

예지혜

이화여자대학교 통계학과

212STG18

Abstract

이미지 이상치 탐지는 이상치 이미지의 데이터가 정상 이미지에 비해 매우 적은 불균형 데이터라는 점에서 일반적인 이미지 분류 문제와는 차이가 있다. 본 연구에서는 GAN을 이미지 이상치 탐지에 이용한 모델인 AnoGAN과 GANomaly의 구조를 비교하고, 2개의 데이터에 적용한다. 실험 데이터로는 위성 이미지 중 구름이 걸린 이미지를 탐지하는 데이터와 제조 결함을 탐지하는 데이터를 사용하였다. 두 이미지에 각 모델을 적용 후 성능을 비교하고, 이미지를 어떻게 처리하였는지 시각화하여 모델 학습의 결과를 해석한다. 결과적으로 GANomaly가 AnoGAN보다 우수하였으나, 이는 데이터 특성과 모델 설정에 따라 차이가 있을 수 있다.

1 Introduction

컴퓨터 비전에서 이미지 분류 분야는 이미 사람의 능력을 넘어설 만큼 발전되어왔다. 대부분 지도 학습 방식으로, 각 클래스에 대한 충분한 학습 데이터가 주어져야 해당 클래스의 정보를 학습하여 테스트 데이터를 분류해낼 수 있다. 따라서 어떠한 클래스에 대한 데이터가 충분하지 않다면 효과적으로 학습할 수 없다. 이상 탐지 분야의 많은 경우가 여기에 속하는데, 이상치 데이터는 자주 관측되지 않으므로 데이터 또한 충분히 모으기 어렵다. 예를 들어, 전 세계의 극히 소수만이 가지는 질병이나, 비행기 테러리스트의 무기를 잡아내는 일은 매우 드문 경우이므로 데이터가 충분하지 않다. 하지만, 이 드문 케이스를 적발해내지 못한다면 큰 피해를 얻게 되기 때문에 이미지 이상치 탐지는 매우 중요한 문제이다. 이 외에도 이미지 이상치 탐지는 제조업 분야에서 제품의 품질을 검사하거나, 방범용 CCTV를 통해 범죄자를 잡거나 쓰러진 사람을 탐지해내는 데에 사용되고 있다.

최근, 이미지 이상치 탐지의 데이터 불균형 문제를 해결하기 위해 많은 연구가 진행되어왔다. 크게, GAN을 이용해 정상 이미지와 비슷한 데이터를 재생산해내는 방식과 normal features가 한 점에 모이도록 하는 one class classification 방식, feature의 확률 분포를 학습하는 feature matching 방식이 있다. 이 중 가장 먼저 연구된 GAN 기반의 AnoGAN과 GANomaly의 모델 구조를 이해하여 GAN을 이미지 이상치 탐지에 어떻게 적용했는지 알아보고, 새로운 데이터에 적용해보고자 한다.

2 Structure of Generative Adversarial Networks (GAN)

GAN은 2014년 Goodfellow et al.가 처음 제안한 모델로, [1] 실제에 가까운 이미지를 생성해내기 위한 모델이다. 이미지를 생성하는 Generator와 진짜와 가짜 이미지를 분류하는 Discriminator를 적대적으로 학습시켜 Generator의 성능을 높이는 구조이다. generator G는 latent space에서 생성된 노이즈인 1d vector z 를 실제 이미지 공간에 위치한 2d images x 로 매핑하며 실제 이미지의 분포인 $p_{data}(x)$ 를 학습한다. discriminator D는 2d image를 single scalar 값으로 학습하며 해당 이미지가 실제 이미지인지 가짜 이미지인지 학습하는 분류기이다. D와 G는 상호 반복적으로 학습되며 다음과 같은 목적 함수를 최적화한다. 실제로 코드에서는 generator loss와 discriminator loss를 정의하여 모두 최소화 문제로 바꾸어 진행한다.

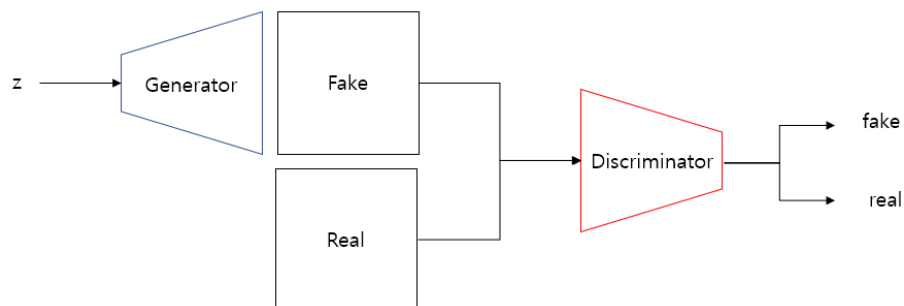
$$\min_G \max_D V(D, G) = E_{x \sim p_{data}(x)} [\log D(x)] + E_{x \sim p_z(z)} [\log (1 - D(G(z)))]$$

DCGAN은 주로 이미지 생성에 쓰이는 GAN에 기존의 컴퓨터 비전에서 우수한 성능을 보이고 있는 CNN 구조를 적용한 모델로, GAN에 CNN을 적용하려면 어떤 요소를 사용하는 게 가장 좋은 성능을 보이는지 Radford et al.이 제안하였다. [2] GAN의 fully connected layers 대신 convolution layers와 batch-normalization을 사용한다. generator의 활성화 함수에는 ReLU를 사용하지만, discriminator에는 LeakyReLU를 사용하는 것을 권장한다. 또한, Generator에서 다운샘플링을 할 때는 maxpooling 대신 stride를 이용한다. 본 연구에서 사용한 AnoGAN과 GANomaly는 모두 DCGAN을 기반으로 하고 있다.

3 Structure of AnoGAN

첫 번째로 살펴볼 모델은 Thomas Schlegli가 2017년 제안한 AnoGAN으로, [3] GAN을 이상치 탐지에 처음으로 사용한 모델이다. 정상 이미지만을 가지고 학습한 Generator는 이상치 이미지를 잘 생성해내지 못할 것이고, 이상치 이미지를 정상 이미지의 latent space에 적절히 배치하지 못할 것이라는 아이디어를 기반으로 한다. 따라서 GAN의 기본적인 구조를 정상 이미지만으로 학습한 후, 이상치 이미지를 학습된 latent space에 매핑하고, Generator로 이미지를 생성하여, 얼마나 그 성능이 떨어지는지를 loss로 정의한다.

Step 1. train G and D



Step 2. find z for each new images

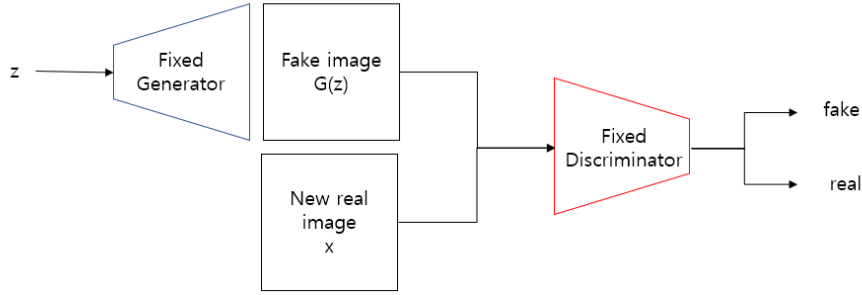


그림 1 AnoGAN의 구조. 모델 학습 순서대로 시각화 하였으며, Step 1에서는 G와 D를, Step 2에서는 z를 학습한다.

AnoGAN은 크게 2단계로 나눌 수 있다. 첫 번째 단계는, DCGAN과 동일한 구조의 모델을 정상 이미지로만 학습시킨다. 정상 데이터만 사용할 뿐, 모델의 모든 구조와 손실 함수는 DCGAN과 동일하다. 두 번째 단계에서는 새로운 이미지를 기존의 latent space에 매핑시킨다. 새로운 이미지에 가장 적합한 latent variable z를 찾는 과정이다. 앞서 학습시킨 DCGAN의 G와 D의 모든 파라미터를 고정한 후, 랜덤으로 발생시킨 z만을 업데이트하는 것이다. 이를 위해 2개의 loss를 정의한다.

$$\text{residual loss} : L_R(z_\gamma) = \sum |x - G(z_\gamma)|$$

$$\text{discrimination loss} : L_D(z_\gamma) = \sum |f(x) - f(G(z_\gamma))|$$

residual loss는 새로운 이미지와 Generator로 생성된 이미지가 시각적으로 얼마나 비슷한지 평가한다. 이 loss가 작아지려면, 우선 새로운 이미지가 latent space에 잘 매핑되어 적절한 z_γ 을 찾아야 하고, 이를 잘 생성해내는 G를 통과하여 원본 이미지와 유사한 이미지가 생성되어야 한다. 이상치 이미지라면 z_γ 를 찾기도 어렵고, 이를 통해 생성된 이미지 또한 정상 이미지의 특성을 가질 것이므로 loss가 높아질 수밖에 없다.

discrimination loss는 feature matching 아이디어를 사용한 것으로, 새로운 이미지를 latent space에 잘 매핑하기 위한 목적이다. 앞서 학습된 discriminator를 feature extractor로 활용하여 x와 $G(z_\gamma)$ 의 feature 정보를 풍부하게 추출하기 위함이다.

$$L(z_\gamma) = (1 - \lambda) \cdot L_R(z_\gamma) + \lambda \cdot L_D(z_\gamma)$$

$$A(x) = (1 - \lambda) \cdot R(x) + \lambda \cdot D(x)$$

이 loss들의 가중치합을 두 번째 단계의 최종 loss로 정의하여 새로운 이미지가 들어올 때마다 해당 이미지에 맞는 latent variable z를 학습시키고, 마지막 단계에서의 이 loss값을 anomaly score로 활용한다. 또한, 생성된 이미지와 실제 이미지의 차이를 의미하는 residual image $x_R = |x - G(x_T)|$ 를 통해 어떤 부분이 학습되지 못해 이상치로 판단되었는지 이해할 수 있다.

4 Structure of GANomaly

두 번째로 살펴볼 GANomaly는 2018년 Samet Akcay가 제안한 모델로, [4] AnoGAN의 모델 구조를 기반으로 하나 각 이미지마다 latent variable을 학습시키지 않도록 다른 장치를 둔 모델이다. 이 모델은 latent variable z 를 랜덤하게 생성해서 시작하는 것이 아니라, 실제 이미지를 압축하는 과정부터 시작하기 때문에, 정상 이미지만을 입력으로 학습하여 모델을 고정한 후, 새로운 이미지에 대해서는 모델을 단순히 통과시켜 anomaly score를 계산하기만 하면 된다.

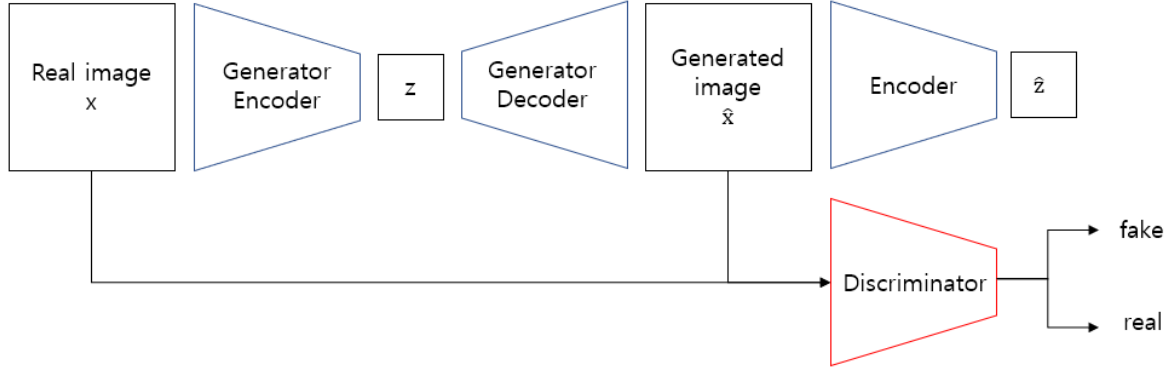


그림 2 GANomaly의 구조. 학습 과정에서는 3개의 Encoder와 1개의 Decoder를 학습하며, 새로운 이미지에 대해서는 이미 학습된 모델을 통해 anomaly score를 계산한다.

GANomaly는 크게 3개의 subnetwork로 나눌 수 있다. 입력과 비슷한 이미지를 생성해내는 Generator와 이를 다시 압축하는 Encoder, 실제와 가짜 이미지를 분류하는 Discriminator가 그것이다. Generator는 또다시, Encoder와 Decoder로 이루어지기 때문에 이들은 앞으로 G_E , G_D 라 칭하겠다. 따라서 전체적으로 봤을 때는 3개의 Encoder 구조와 1개의 Decoder 구조로 이루어졌다고 할 수 있다.

먼저 Generator는 입력 이미지 x 를 가장 잘 표현하는 1d vector의 z 로 압축시키고, 이를 통해 원본 이미지와 유사한 $\hat{x} = G_D(z) = G(x)$ 를 생성해내는 것이 목표이다. 정상 이미지의 contextual 정보를 학습하는 구간이다. 따라서 이 구간에서 최소화할 loss를 contextual loss라 정의하고, 그 식은 다음과 같다.

$$\text{contextual loss} : L_{con} = E_{x \sim p_x} \|x - G(x)\|_1$$

두 번째로 Encoder는 Generator에서 생성된 이미지를 다시 \hat{z} 로 압축시킨다. G_E 의 구조와 완전히 같지만, 다른 파라미터를 가지는데, 이를 통해 정상 이미지의 feature representation을 학습한다. Generator에서 정상 이미지의 재생산을 학습한다면, Encoder에서 정상 이미지의 feature 압축을 학습하는 것이다. 따라서 이 과정에서의 loss는 encoder loss라 정의하며 다음과 같다.

$$\text{encoder loss} : L_{enc} = E_{x \sim p_x} \|G_E(x) - E(G(x))\|_2$$

쉽게 표현하면 z 와 \hat{z} 의 거리를 최소화하는 것이다. 이 loss는 학습이 끝난 후 anomaly score로 활

용되는데, 이상치 이미지는 제대로 된 z 로 압축하기 어려우며, 그를 통해 생성된 \hat{x} 또한 정상 이미지의 문맥 정보만을 가지고 있기 때문에 이상치 정보는 누락된 상태이다. 따라서, 이를 다시 압축한 \hat{z} 는 z 와 차이가 클 수밖에 없다.

마지막 Discriminator는 AnoGAN의 Discriminator와 동일하며 loss 함수 또한 동일하다. 적대적 학습을 위한 장치이며, D의 intermediate layer의 출력에 해당하는 함수 f 를 통해 각 이미지의 feature를 추출해낸다. loss 함수는 다음과 같다.

$$\text{adversarial loss} : L_{adv} = E_{x \sim p_x} \|f(x) - E_{x \sim p_x} f(G(x))\|_2$$

세 가지 loss의 가중치 합을 전체 모델의 loss로 정의하며, 논문의 실험에 따르면 contextual loss를 나머지 가중치의 50배로 설정했을 때 가장 성능이 좋았다고 한다.

$$L = w_{adv}L_{adv} + w_{con}L_{con} + w_{enc}L_{enc}$$

GANomaly는 꽤 복잡한 구조를 가지지만, 사실상 AnoGAN의 G와 D에 Encoder 하나를 추가했을 뿐이다. 각 이미지의 latent variable z 를 학습하는 과정을 대체하기 위해, 정상 이미지를 학습할 때부터 입력 이미지를 1d vector인 z 로 압축하는 과정을 추가하였고, 이를 재생산된 이미지에도 적용함으로써 그 차이를 anomaly score로 활용한다. 또한, 이 anomaly score를 직관적으로 이해하기 위해 전체 테스트 데이터의 score를 활용해 min-max scaling 시켜, 0과 1 사이의 값으로 이 데이터가 얼마나 이상치에 가까운지 판단한다.

$$A(\hat{x}) = \|G_E(\hat{x}) - E(G(\hat{x}))\|_1$$

$$s'_i = \frac{s_i - \min(S)}{\max(S) - \min(S)} \text{ when } S = \{s_i : A(\hat{x}_i), \hat{x}_i \in \text{test set } \hat{D}\}$$

5 Experiments

5.1 Data

현실에서도 이상치 탐지를 필요로 하는 이미지 데이터 2개를 적용하였으며, 캐글에서 다운로드할 수 있다. 구현한 코드는 <https://github.com/jihye0115/2022-Anomaly-Detection-using-GAN>에서 참고할 수 있다.

Cloud and non-cloud data는 위성으로 지면을 촬영한 데이터로, [5] 구름이 껴 있으면 지면을 제대로 촬영할 수 없어, cloud data를 이상치로 하는 데이터이다. 학습 데이터로는 구름이 없는 non-cloud 데이터를 총 1,400개 사용하였으며, 검증 데이터로는 cloud data 100개와 학습에 사용하지 않은 non-cloud data 100개를 사용하였다. 이미지 사이즈로는 256부터 64까지 시도하였으며, 64일 때의 성능이 오히려 좋아 이후 결과에서 소개하겠다.

Casting data는 금속 공정에 사용되는 주형 이미지 데이터로, [6] 여기에 금속 액상을 부어 굳히르

로 주형의 결함은 곧 주조 결함으로 이어진다. 대부분 이미지로 비슷하게 생겼기 때문에, 미세한 이상치를 잡아내야 한다. 학습 데이터로는 정상 이미지 2,875개를 이용하였으며, 검증 데이터로는 학습에 사용하지 않은 정상 이미지 100개, 이상치 이미지 100개를 사용하였다. 이미지 사이즈로는 256부터 64를 시도하였으나, 256 사이즈의 경우, 부족한 RAM으로 끝까지 학습시킬 수 없었다. 따라서 128 사이즈를 선택하였고, channel 개수는 3으로 지정하면 오히려 이상치 이미지의 loss가 작게 드러나는 경향이 있어 1로 설정하였다. 이미지의 색상이 거의 회색 계열이기 때문에, 오히려 1개의 channel로 학습시키는 것이 안정적이다.

5.2 Results

Cloud and non-cloud data

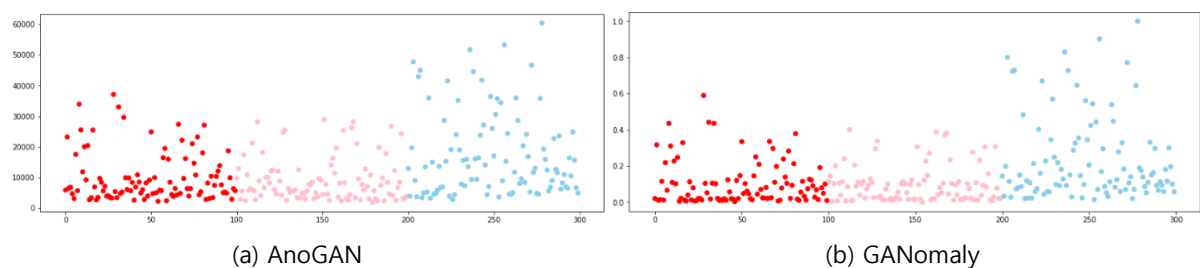


그림 3 cloud data를 AnoGAN과 (a) GANomaly에 (b) 적용한 결과이다. 빨간색은 train normal image, 분홍색은 test normal image, 하늘색은 test abnormal image이다.

	AnoGAN		GANomaly	
정상 이미지				
이상치 이미지				

그림 4 정상 이미지와 이상치 이미지를 각 모델이 재생산해낸 이미지이다. 각 칸 내에서 왼쪽은 실제 이미지, 오른쪽은 재생산된 이미지이다.

두 모델의 결과는 비슷해 보이지만, GANomaly의 결과가 (b) 몇몇 이상치 이미지를 제외하고는 정상 이미지들은 작은 값에 대부분 몰려 있는 것을 확인할 수 있다. 각 모델에 정상 이미

지와 이상치 이미지를 통과시켜 재생산한 이미지를 보면, 어떠한 모양을 감지했다기보다, 회색의 단색 이미지를 생성하였다. 따라서, 구름이 많을수록 이미지가 하얀 색상을 띠기 때문에 loss가 컸을 것으로 보인다. 하지만 정상 이미지 중 어두운 산이 아닌, 강이나 밝은 길이 있는 이미지의 경우, 즉 밝은 색상이 많이 분포한 이미지의 경우 이를 구름으로 탐지하여 이상치로 분류되었을 가능성이 높다. 반대로 구름이 있더라도 회색 비구름이 끼거나, 옅은 안개로 인해 어두운 지상의 이미지가 뿌옇게 보일 뿐이라면 정상 이미지로 분류될 것이다. 데이터 특성상, 학습 데이터에서 어두운 땅의 이미지를 많이 학습하고, 특정한 형태를 학습하는 것이 아니기 때문에 실제에 가까운 이미지를 생성하기보단, 이상치를 판단할 기준으로 단순히 색상을 선택한 것으로 해석할 수 있다.

Casting data

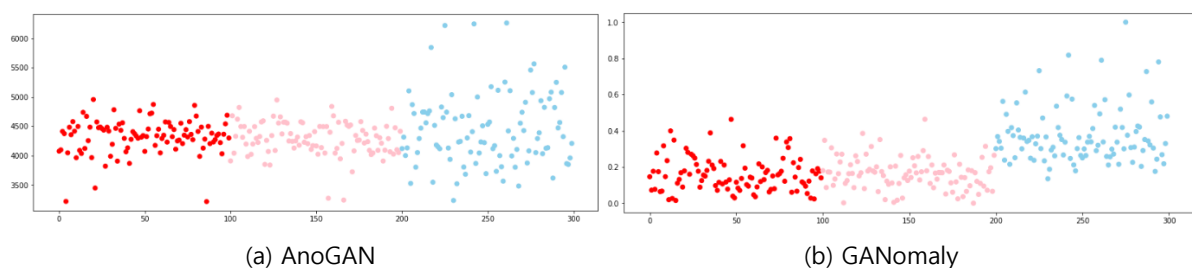


그림 5 casting data를 AnoGAN과 (a) GANomaly에 (b) 적용한 결과이다. 빨간색은 train normal image, 분홍색은 test normal image, 하늘색은 test abnormal image이다.

Casting data는 AnoGAN보다 GANomaly가 훨씬 나은 것을 한눈에 확인할 수 있다. AnoGAN에서는 이상치의 loss가 매우 퍼져있고, 정상 이미지보다도 낮은 loss를 갖는 데이터가 많은데, GANomaly에서는 loss가 확연히 크고, 기준치를 잘 잡으면 80% 이상의 정확도를 가질 수 있다.

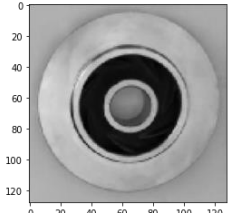
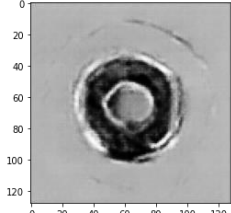
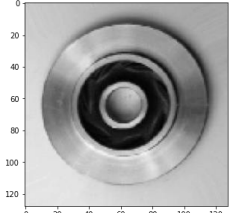
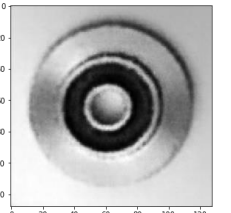
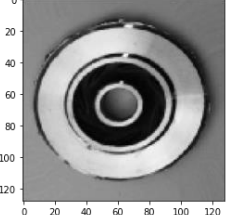
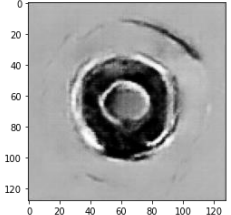
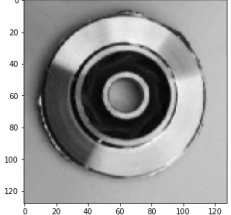
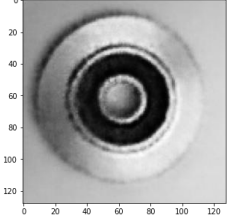
	AnoGAN		GANomaly	
정상 이미지				
이상치 이미지				

그림 6 정상 이미지와 이상치 이미지를 각 모델이 재생산해낸 이미지이다. 각 칸 내에서 왼쪽은 실제 이미

지, 오른쪽은 재생산된 이미지이다.

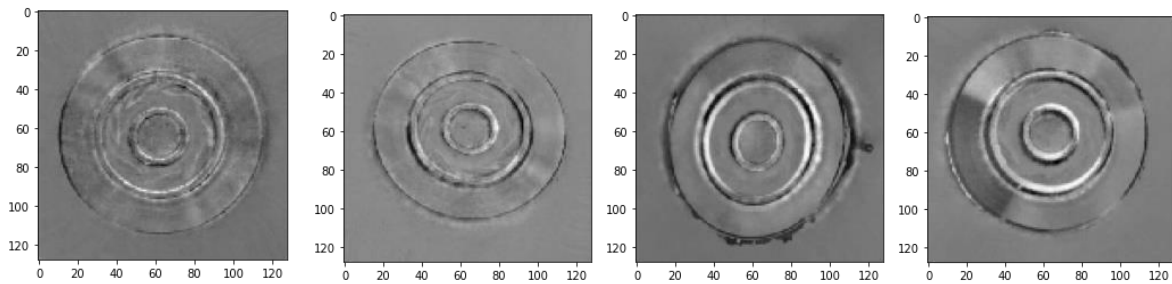


그림 7 Residual image. 왼쪽 2개는 정상 이미지의 residual image이고, 오른쪽 2개는 이상치 이미지의 residual image이다.

재생산된 이미지를 통해 그 이유를 알 수 있는데, 우선 AnoGAN에 비해 GANomaly의 재생산된 이미지가 훨씬 실제 이미지와 유사하다. 육안으로 구별하기 힘들 정도로 실제에 가까운 이미지를 생산해내고 있다. residual image를 보면, 정상 이미지는 윤곽의 미세한 부분만 남았는데, 이상치 이미지는 윤곽이 비교적 선명하게 남아있고, 고르지 못한 부분들이 남아 loss를 크게 만들고 있다.

6 Conclusion

AnoGAN과 GANomaly 두 모델의 구조를 이해하고, GAN을 이용해 이미지 이상치를 어떻게 탐지할 수 있는지 알아보았다. 이를 기반으로 참고한 코드들을 수정하여 2개의 데이터에 대해 실험을 진행하였으며, 데이터 특성에 따라 성능이 매우 차이 났지만, 대체로 AnoGAN보다 GANomaly가 성능이 조금 더 좋았다. 이 모델들은 기존 연구에서 mnist나 cifar 데이터와 같이 특정 형태를 가지는 하나의 클래스를 학습한 후 다른 형태를 가지는 클래스를 이상치로 분류해내는 작업은 잘 수행하였는데, 본 연구에서 사용한 cloud data의 경우 구름이 없는 이미지를 정상 이미지로 학습해야 했기 때문에 성능이 좋지 않았다. casting data는 육안으로 보았을 때는 대부분 비슷한 원형의 형태를 띠고 있고, 약간의 결함을 잡아내야 하는데, 이는 GANomaly가 실제에 가까운 이미지를 잘 생산해 냄으로써 탐지할 수 있었다.

이미지 사이즈, Generator 구조, channel 수, loss의 가중치에 대해서도 여러 실험을 진행하였으나, GPU와 RAM 사용량 제한으로 풍부한 실험은 할 수 없었다. 충분한 자원과 시간을 가지고 이 실험을 비교한다면 같은 모델 구조에서도 더 좋은 성능을 가지는 모델을 찾을 수 있을 것이다. 두 모델은 이 데이터들을 학습시키기에 한계가 있었으나, cloud와 casting data에서 이상치를 탐지하는 것은 실제 산업에서도 사용하는 실용적인 주제이므로 이 데이터의 특성에 맞는 최근의 모델을 적용하여 성능을 비교해보는 것 또한 추후의 좋은 연구 주제가 될 것이다.

References

- [1] Goodfellow, I., Pouget-Abadie, J., Mirza, M., Xu, B., Warde-Farley, D., Ozair, S., Courville, A., Bengio, Y.: Generative adversarial nets. In: Advances in Neural Information Processing Systems. (2014)
 - [2] Radford, A., Metz, L., Chintala, S.: Unsupervised representation learning with deep convolutional generative adversarial networks. arXiv:1511.06434 (2015)
 - [3] Schlegl, T., Seebock, P., Waldstein, S.M., Schmidt-Erfurth, U., Langs, G.: Unsupervised anomaly detection with generative adversarial networks to guide marker discovery. arXiv:1703.05921 (2017).
 - [4] Samet Akcay, Amir Atapour-Abarghouei, Toby P. Breckon: GANomaly: Semi-Supervised Anomaly Detection via Adversarial Training. arXiv:1805.06725 (2018)
 - [5] Cloud and Non-Cloud Images (Anomaly Detection). <https://www.kaggle.com/datasets/ashoksrinivas/cloud-anomaly-detection-images>
 - [6] Casting product image data for quality inspection. <https://www.kaggle.com/datasets/ravirajsinh45/real-life-industrial-dataset-of-casting-product>
- AnoGAN 코드 참고 : https://github.com/soomin9106/Deep-Learning/blob/main/AnoGAN/DCGAN_AnnoGAN.ipynb
- GANomaly 코드 참고 : https://github.com/leafinity/keras_ganomaly/blob/master/ganomaly.ipynb