, B. Stottelaar, M. Feiri, and F. Kargl, "Remote attacks on automated vehicles sensors: Experiments on camera and lidar," *Black Hat Europe*, 2015.

[11] E. A. Lee, "Cyber physical systems: Design challenges," in *2008 11th IEEE International Symposium on Object and Component-Oriented Real-Time Distributed Computing (ISORC)*. IEEE, 2008, pp. 363–369.

[12] S. Etalle, "Network monitoring of industrial control systems: The lessons of securitymatters," in *Proceedings of the ACM Workshop on Cyber-Physical Systems Security & Privacy*, 2019, pp. 1–1.

[13] S. M. Erfani, S. Rajasegarar, S. Karunasekera, and C. Leckie, "High-dimensional and large-scale anomaly detection using a linear one-class svm with deep learning," *Pattern Recognition*, vol. 58, pp. 121–134, 2016.

[14] C. Liu, S. Ghosal, Z. Jiang, and S. Sarkar, "An unsupervised spatiotemporal graphical modeling approach to anomaly detection in distributed cps," in *2016 ACM/IEEE 7th International Conference on Cyber-Physical Systems (ICCPS)*. IEEE, 2016, pp. 1–10.

[15] J. Yang, C. Zhou, S. Yang, H. Xu, and B. Hu, "Anomaly detection based on zone partition for security protection of industrial cyber-physical systems," *IEEE Transactions on Industrial Electronics*, vol. 65, no. 5, pp. 4257–4267, 2017.

[16] J. Goh, S. Adepu, M. Tan, and Z. S. Lee, "Anomaly detection in cyber physical systems using recurrent neural networks," in *2017 IEEE 18th International Symposium on High Assurance Systems Engineering (HASE)*. IEEE, 2017, pp. 140–145.

[17] H. He and E. A. Garcia, "Learning from imbalanced data ieee transactions on knowledge and data engineering v. 21 n. 9," 2009.

[18] F. Provost, "Machine learning from imbalanced data sets 101," in *Proceedings of the AAAI'2000 workshop on imbalanced data sets*, vol. 68. AAAI Press, 2000, pp. 1–3.

[19] M. Frid-Adar, I. Diamant, E. Klang, M. Amitai, J. Goldberger, and H. Greenspan, "Gan-based synthetic medical image augmentation for increased cnn performance in liver lesion classification," *Neurocomputing*, vol. 321, pp. 321–331, 2018.

[20] G. Mariani, F. Scheidegger, R. Istrate, C. Bekas, and C. Malossi, "Bagan: Data augmentation with balancing gan," *arXiv preprint arXiv:1803.09655*, 2018.

[21] C. Bowles, L. Chen, R. Guerrero, P. Bentley, R. Gunn, A. Hammers, D. A. Dickie, M. V. Hernández, J. Wardlaw, and D. Rueckert, "Gan augmentation: Augmenting training data using generative adversarial networks," *arXiv preprint arXiv:1810.10863*, 2018.

[22] S. K. Lim, Y. Loo, N.-T. Tran, N.-M. Cheung, G. Roig, and Y. Elovici, "Doping: Generative data augmentation for unsupervised anomaly detection with gan," in *2018 IEEE International Conference on Data Mining (ICDM)*. IEEE, 2018, pp. 1122–1127.

[23] H.-C. Shin, N. A. Tenenholtz, J. K. Rogers, C. G. Schwarz, M. L. Senjem, J. L. Gunter, K. P. Andriole, and M. Michalski, "Medical image synthesis for data augmentation and anonymization using generative adversarial networks," in *International workshop on simulation and synthesis in medical imaging*. Springer, 2018, pp. 1–11.

[24] I. Goodfellow, J. Pouget-Abadie, M. Mirza, B. Xu, D. Warde-Farley, S. Ozair, A. Courville, and Y. Bengio, "Generative adversarial nets," in *Advances in neural information processing systems*, 2014, pp. 2672–2680.

[25] D. P. Kingma and J. Ba, "Adam: A method for stochastic optimization," *arXiv preprint arXiv:1412.6980*, 2014.

[26] I. Goodfellow, Y. Bengio, and A. Courville, *Deep learning*. MIT press, 2016.

[27] M. Arjovsky and L. Bottou, "Towards principled methods for training generative adversarial networks. arxiv e-prints, art," *arXiv preprint arXiv:1701.04862*, 2017.

[28] S. Kullback and R. A. Leibler, "On information and sufficiency," *The annals of mathematical statistics*, vol. 22, no. 1, pp. 79–86, 1951.

[29] B. Fuglede and F. Topsoe, "Jensen-shannon divergence and hilbert space embedding," in *International Symposium onInformation Theory, 2004. ISIT 2004. Proceedings.* IEEE, 2004, p. 31.

[30] B. Chen, K. L. Butler-Purry, A. Goulart, and D. Kundur, "Implementing a real-time cyber-physical system test bed in rtds and opnet," in *2014 North American Power Symposium (NAPS)*. IEEE, 2014, pp. 1–6.

[31] L. v. d. Maaten and G. Hinton, "Visualizing data using t-sne," *Journal of machine learning research*, vol. 9, no. Nov, pp. 2579–2605, 2008.

[32] F. T. Liu, K. M. Ting, and Z.-H. Zhou, "Isolation forest," in *2008 Eighth IEEE International Conference on Data Mining*. IEEE, 2008, pp. 413–422.

[33] M. M. Breunig, H.-P. Kriegel, R. T. Ng, and J. Sander, "Lof: identifying density-based local outliers," in *Proceedings of the 2000 ACM SIGMOD international conference on Management of data*, 2000, pp. 93–104.

[34] Z. He, X. Xu, and S. Deng, "Discovering cluster-based local outliers," *Pattern Recognition Letters*, vol. 24, no. 9-10, pp. 1641–1650, 2003.

[35] ——, "Squeezer: an efficient algorithm for clustering categorical data," *Journal of Computer Science and Technology*, vol. 17, no. 5, pp. 611–624, 2002.

[36] H.-P. Kriegel, M. Schubert, and A. Zimek, "Angle-based outlier detection in high-dimensional data," in *Proceedings of the 14th ACM SIGKDD international conference on Knowledge discovery and data mining*, 2008, pp. 444–452.

[37] Y. Mo and B. Sinopoli, "Integrity attacks on cyber-physical systems," in *Proceedings of the 1st international conference on High Confidence Networked Systems*, 2012, pp. 47–54.

[38] D. Sonntag, S. Zillner, P. van der Smagt, and A. Lörincz, "Overview of the cps for smart factories project: Deep learning, knowledge acquisition, anomaly detection and intelligent user interfaces," in *Industrial internet of things*. Springer, 2017, pp. 487–504.

[39] N. V. Chawla, K. W. Bowyer, L. O. Hall, and W. P. Kegelmeyer, "Smote: synthetic minority over-sampling technique," *Journal of artificial intelligence research*, vol. 16, pp. 321–357, 2002.

[40] X. Chen, Y. Duan, R. Houthooft, J. Schulman, I. Sutskever, and P. Abbeel, "Infogan: Interpretable representation learning by information maximizing generative adversarial nets," in *Advances in neural information processing systems*, 2016, pp. 2172–2180.

[41] T. Liang, "On how well generative adversarial networks learn densities: Nonparametric and parametric results," *arXiv preprint arXiv:1811.03179*, 2018.

[42] T. Dao, A. Gu, A. J. Ratner, V. Smith, C. De Sa, and C. Ré, "A kernel theory of modern data augmentation," *Proceedings of machine learning research*, vol. 97, p. 1528, 2019.

[43] L. Taylor and G. Nitschke, "Improving deep learning using generic data augmentation," *arXiv preprint arXiv:1708.06020*, 2017.