

Deep Learning with AnoGAN and Efficient GAN to Judge Agricultural Harvest Image Data

Shinji KAWAKURA

Research Center for Artificial Photosynthesis
Osaka City University
Osaka, Japan
E-mail address: s.kawakura@gmail.com

Ryosuke SHIBASAKI

Center for Spatial Information Science
The University of Tokyo
Tokyo, Japan
E-mail address: shiba@csis.u-tokyo.ac.jp

Abstract—Recent social and academic endeavors have required judging harvest images to determine whether they show edible produce. Thus, we construct, apply, analyze, and present deep-learning-based Anomaly Detection with Generative Adversarial Nets (AnoGAN) and Efficient Generative Adversarial Nets (Efficient GAN) including the system for implementation. Specifically, we first capture and accumulate two image datasets, one where the images were judged by experienced agri-workers as showing edible produce, and the other where the images were judged as showing inedible content. Then, we construct a system to generate corrected picture datasets using Keras based on the GAN (in this study, particularly AnoGAN and Efficient GAN). Furthermore, we apply a discriminator to judge the images in the dataset. Considering past studies, the discriminator has adequate accuracy for certain practical uses, however not for our purpose, so we apply AnoGAN and Efficient GAN to improve the results. Here, we present numerical data on effective detection and removal of inedible harvests. This system could provide benefits for agri-workers and developers, and, in future, realize more precise and stable agri-analyzing.

Contribution— The methodologies could be applied to real agri-sites to promote agricultural productivity.

Keywords—Agricultural harvests, AnoGAN, Efficient GAN, deep learning, anomaly score.

I. INTRODUCTION

Recent social and academic requirements have seen an increasing focus on image recognition methodologies for agricultural applications. The targets are agricultural (agri-) workers, managers, technicians, researchers, and non-living targets (e.g., workers, harvests, agri-tools). Recent machine-learning-based studies, particularly regarding deep learning, have suggested diverse measures including from other technical fields' methodologies using camera systems to judge targets (e.g., various robot arm-based machines for harvesting vegetables, machines to pull up weeds) [1]–[14].

Although some of these approaches have included Artificial Intelligence (AI)-based monitoring and identification tools for edible and inedible targets, their accuracy has not been sufficient for practical use. Thus,

further improvements have incorporated computing processes based on human feelings and commonsense-based thinking. These solutions consider up-to-date technologies and how to reflect the experience of traditional agri-workers.

In light of the social and academic background, in this study, we developed a Generative Adversarial Networks (GAN)-based deep learning system to gather field images and perform static visual data analyses using AI computing. For GAN, we selected and utilized (1) Anomaly Detection with Generative Adversarial Networks (AnoGAN), and (2) Efficient Generative Adversarial Networks (Efficient GAN) with utilizing total loss score (*loss*) as the *anomaly score* to judge pictures anomaly or not, as one method to judge edible harvests or inedible them, for the following reasons. Generally, GAN-based deep learning systems extract the features of the inputted image data and generate artificial pictures closer to the original inputted picture than past similar methodologies. Theoretically, users can produce non-existent (artificially produced) image data that is as close to the real thing as possible. However, generally speaking, the GAN parameter tunings are relatively difficult compared to other deep learning techniques.

Past studies' results have confirmed the utility of diverse GAN-based methodologies, supporting similar analyses in different academic research fields, to some extent [1]–[10]. Deep-learning-based systems can pick up on subtle disorders or unusual characteristic points that humans might fail to notice. Furthermore, by mixing these with other techniques, we can reconstruct picture or movie files.

Thus, we construct a deep-learning-based system, and estimate its usability and effectiveness with regards to judging Japanese edible and inedible harvests. The targets are (1) kiwi fruits, (2) eggplants, and (3) mini tomatoes in outdoor farmlands. We accumulated images from a variety of real and non-specific Japanese-styled agri-fields. Finally, we numerically present the systems validity and limitations. In future, these achievements could assist the development of automatic agricultural harvesting systems and other high-tech tools.

II. METHODS


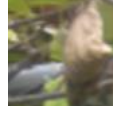


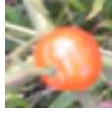
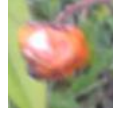
A. Target

For the visual data used in this study, we captured and aggregated original pictures from non-specific outdoor farmlands for test data. Prior to the visual data collection, we consulted agri-managers and workers with five years' experience to mitigate the difficulties of handling samples in farmlands.

Specifically, we first captured the following three categories of image data at Japanese small- to middle-sized outdoor farms, after consultation and negotiation with real farmers using a digital camera, SONY Cyber Shot (SONY Inc., Japan): 1) "Kiwi-Edible," "Kiwi-Inedible" (edible: sample number (n) = 5, inedible: n = 5), 2) "Eggplant-Edible," "Eggplant-Inedible" (for both, n = 5), and 3) "Mini tomato-Edible," "Mini tomato-Inedible" (for both, n = 5).

Table I presents sample picture sets of these harvests, with standardized formats and sizes prepared for the main computation. Considering similar past trials [15]-[18], we standardize the picture sizes (28×28 pixels for AnoGAN, 64×64 pixels for Efficient GAN).

TABLE I. SAMPLE PICTURES OF TARGET AGRI-HARVESTS.

Edible harvests	Inedible harvests
	
	
	

B. Computing Environment

In this study, considering the relatively long computation time for machine learning, we opted to use open, distributed libraries and packages thinking of the speeds of improvement and the technical accuracies. Specifically, we selected and applied TensorFlow, a recent

software library, to appropriately construct the environment with decreased coding cost and time.

For the computing environment, we coded and used programs written in python 3.6 language and PyTorch (Facebook Inc., the U.S.) [15]-[18]. Specifications of the computing environment are Ubuntu 18.04, 13 GB RAM, GPU NVIDIA Tesla K80 12 GB, Instance almost same as "N1-highmem-2". According to the aforementioned academic trends and past results, we judged the methodological set as adequate in the computing-field.

We present the flow of this study: 1) Obtaining and accumulating image data from outdoor farmlands; 2) Executing AnoGAN and Efficient GAN based approaches to the edible and inedible agri-harvest image datasets; and 3) Performing and discussing statistical analyses using numerical data, including confirming the Anomaly score.

C. Computing based on AnoGAN and Efficient GAN

For the usages of AnoGAN and Efficient GAN, as presented in Figs. 1, 2 and 3, we selected and utilized (1) AnoGAN utilizing Deep Convolutional GAN (DCGAN), which is a domain method to learn a Convolutional Neural Network (CNN) utilizing GAN, and (2) Efficient GAN thinking of these specialties [19]-[21].

Fig. 1 presents the flow of the ordinary generating methodology using AnoGAN. Fig. 2 presents the flow for diagnosing inedible harvests using AnoGAN. Fig. 3 presents a model comparing the three cases, (A) AnoGAN, (B) EfficientGAN-Anomaly, and (C) GANomaly.

Case (1) is a common domain GAN-type that is mainly used for various disorder detection. However, since it searches for noise corresponding to the pictures during inference, real-time processing is difficult in reality. Case (2) performs deep learning concerning the encoder that changes pictures into noises. Generally, during the inferences, this approach realizes executions at hundreds of times the speed of past methodologies. This is because DCGAN is basically a specific GAN, which makes the whole network a CNN. Additionally, we can generate pictures from the input vector data by utilizing DCGAN. Finally, we can confirm the change of picture gradually according to the changed vector data after the deep learning. However, we can actually use other GANs, not only DCGAN.

In Figs. 4, 5, and 6, we present the flowcharts of the computing model.

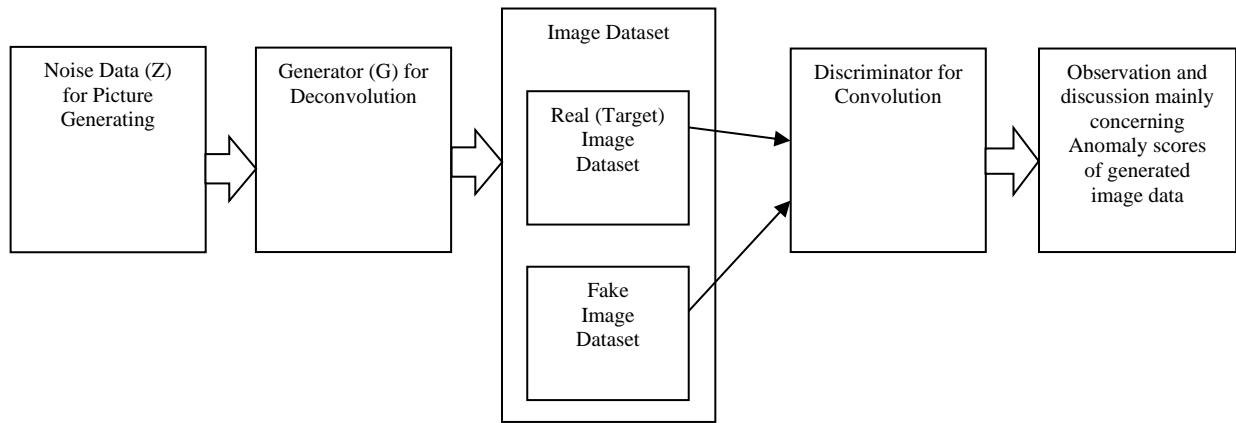


Figure 1. Flow of ordinary generating methodology using AnoGAN.

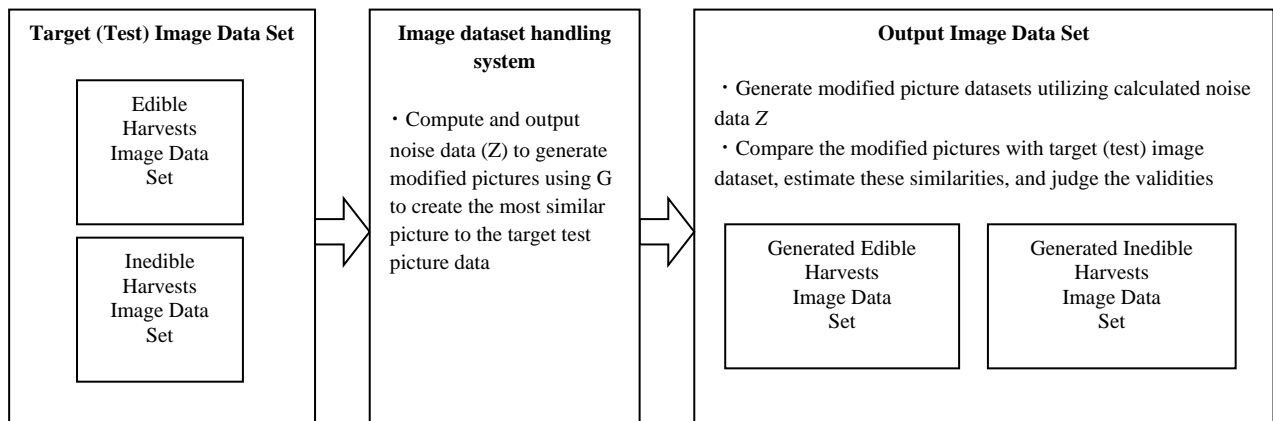


Figure 2. Flow of diagnosing inedible harvests using GANs.

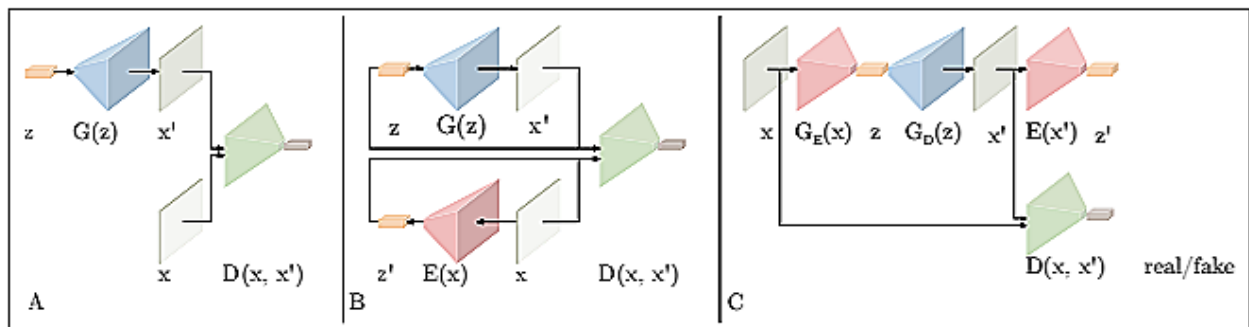


Figure 3. Comparison of the three models. A) AnoGAN, B) Efficient-GAN-Anomaly, and C) GANomaly [12]-[14].

[Pre-processing for following processes: AnoGAN component and Efficient GAN component]

- Making folders and importing files (pandas, numpy, sklearn, matplotlib, torch, etc.)
- Inputting raw picture datasets
- Declaring diverse factors
- Setting and visualizing each image's data for handling
- Setting other variables
- Outputting and storing image data
- Saving and accumulating jpeg-formatted picture files into specific folders for the subsequent handlings

Figure 4. Chart of pre-processing for AnoGAN and Efficient GAN-based deep learning.

■ AnoGAN component

[Pre-processing]

- Preparing raw datasets
- Declaring, setting, and confirming Generator (G)'s class, and the function of Discriminator (D)'s class
- Confirming D 's function
- Generating false pictures
- Inputting and setting false pictures into D and "Data Loader"

[Declaring and setting class of Image data pre-processing]

- Inheriting both pictures' dataset class and PyTorch's dataset class
- Declaring and setting Data Loader's function
- Setting file list, dataset, and Data Loader
- Confirming these functions by several tests
- Initializing network for successive learning

[Setting the function for learning of model]

- Setting method for optimization
- Defining the function for calculating differences

[Main process of learning]

1. Learning of D

- Generating "true label" and "false label"
- Judging true pictures, generating false pictures, and executing judgements
- Calculating the difference between true pictures and false ones
- Executing back-propagation

2. Learning of G

- Generating false pictures for the main judges
- Calculating the difference between true pictures and false ones

3. Data recording and presenting

- Recording and presenting data concerning "loss" and "accuracy" for each deep learning epoch

Figure 5. Chart of AnoGAN-based deep learning processing.

■ Efficient GAN part

[Pre-processing]

- Generating Efficient GAN
- Setting Efficient GAN's network, and executing learnings with it

[Setting Generator]

- Confirming these functions of Generator (G) and Discriminator (D)
- Generating false pictures
- Inputting false picture into D
- Multiplying output data (d_out) and “Sigmoid”, and changing it from 0 into 1
- Declaring Encoder (E), changing pictures into z

[Confirming Encoder's function]

- Encoding fake image data for inputting made by G , and outputting encoded z

[Confirming Data Loader's function]

- Generating file lists, data sets, and Data Loader
- Confirming E 's function

[Learning of the model for deep learning]

- Learning of the model
- Outputting recording, and presenting the D , G , and E
- Initializing the network
- Executing learning and verifying the result
- Visualizing and comparing generated pictures and using these for training

[Making DataLoader]

- Making DataLoader for the testing using test pictures to diagnosing disorders
- Confirming test pictures

[Definition of the Function to obtain “loss” value]

- Calculating the difference between test picture (x) and generated picture ($fake_img$) concerning pixels, and also calculating the absolute values
- Summarizing values of the aforementioned absolute values concerning every mini-batch values
- Inputting test picture (x) and generated picture ($fake_img$) into D , outputting characteristic values' map
- Calculating the difference of characteristic values between test picture (x) and generated picture ($fake_img$), and also calculating the absolute values
- Summarizing values of the aforementioned absolute values concerning every mini-batch values
- Adding two types of $loss$ value concerning every batch
- Calculating the $loss$ of all mini-batches
- Inputting pictures for the disorder detecting, and presenting these pictures
- Encoding the training data into z , and generating G

[Data recording and presenting]

- Recording and presenting data concerning “loss” and “accuracy” for each deep learning epoch

Figure 6. Chart of Efficient GAN based Deep Learning Processing.

Concretely, for AnoGAN, we calculate the differences between input picture datasets and generated data using equation (1), which considers *discrimination loss* and

residual loss. Note that λ can control the balance between *residual loss* and *discrimination loss*. Recent studies in similar fields have set $\lambda = 0.10$, so we use this value.

$$\text{loss} = (1 - \lambda) \times (\text{residual loss}) + \lambda \times (\text{discrimination loss}). \quad (1)$$

After that, we execute the deep learning for the calculation of noise z ; we create and execute the functions of the deep learning and of the disorder-detecting concerning AnoGAN, and then, output the *loss* value.

For Efficient GAN, in this study, we create the encoder E for GAN and have Generator G and Discriminator D at once, in light of both the characteristics and academic trends concerning the computing. Additionally, we have E for training data x of the deep learnings. Thus, we describe the logic for the learning of encoder E 's model with GAN.

First, we present general GAN-oriented equations following (2). We define the result as y judged by our trained teaching data and the judging discriminating module (discriminator) D using generated data. We maximize data output with the discriminant calculated with D .

Further, we select and use Efficient GAN to include the characteristics of Bidirectional Generative Adversarial Network (BiGAN) [22]-[23] to diagnose the pictured disorders. It is a way to learn inverse mappings, and to present that the resulting learned feature representation is mathematically useful for discrimination tasks, competitive with contemporary methods (approaches) to unsupervised and self-supervised feature learning.

We present the theories in the following equation:

$$\begin{aligned} & \sum_{i=1}^M \{l_i * \log(y_i) + (1 - l_i) * \log(1 - y_i)\} \\ &= \sum_{i=1}^M \{l_i * \log D(x) + (1 - l_i) * \log(1 - D(G(z)))\}. \end{aligned} \quad (2)$$

Specifically, related to the discriminator D , generator G , and encoder E , we think about which equations will minimize loss, as well as the three specific types of Efficient GAN. For the loss function concerning D , we calculate the following, similar to existing GANs:

$$- \sum_{i=1}^M \log D(x_i, E(x_i)) - \sum_{j=1}^M \log(1 - D(G(z_j), z_j)). \quad (3)$$

We must also learn functions $(x, E(x))$ and $(G(z), z)$. G will gradually change as D becomes better at discriminating. We want to minimize the second term of (3), however, not to zero. To avoiding such a situation, we minimize the following:

$$\sum_{j=1}^M \log D(G(z_j), z_j). \quad (4)$$

For the loss equation concerning E , we need $E(x) = z$. In short, we want generated random numbers and $E(x)$ after we put picture x into E to be equal. (Generally, G tends to make x and $G(x)$ equal.) Thus, for the ideal situation, $E(x)$ can execute the learning as $E(x) = z$, in the case where $(x, E(x))$ and $(G(z), z)$ can be extremely close to each other in D . Next, we minimize the following:

$$\sum_{i=1}^M \log D(x_i, E(x_i)) + \sum_{i=1}^M \log(1 - D(G(z_i), z_i)). \quad (5)$$

For equation (5), the second term is not concerned with E . We present the loss equation:

$$\sum_{i=1}^M \log D(x_i, E(x_i)). \quad (6)$$

For equation (5), as per the loss equation concerning G , in the early period of the learning, E is likely to be judged correctly without D confusing $(x_i, E(x_i))$ and $(G(z), z)$. However, we have:

$$\log D(x_i, E(x_i)) = \log 1 = 0. \quad (7)$$

Since the loss tends to be zero, the deep learning is easy to process. Thus, minimizing the following equation will contribute to the effectiveness of learning by E :

$$- \sum_{i=1}^M \log(1 - D(x_i, E(x_i))). \quad (8)$$

As mentioned above, we should avoid techniques that deliver isolated, individual executions after a general GAN's deep learning because of the certainty. Then, the below algorithm executes parallel deep learning of discriminator D , generator G , and encoder E with Efficient GAN utilizing the BiGAN model.

III. RESULTS

We obtained sets of output pictures reconstructed from deep learning procedures with the aforementioned AnoGAN and Efficient GAN. Then, we compared input picture sets with output data sets both qualitatively and quantitatively. Table II presents the statistical numerical data sets concerning the three targets: (1) kiwi, (2) eggplant, and (3) mini tomato.

The values in Table II are the average values of the *loss* (anomaly) scores for AnoGAN and Efficient GAN, and their standard deviations. We standardized the sample numbers as the execution time and the computing resources; we needed a rather long time for each execution.

When we utilized AnoGAN, for the (1) – (3) agri-targets, the average Anomaly Score values of edible targets ranged from

129.3 to 405.2, and the SD values ranged from 66.1 to 156.7. For inedible targets, the average Anomaly Score values ranged from 759.4 to 895.2, and the SD values ranged from 190.0 to 290.8. However, with Efficient GAN, these ranges were generally much lower (edible: average, 99.8 - 148.1, SD values, 12.4 - 32.0; and inedible: average, 171.2 - 359.0, SD values, 30.2 - 40.2). We observed large differences and features related to each category from the matrix-formed data.

For the usages of the numerical data in Table II, we thought that the mean values between “Edible” and “Inedible” groups could be the threshold for discriminating edible harvests and inedible them. For instance, in the case of “(1) Kiwi, AnoGAN”, the threshold value is 548.2 ($= (279.9 + 816.5) \div 2$). In future studies, we could invent and use the automatic agri-system (e.g. arm-shaped robots for the harvesting) based method with deep learning judging them.

TABLE II. ANOMALY SCORES OF RECONSTRUCTED PICTURE DATA FROM DEEP-LEARNING-BASED PROCEDURES WITH ANOGAN AND EFFICIENT GAN.

Description	1) Kiwi		2) Eggplant		3) Mini tomato	
	Edible ($n = 5$)	Inedible ($n = 5$)	Edible ($n = 5$)	Inedible ($n = 5$)	Edible ($n = 5$)	Inedible ($n = 5$)
AnoGAN	279.9 (156.7)	816.5 (190.0)	405.2 (70.4)	759.4 (290.8)	129.3 (66.1)	895.2 (288.8)
Efficient GAN	99.8 (12.4)	359.0 (36.5)	148.1 (25.0)	322.7 (30.2)	117.8 (32.0)	171.2 (40.2)

IV. DISCUSSION, CONCLUSION AND FUTURE TASKS

A. Discussion

In this study, we presented the numerical features of the cases of (1) kiwi, (2) eggplant, and (3) mini-tomato. For the data in Table II, we observed the numerical range of anomaly score values from 99.8 to 895.2, which seemed adequate, considering past AnoGAN and Efficient GAN based studies.

The judgements of sample data adequacies made by an agri-worker with five years’ experience suggested that these pictures could be understood using common sense. However, we could not obtain statistically sufficient volumes of visual data; five samples for each case are fewer than in past studies [1]-[10].

For the datasets, we observed the apparent differences concerning the range of anomaly scores and SD values between the edible and inedible datasets, and between the AnoGAN results and the Efficient GAN results. The “(2) eggplant” picture sets had the darkest and least-vivid visual colors, whereas the “(3) mini-tomato” datasets had the most vivid colors, which is why the calculated data values could be investigated with respect to standardization in future studies.

Related to the system’s construction, it was difficult to obtain comments that incorporated a combination of executions (e.g., the type of GAN). Further, we could not determine whether the ratios between the sizes of a captured target and the whole picture were accurate. Additionally, we could not definitively determine whether the statistical datasets were adequate for judging. Other statistical variables, such as the median could be examined.

B. Conclusion

In this study, we constructed and demonstrated a deep-learning-based visual data reconstruction from outdoor agri-site images. We had three typical Japanese agri-harvests: (1) kiwi fruits (kiwi), (2) eggplants, and (3) mini tomatoes. The proposed methods could present statistically significant data tendencies and inclinations. For the findings of this research, we found these anomaly score-based methods for judging edible and inedible harvest as practically useful to some extent; there have not been such studies. However, also have limitations. GANs, such as AnoGAN and Efficient GAN methodologies using our originally captured picture files, may have various future practical systematic usages like automatic image datasets discriminations, etc. Future works would provide further validation for varieties of categorizing and analyzing agri-targets and background conditions. We would also check the accuracy of the systems and the appropriateness for other computing patterns.

C. Future Tasks

We believe the system could be more efficient and practical, and we would present more useful outputs to agricultural system developers in future. The system has benefits not only for agri-workers and agri-managers, however, also for security guards. We hope that progressive promising methodologies will be widely applied to real agri-working sites. Future research would provide further confirmation concerning agri-harvests. Specifically, for our long-term studies, we aim to check and heighten the system durability, investigate long-term performance, and consider other methodological patterns. Our

results could be used for automatic systems to support both indoor and outdoor farmlands by improving their accuracy in selecting edible and inedible agri-harvest. We hope the aforementioned methodologies will be applied to real agri-sites to promote agricultural productivity.

ACKNOWLEDGMENT

Our heartfelt appreciation goes to the members of Mitsui Fudosan Co., Ltd., Kashiwano-Ha Farm Inc., Kashiwa-shi, and The University of Tokyo who provided considered support, feedback, and comments.

REFERENCES

- [1] T. Schlegl, P. Seeböck, P. Waldstein, S. M. Langs, and U. Schmidt-Erfurth, “f-AnoGAN: Fast unsupervised anomaly detection with generative adversarial networks. Medical image analysis,” vol. 54, pp. 230–44, January 2019.
- [2] H. Zenati, C. S. Foo, B. Lecouat, G. Manek, and V. R. Chandrasekhar, “Efficient gan-based anomaly detection,” arXiv preprint arXiv:1802.06222, May 2018.
- [3] M. Kimura, “PNUNet: Anomaly Detection using Positive-and-Negative Noise based on Self-Training Procedure,” arXiv preprint arXiv:1905.10939, May 2019.
- [4] L. Deecke, R. Vandermeulen, L. Ruff, S. Mandt, and M. Kloft, “Image anomaly detection with generative adversarial networks,” *Proceedings of Joint European Conference on Machine Learning and Knowledge Discovery in Databases*, Springer, Cham, pp. 3–17, September 2018.
- [5] A. Berg, J. Ahlberg, and M. Felsberg, “Unsupervised Learning of Anomaly Detection from Contaminated Image Data using Simultaneous Encoder Training,” arXiv preprint arXiv:1905.11034, May 2019.
- [6] C. Zhang and Y. Chen, “Time Series Anomaly Detection with Variational Autoencoders,” arXiv preprint arXiv:1907.01702, July 2019.
- [7] Y. Lu and P. Xu, “Anomaly detection for skin disease images using variational autoencoder,” arXiv preprint arXiv:1807.01349, July 2018.
- [8] S. Akcay, A. Atapour-Abarghouei, and T. P. Breckon, “Ganomaly: Semi-supervised anomaly detection via adversarial training,” *Proceedings of Asian Conference on Computer Vision*. Springer, pp. 622–637, December 2018.
- [9] H. Zenati, C. S. Foo, B. Lecouat, G. Manek, and V. R. Chandrasekhar, “Efficient ganbased anomaly detection,” arXiv preprint arXiv:1802.06222, February 2018.
- [10] T. Schlegl, P. Seeböck, S. M. Waldstein, U. Schmidt-Erfurth, and G. Langs, “Unsupervised anomaly detection with generative adversarial networks to guide marker discovery,” *Lecture Notes in Computer Science Subseries (Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, 10265 LNCS, https://doi.org/10.1007/978-3-319-59050-9_12, pp. 146–147, March 2017.
- [11] R. Wang, J. Zhang, W. Dong, J. Yu, C. J. Xie, R. Li, and H. Chen, “A Crop Pests Image Classification Algorithm Based on Deep Convolutional Neural Network,” *Telkomnika*, vol. 15, no. 3, pp. 1239–1246, January 2017.
- [12] N. Zhu, X. Liu, Z. Liu, K. Hu, Y. Wang, J. Tan, and Y. Guo, “Deep learning for smart agriculture: Concepts, tools, applications, and opportunities,” *International Journal of Agricultural and Biological Engineering*, vol. 11, no. 4, pp. 32–44, July 2018.
- [13] S. Kawakura and R. Shibasaki, “Development of AI-based System for Classification of Objects in Farms using Deep Learning by Chainer and a Template-Matching based Detection Method,” *Journal of Advanced Agricultural Technologies*, vol. 6, no. 3, pp. 175–179, September 2019.
- [14] S. Kawakura and R. Shibasaki, “Accuracy Analyses for Detecting Small Creatures using an OpenCV-based System with AI for Caffè’s Deep Learning Framework,” *Journal of Advanced Agricultural Technologies*, vol. 6, no. 3, pp. 166–170, September 2019.
- [15] A. Gulli, S. Pal, “Chokkan Deep Learning —Python×Keras De Aidea Wo Katachi Nisuru Recipe,” Tokyo, Japan: Ohmesha Inc., 2018.
- [16] S. To, “Genba De Tsukaeru PyTorch Kaihatsu Nyumon,” Japan: SHOEISHA Inc., 2018.
- [17] T. Niimura, “TensorFlow De Hajimeru DeepLearning Jissou Nyuumon,” Tokyo, Japan: Impress Inc., 2018.
- [18] Y. Ogawa, “PyTorch Ni Yoru Hatten Deep Deep Learning,” Tokyo, Japan: Mainavi Publishing Inc., 2019.
- [19] T. Schlegl, P. Seeböck, S. M. Waldstein, U. Schmidt-Erfurth, and G. Langs, “Unsupervised anomaly detection with generative adversarial networks to guide marker discovery,” *Proceedings of International conference on information processing in medical imaging*, Springer, Cham, pp. 146–157, June 2017.
- [20] H. Zenati, C. S. Foo, B. Lecouat, G. Manek, and V. R. Chandrasekhar, “Efficient gan-based anomaly detection,” arXiv preprint arXiv:1802.06222, February 2018.
- [21] V. Dumoulin, I. Belghazi, B. Poole, O. Mastropietro, A. Lamb, M. Arjovsky, and A. Courville, “Adversarially learned inference,” arXiv preprint arXiv:1606.00704, February 2017.
- [22] J. Donahue, P. Krähenbühl, and T. Darrell, “Adversarial feature learning,” arXiv preprint arXiv:1605.09782, May 2016.
- [23] H. Zenati, C. S. Foo, B. Lecouat, G. Manek, and V. R. Chandrasekhar, “Efficient gan-based anomaly detection,” arXiv preprint arXiv:1802.06222, May 2019.