Unsupervised Anomaly Detection Overview using Convolutional Autoencoders

2021/02/02

Ho Seong Lee (hoya012)

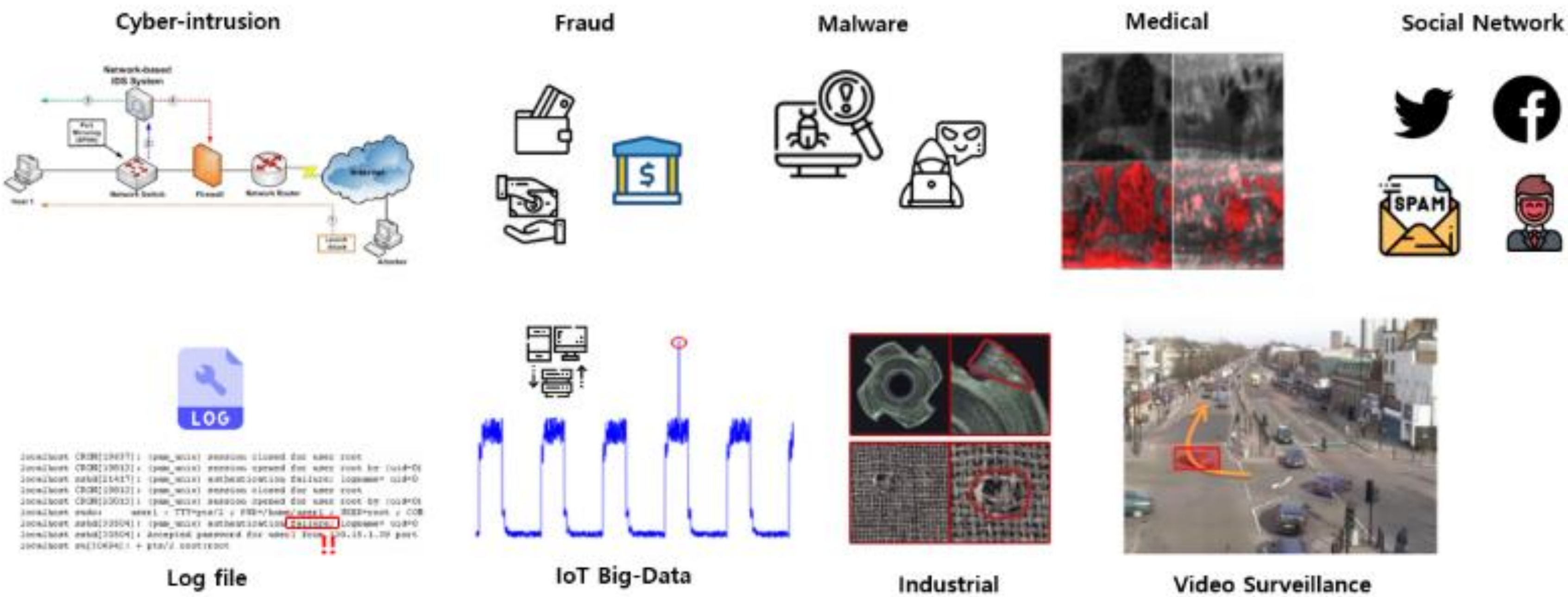
Cognex Deep Learning Lab

Research Engineer

Contents

What is Anomaly Detection?

- Detecting anomalous regions in images or videos or time-series data
- Today, we will focus anomaly detection in industrial images



Reference: https://hoya012.github.io/blog/anomaly-detection-overview-1/

Definition of Anomaly Detection

- I wrote technical blog post about anomaly detection
- I recommend reading this post if you want to know more about anomaly detection

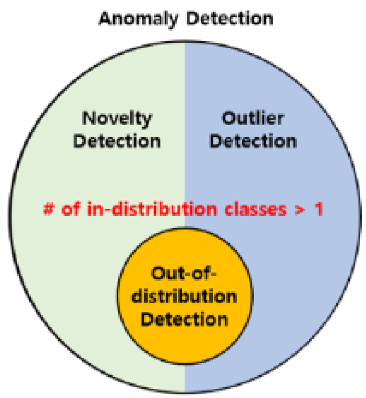
1. 학습시 비정상 sample의 사용 여부 및 label 유무에 따른 분류

용어	정상 sample	비정상 sample
Supervised Anomaly Detection	학습에 사용	학습에 사용
Semi-Supervised (One-Class) Anomaly Detection	학습에 사용	**학습에 사용 X**
Unsupervised Anomaly Detection	**모름.(label이 없음)** 학습에 사용하는 데이터의 대다수가 정상 sample일 것이라고 가정.	

2. 비정상 sample 정의에 따른 분류

용어	비정상 sample
Novelty Detection	지금까지 등장하지 않았지만 충분히 등장할 수 있는 sample
Outlier Detection	지금까지 등장하지 않았고 앞으로도 등장할 가능성이 없는, 데이터에 오염이 발생했을 가능성이 있는 sample

3. 정상 sample의 class 개수에 따른 분류



[Anomaly Detection 관련 3가지 용어의 분류 방법 정리]

Reference: https://hoya012.github.io/blog/anomaly-detection-overview-1/

Unsupervised Anomaly Detection 주요 방법론 소개

- Generative Adversarial Networks 기반 연구
  - "Unsupervised Anomaly Detection with Generative Adversarial Networks to Guide Marker Discovery" (AnoGAN), 2017 IPMI
  - "GANomaly: Semi-Supervised Anomaly Detection via Adversarial Training", 2018 ACCV
  - "Skip-GANomaly: Skip Connected and Adversarially Trained Encoder-Decoder Anomaly Detection", 2019 IJCNN
  - "f-AnoGAN: Fast unsupervised anomaly detection with generative adversarial networks", 2019 MIA

Unsupervised Anomaly Detection 주요 방법론 소개

- Generative Adversarial Networks 기반 연구 - AnoGAN
    - Train GAN using normal samples, and fix Generator and Discriminator
    - Search for a latent sample that reproduces a given input image and manages to fool discriminator

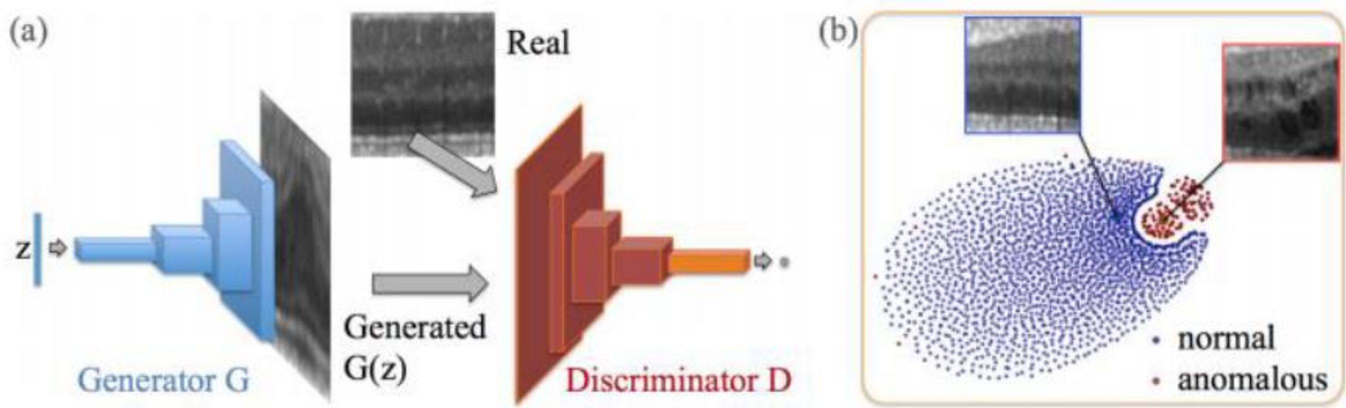
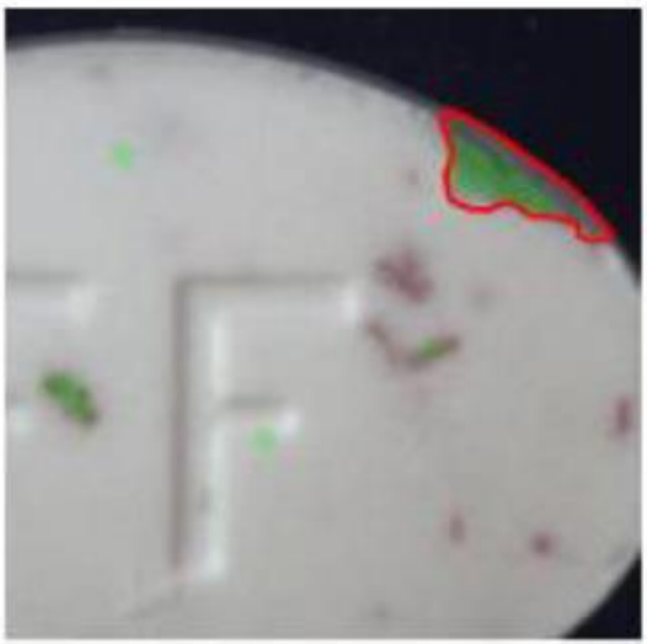
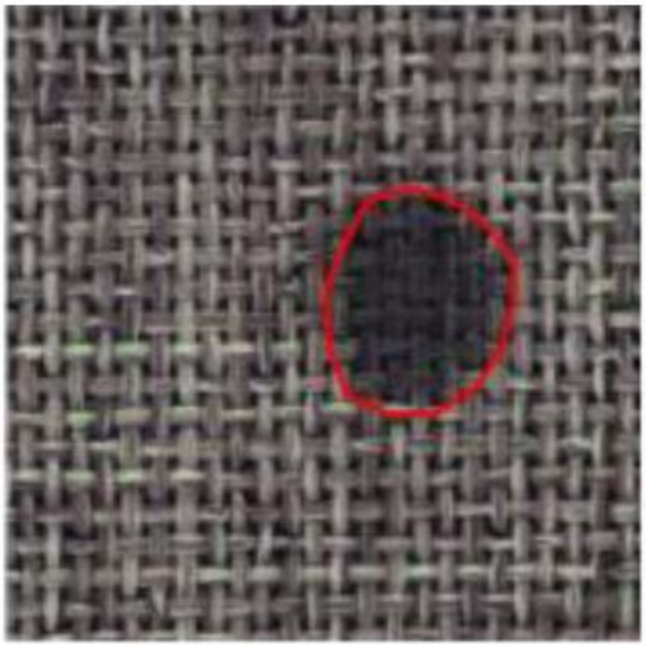


Fig. 2. (a) Deep convolutional generative adversarial network. (b) t-SNE embedding of normal (blue) and anomalous (red) images on the feature representation of the last convolution layer (orange in (a)) of the discriminator.

Unsupervised Anomaly Detection 주요 방법론 소개

- Generative Adversarial Networks 기반 연구 - AnoGAN
  - GAN의 고질적인 문제인 mode collapse가 자주 발생하고 학습이 불안정함
  - 데이터 셋의 복잡도가 큰 경우 (ex, object의 모양이나 방향이 다양) 성능이 매우 떨어지는 문제





Unsupervised Anomaly Detection 주요 방법론 소개

- CNN Feature Dictionary 기반 연구
  - "Anomaly Detection in Nanofibrous Materials by CNN-Based Self Similarity", 2018 Sensors
  - Use feature descriptors obtained from ImageNet pre-trained CNN
  - Sliding window-based approach → for large images, very very slow!

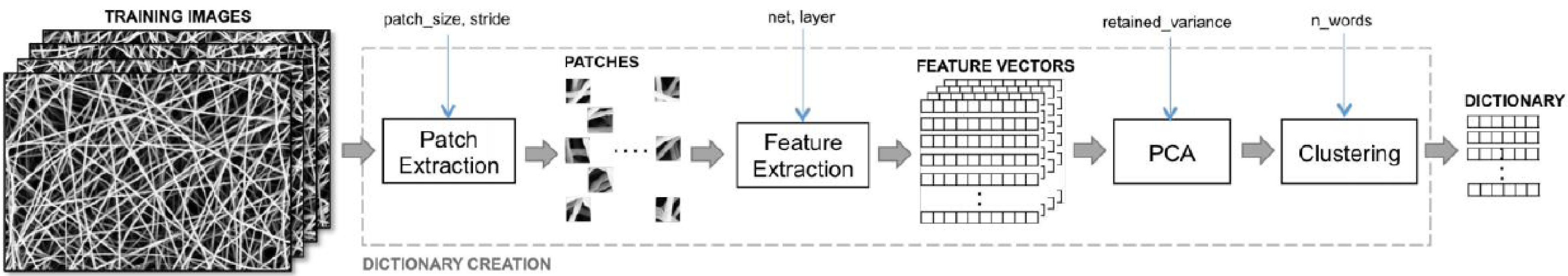
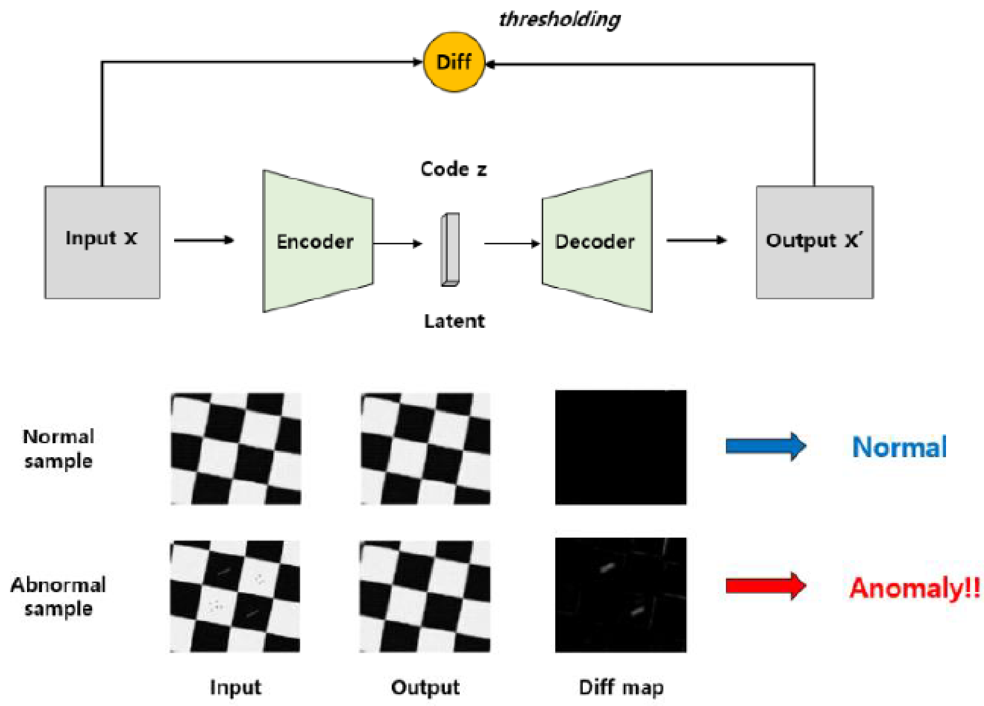


Figure 4. Examples of dictionary achieved considering different patch sizes and different number of subregions.

Unsupervised Anomaly Detection 주요 방법론 소개

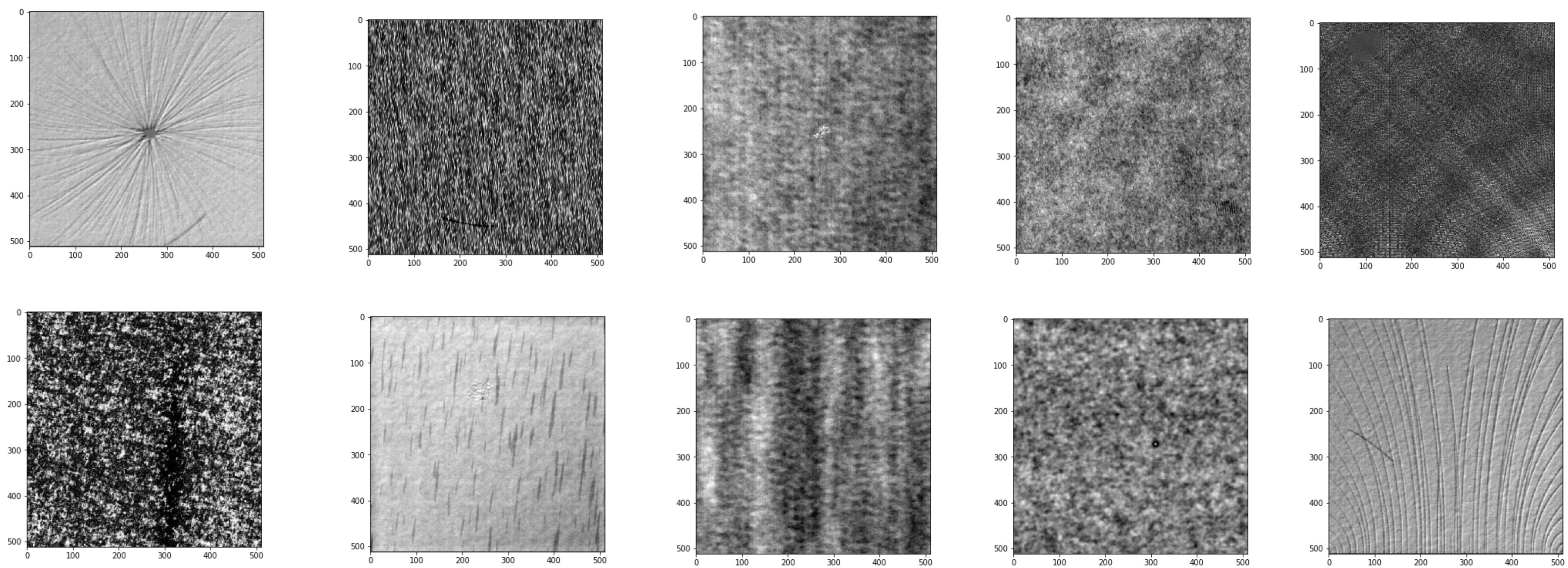
- Deep Convolutional Autoencoders and Variational Autoencoders (VAE) 기반 연구
  - Reconstruct normal training samples through a bottleneck (latent space)
  - At test phase, fail to reproduce images that differ from the data that was observed during training



Many papers provide further evidence that probabilities obtained from VAEs and other deep generative models might fail to model the true likelihood of the training data

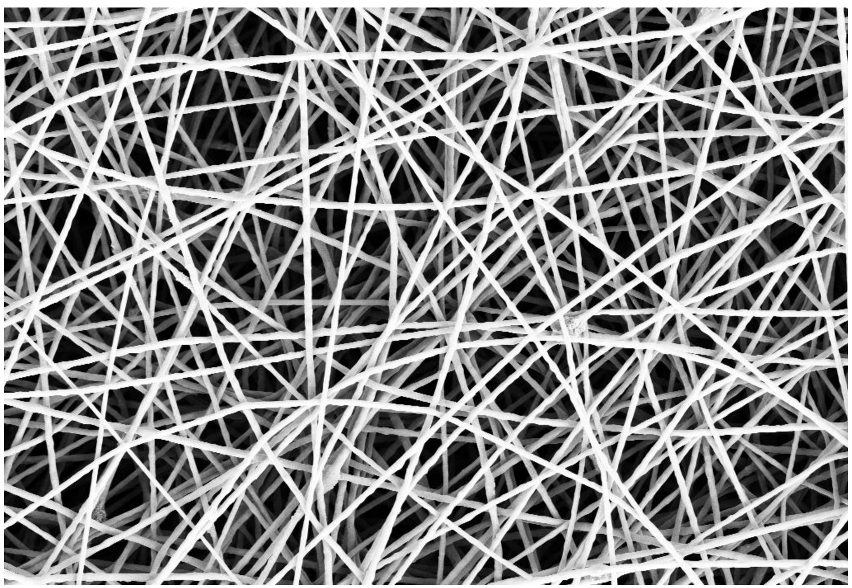
Unsupervised Anomaly Detection 주요 데이터 셋 소개

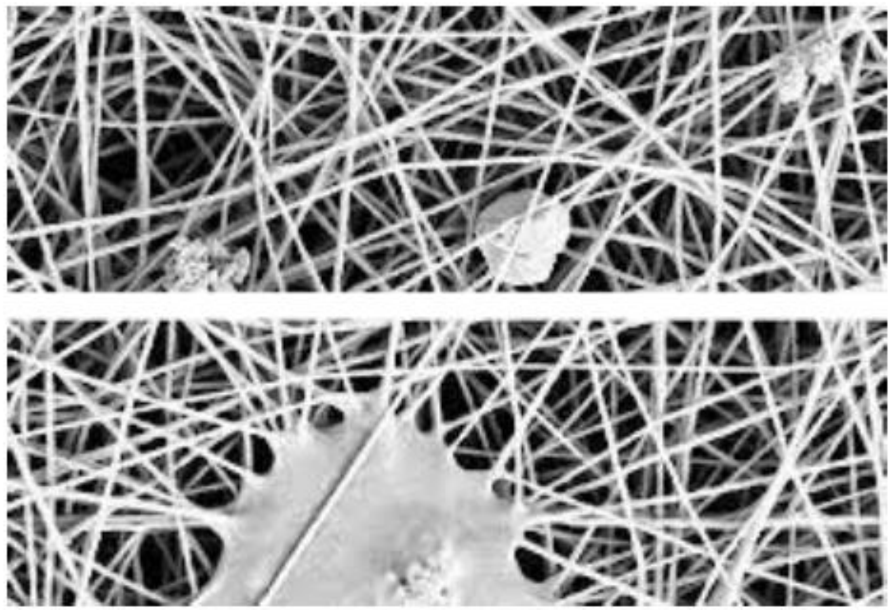
- DAGM 2007 데이터 셋
    - Industrial Optical Inspection을 위해 Defect(결함)을 가상으로 생성시켜 만든 데이터 셋



Unsupervised Anomaly Detection 주요 데이터 셋 소개

- NanoTwice 데이터 셋
    - Nano 섬유 이미지에서 Anomalous한 영역을 촬영한 이미지 데이터 셋
    - 학습용 정상 이미지는 단 5장만 존재하고 테스트용 이미지도 결함 이미지만 40장 존재하여 사용성 낮음





Unsupervised Anomaly Detection 주요 데이터 셋 소개

- MVTec-AD 데이터 셋

OK only

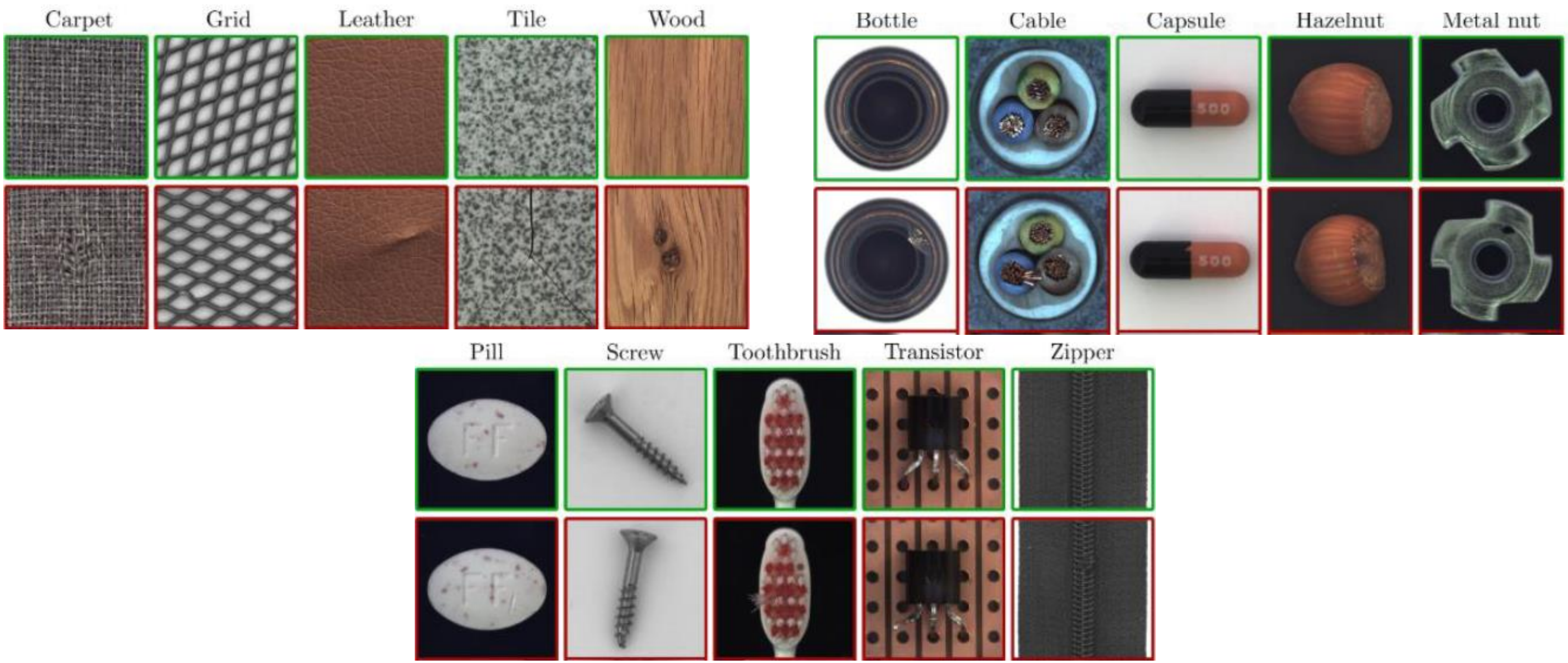
OK / NG

  - 앞서 두 데이터 셋의 다양성, 개수 부족을 해결하기 위해 제작된 데이터 셋
  - 15 categories with 3629 images for train/validation and 1725 images for test

	Category	# Train	# Test (good)	# Test (defective)	# Defect groups	# Defect regions	Image side length
Textures	Carpet	280	28	89	5	97	1024
	Grid	264	21	57	5	170	1024
	Leather	245	32	92	5	99	1024
	Tile	230	33	84	5	86	840
	Wood	247	19	60	5	168	1024
Objects	Bottle	209	20	63	3	68	900
	Cable	224	58	92	8	151	1024
	Capsule	219	23	109	5	114	1000
	Hazelnut	391	40	70	4	136	1024
	Metal Nut	220	22	93	4	132	700
	Pill	267	26	141	7	245	800
	Screw	320	41	119	5	135	1024
	Toothbrush	60	12	30	1	66	1024
	Transistor	213	60	40	4	44	1024
	Zipper	240	32	119	7	177	1024
	Total	3629	467	1258	73	1888	-

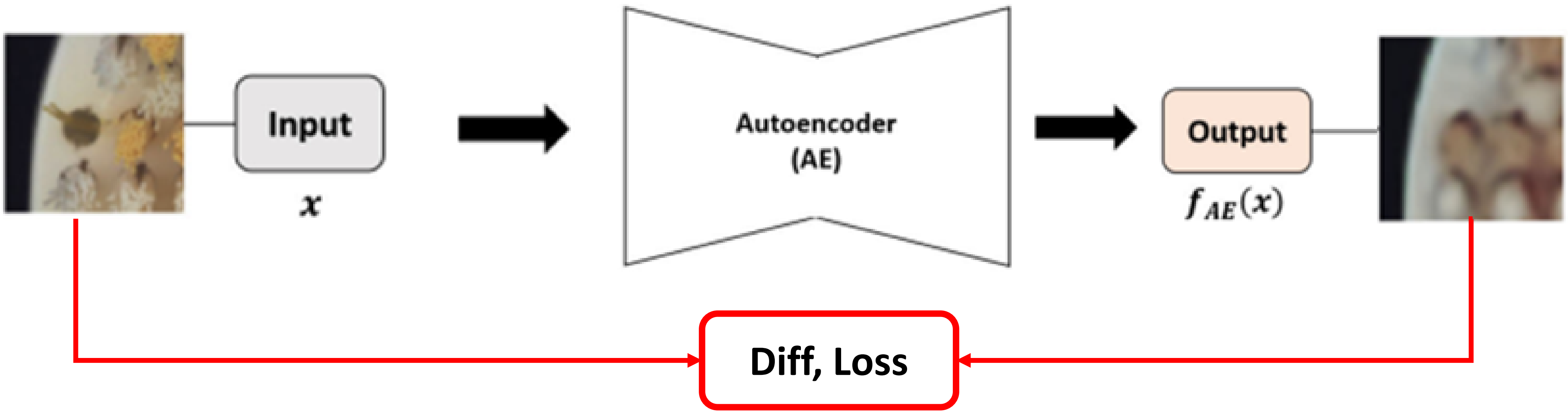
Unsupervised Anomaly Detection 주요 데이터 셋 소개

- MVTec-AD 데이터 셋
  - 실제 검사 현장에서 발생할 법한 결함 유형들을 직접 발생시켜 제작한 Real-World 데이터 셋



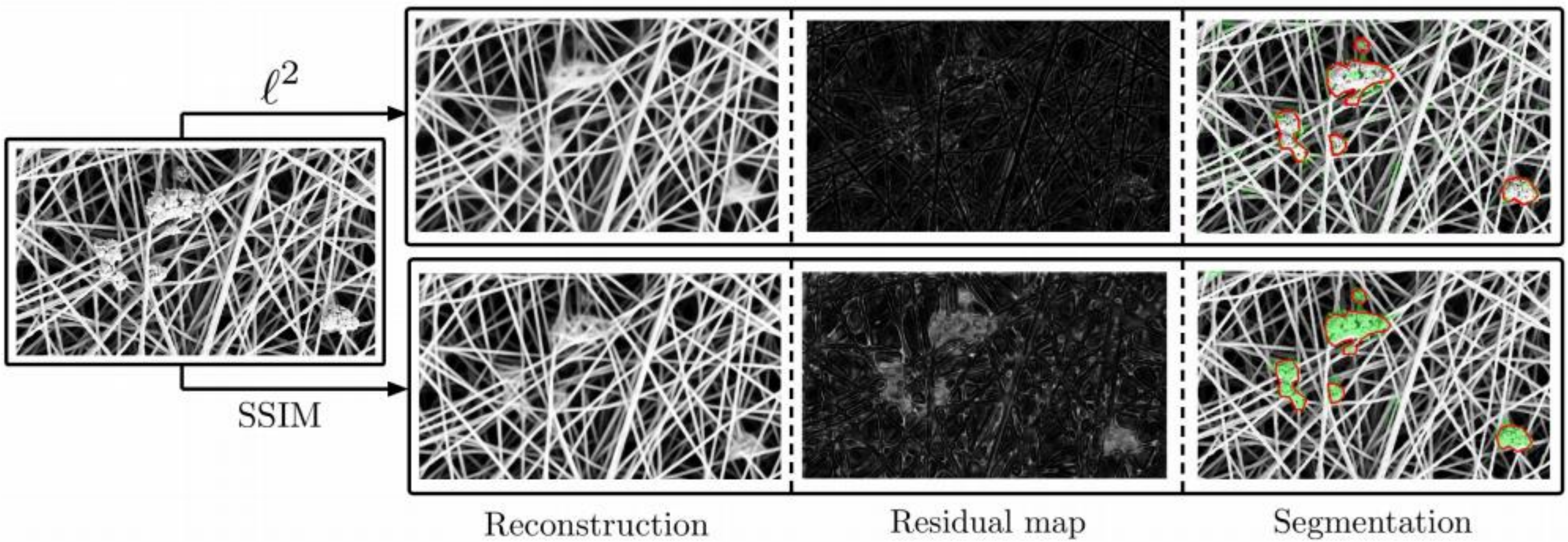
Unsupervised Anomaly Detection Using CAE

- Input image를 Bottlenecked Autoencoder에 넣어준 뒤 loss, difference 계산
    - Anomalous 한 영역은 제대로 복원하지 못해서 difference가 크게 측정되는 점을 이용
    - 다만 Bottleneck architecture와 L2 loss로 인해 Blurry 한 Output 발생



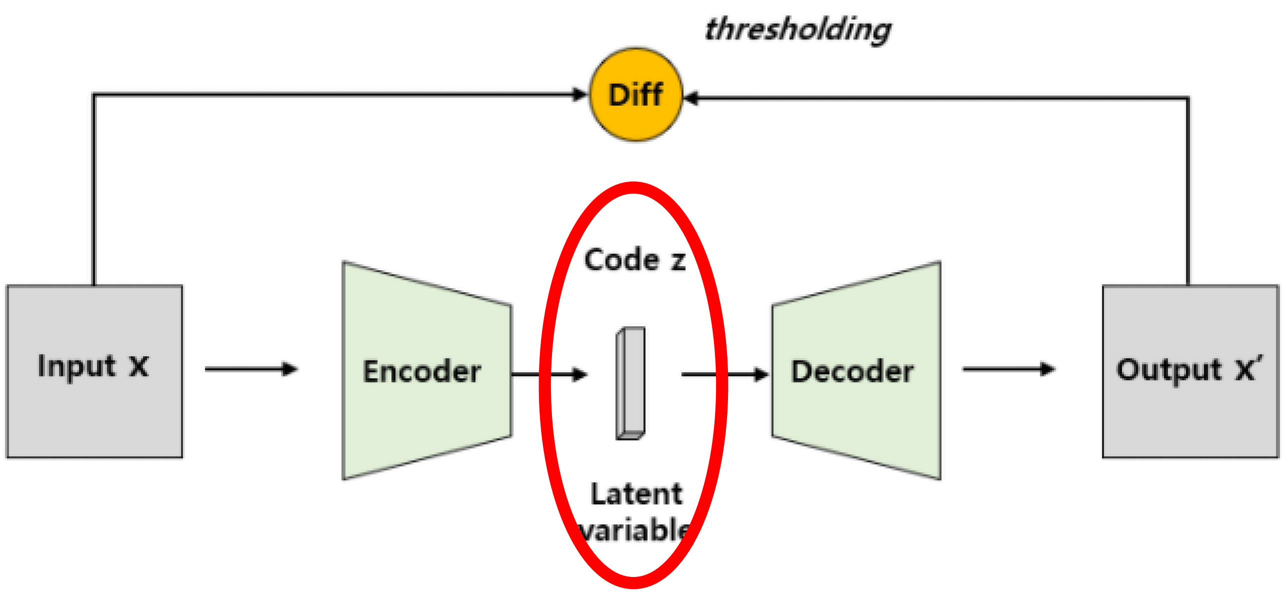
Unsupervised Anomaly Detection Using CAE

- “Improving Unsupervised Defect Segmentation by Applying Structural Similarity To Autoencoders”, 2018
    - L2 loss 대신 Structural Similarity (SSIM) loss를 사용하여 AE를 학습시키는 방법 제안



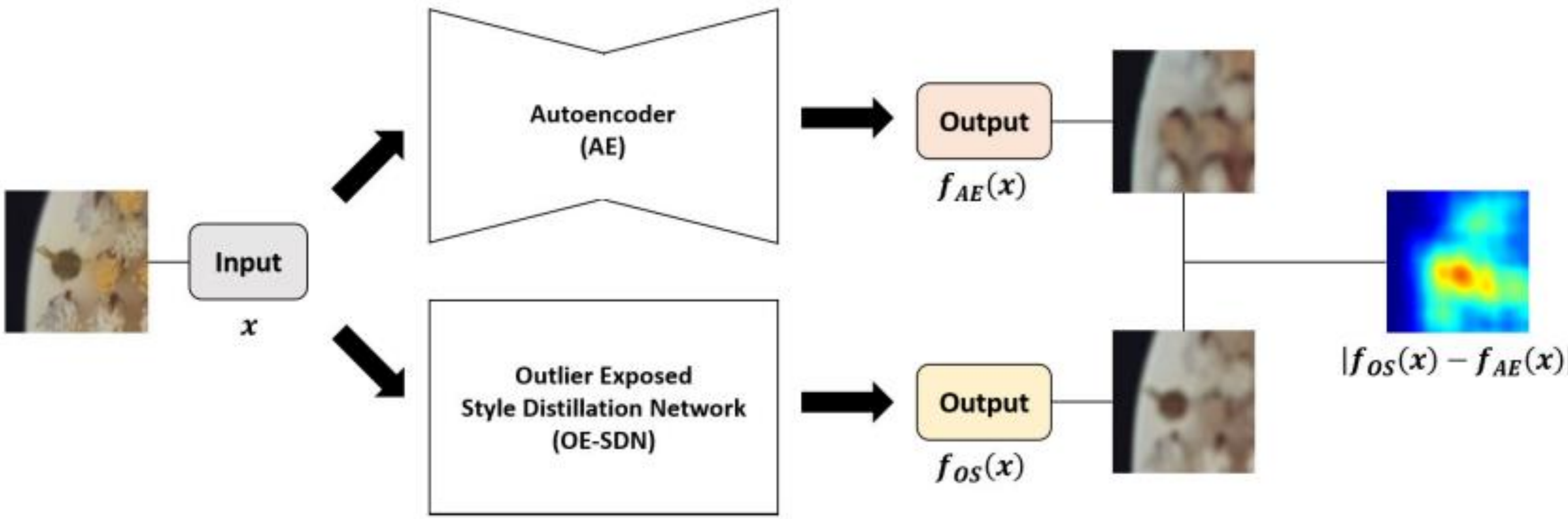
Unsupervised Anomaly Detection Using CAE

- 여전히 Bottleneck architecture로 인해 normal sample의 blurry한 output은 남아있음
- Bottleneck size를 조절하면 그에 따라 Anomaly Detection 성능이 변함
  - Bottleneck의 Code size를 키워주면 전반적인 복원 성능 증가, 결함 영역도 그대로 복원시키는 문제



Unsupervised Anomaly Detection Using Style Distillation

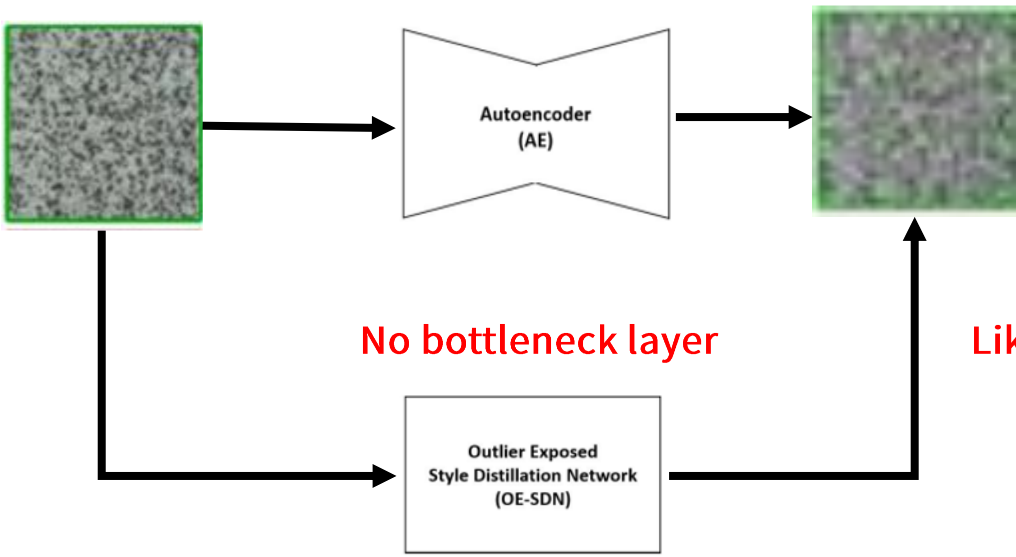
- "Unsupervised Anomaly Detection Using Style Distillation", 2020 Access
  - 이를 해결하기 위해 새로운 구조를 제안한 논문
  - Autoencoder가 Blurry 하게 Output을 내는 것을 그대로 따라하는 Style Distillation Network를 추가



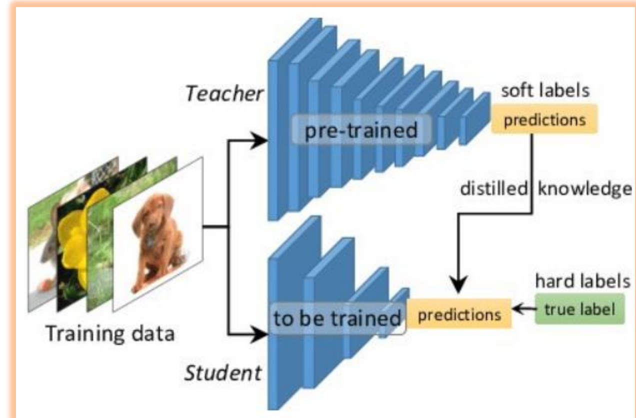
Unsupervised Anomaly Detection Using Style Distillation

- Style Distillation Network(SDN)는 AE의 Blurry한 Output을 내는 것을 따라하도록 학습

Train



Like Knowledge Distillation

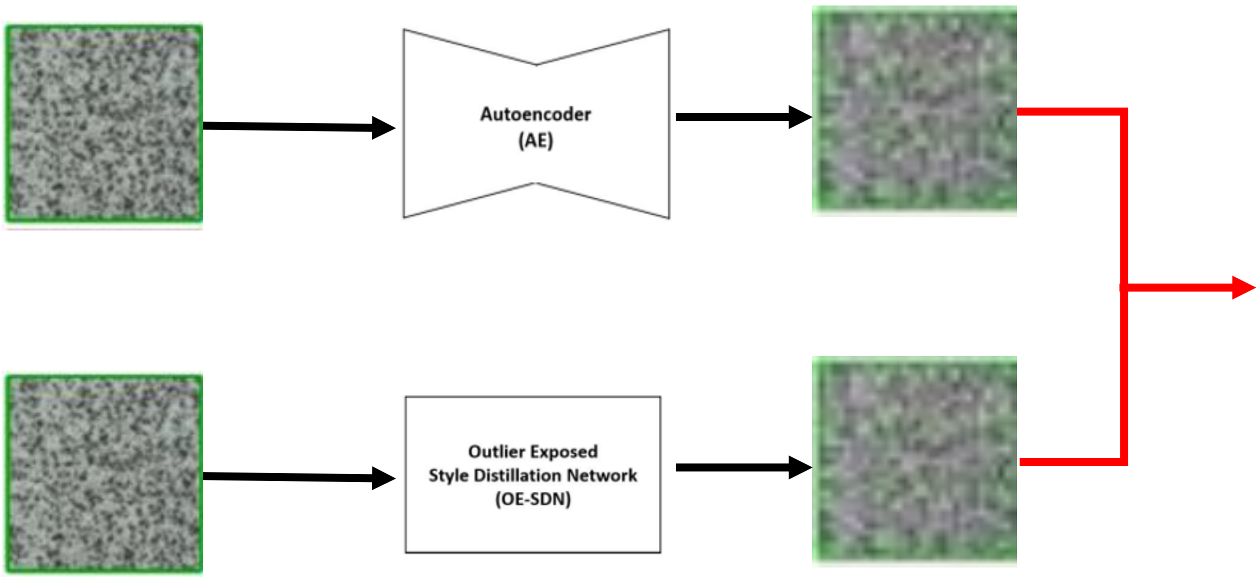


Reference: https://towardsdatascience.com/knowledge-distillation-simplified-dd4973dbc764

Unsupervised Anomaly Detection Using Style Distillation

- Style Distillation Network(SDN)는 AE의 Blurry한 Output을 내는 것을 따라하도록 학습
  - AE와 SDN의 output이 비슷해지도록 학습하는 loss function 고안

Train



$$\mathcal{L}_{KD}(f_{OS}) = \sum_{x \in \mathcal{X}_{\text{normal}}} d\left(f_{OS}(x), f_{AE}(x)\right).$$

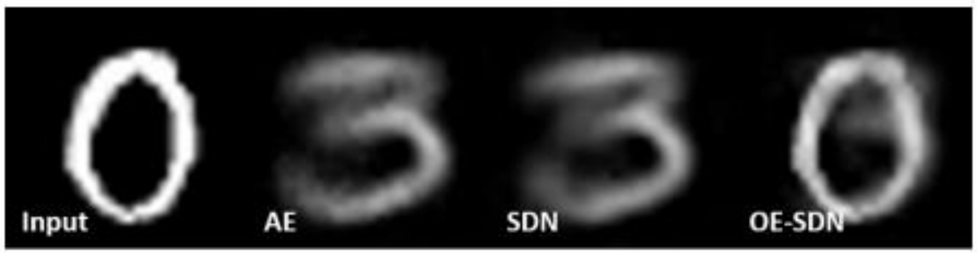
Unsupervised Anomaly Detection Using Style Distillation

- Style Distillation Network(SDN)는 AE의 Blurry한 Output을 내는 것을 따라하도록 학습

Train

  - SDN이 Blurry하게 Output을 생성해내는 것만 배워야하는데, 학습 데이터의 분포가 다양하지 않은 경우 그냥 뭉개진 3을 만드는 것을 imitate하는 부작용 발생
  - SDN이 이렇게 단순 암기하는 것을 막기 위한 Regularization 기법 제안

(a) Results of normal class “3”



(b) Results of anomalous class “0”

MNIST “3”을 OK Class로 학습시킨 경우

Unsupervised Anomaly Detection Using Style Distillation

- Outlier Exposed Style Distillation Network(OE-SDN)

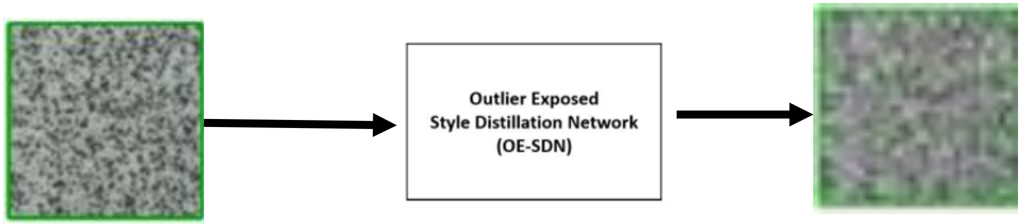
Train

- “Deep Anomaly Detection with Outlier Exposure”, 2019 ICLR 에서 아이디어를 가져옴
- Network의 better representation을 위해 Auxiliary Dataset을 학습에 사용하는 기법

TABLE 3. Results of the comparative experiment for the auxiliary dataset generation methods. Results are shown in terms of average AUROC of each setup on the MNIST and CIFAR-10 datasets.

Method	MNIST	CIFAR-10
Adversarial Noise [29]	0.960	0.631
Gaussian Blurring	0.954	0.645
Gaussian Noise	0.956	0.638
Horizontal Flip	0.968	0.638
Reconstruction of AE	0.959	0.623
Rotation	**0.975**	**0.707**
Shearing	0.958	0.619
SVD Blurring	0.952	0.640
Vertical Flip	0.968	0.694
No Auxiliary Dataset (SDN)	0.958	0.616

Make Auxiliary Dataset using Rotation



$$\mathcal{L}_{OER}(f_{OS}) = \sum_{\tilde{x} \in \mathcal{X}_{\text{aux}}} d\left(\tilde{x}, f_{OS}(\tilde{x})\right).$$

Unsupervised Anomaly Detection Using Style Distillation

- Outlier Exposed Style Distillation Network(OE-SDN)

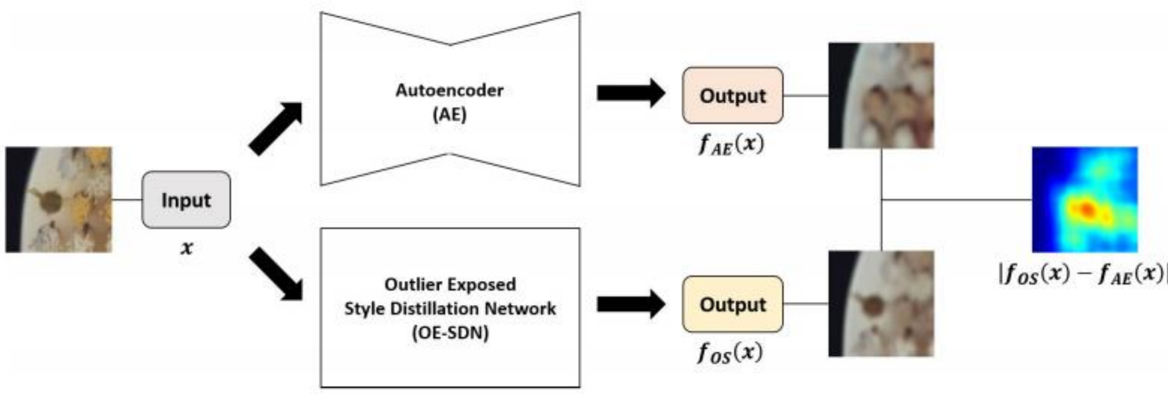
Train

- Autoencoder는 미리 학습을 시키고, 그 뒤 AE와 OE-SDN을 multi-task loss로 학습
- Total loss term은 λ에 의해 조절, loss function은 DSSIM loss + Mean Squared Displacement loss

$$\mathcal{J}_{OS}(f_{OS}) = (1-\lambda)\cdot\mathcal{L}_{KD}(f_{OS}) + \lambda\cdot\mathcal{L}_{OER}(f_{OS}),$$

$$\mathcal{L}_{KD}(f_{OS}) = \sum_{x\in\mathcal{X}_{\text{normal}}} d\left(f_{OS}(x), f_{AE}(x)\right).$$

$$\mathcal{L}_{OER}(f_{OS}) = \sum_{\tilde{x}\in\mathcal{X}_{\text{aux}}} d\left(\tilde{x}, f_{OS}(\tilde{x})\right).$$

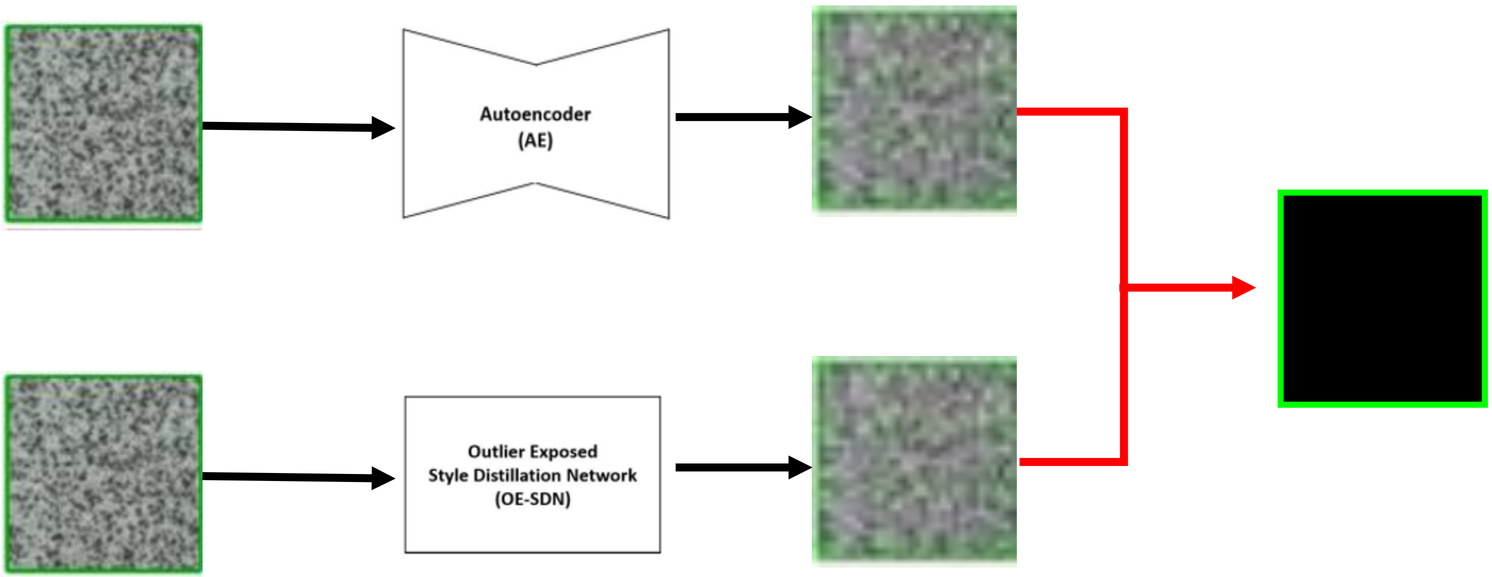


Unsupervised Anomaly Detection Using Style Distillation

- OE-SDN은 AE의 Blurry한 Output을 내는 것을 따라하도록 학습

Test

OK Image로 Test 시

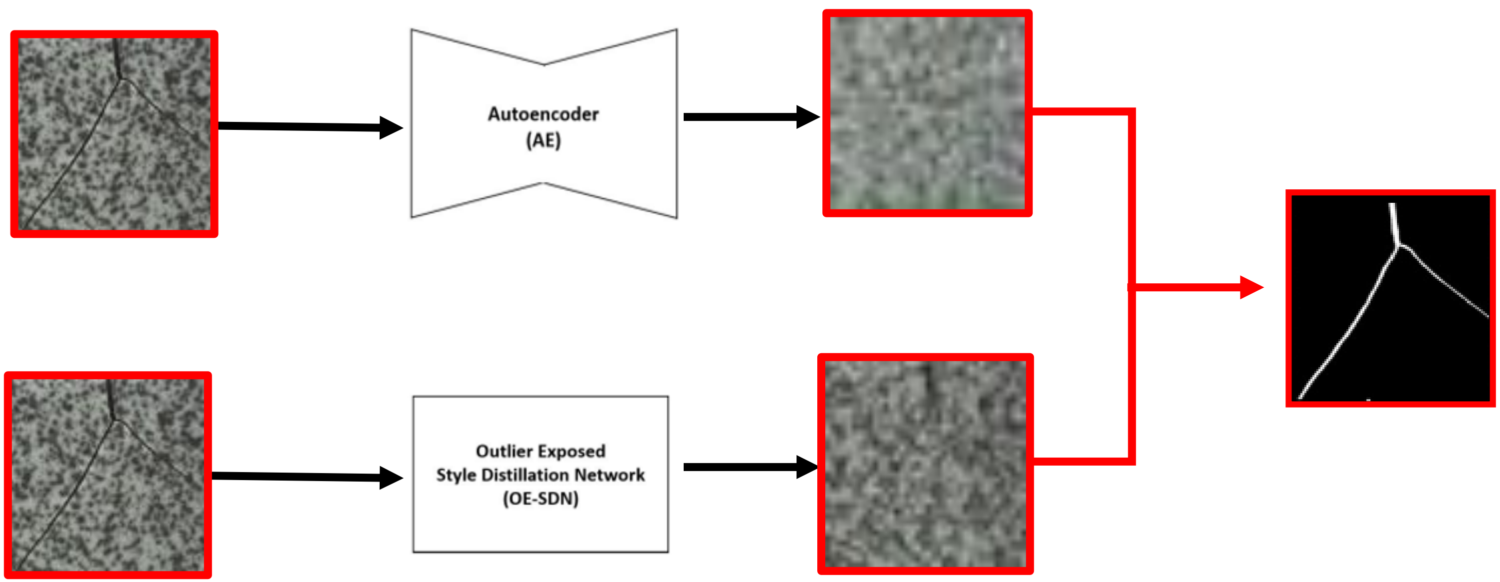


Reference: https://towardsdatascience.com/knowledge-distillation-simplified-dd4973dbc764

Unsupervised Anomaly Detection Using Style Distillation

- OE-SDN은 AE의 Blurry한 Output을 내는 것을 따라하도록 학습

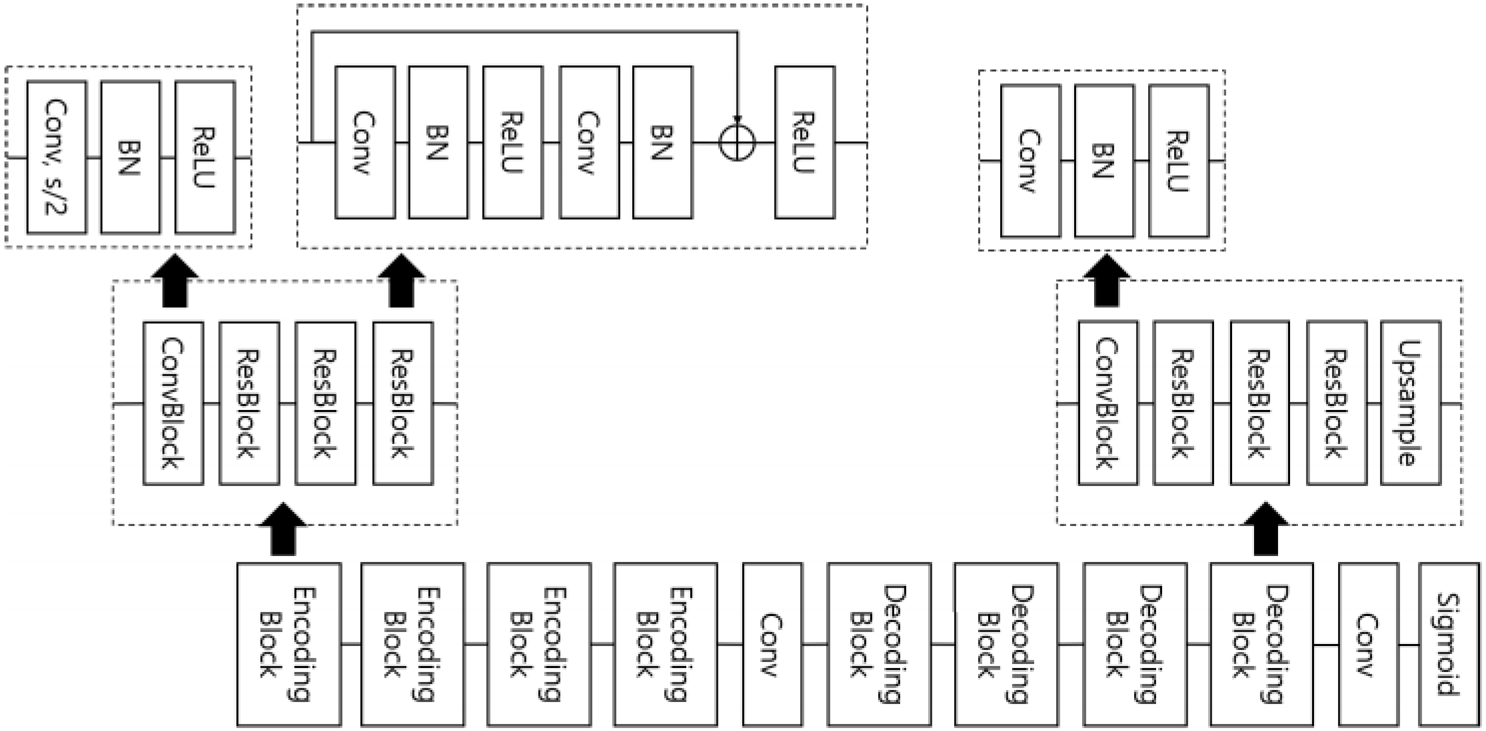
Test NG Image로 Test 시



Reference: https://towardsdatascience.com/knowledge-distillation-simplified-dd4973dbc764

Unsupervised Anomaly Detection Using Style Distillation

- AE & OE-SDN architecture



Autoencoder architecture

Unsupervised Anomaly Detection Using Style Distillation

- AE & OE-SDN architecture
  - RNAN - “Residual non-local attention networks for image restoration”, 2019 ICLR

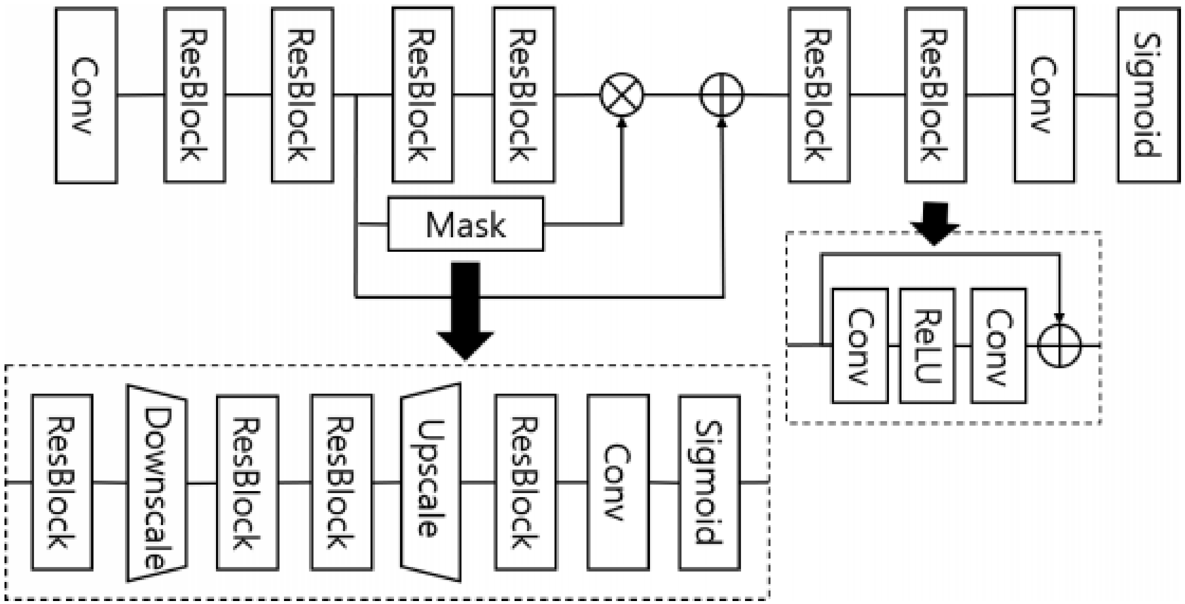


FIGURE 4. Architecture diagram for the OE-SDN. The architecture adapts the RNAN [25], in which we set the number of filters to 32 in all convolution layers.

OE-SDN architecture

Unsupervised Anomaly Detection Using Style Distillation

- Experimental results (Classification)

TABLE 1. AUROC of each method for unsupervised anomaly classification on MNIST and CIFAR-10 datasets. The highest AUROC for each setup is highlighted in bold.

Dataset	Normal Class	OC-SVM [26]	IF [27]	AnoGAN [8]	DeepSVDD [24]	AE	SDN	OE-SDN
MNIST	0	0.986	0.980	0.966	0.980	**0.991**	0.989	0.984
	1	0.995	0.973	0.992	0.997	**0.998**	**0.998**	**0.998**
	2	0.825	0.886	0.850	0.917	0.965	0.952	**0.983**
	3	0.881	0.899	0.887	0.919	0.950	0.933	**0.974**
	4	0.949	0.927	0.894	0.949	0.923	0.933	**0.977**
	5	0.771	0.855	0.883	0.885	0.958	0.953	**0.973**
	6	0.965	0.956	0.947	0.983	0.992	0.991	**0.995**
	7	0.937	0.920	0.935	0.946	0.946	0.948	**0.971**
	8	0.889	0.899	0.849	**0.939**	0.912	0.920	0.921
	9	0.931	0.935	0.924	0.965	0.970	0.968	**0.972**
	Average	0.913	0.923	0.913	0.948	0.961	0.958	**0.975**
CIFAR-10	Airplane	0.616	0.601	0.671	0.617	**0.790**	0.677	0.774
	Automobile	0.638	0.508	0.547	0.659	0.661	0.671	**0.821**
	Bird	0.500	0.492	0.529	0.508	**0.647**	0.598	0.638
	Cat	0.559	0.551	0.545	**0.591**	0.515	0.520	0.569
	Deer	**0.660**	0.498	0.651	0.609	0.587	0.582	0.594
	Dog	0.624	0.585	0.603	0.657	0.571	0.535	**0.659**
	Frog	**0.747**	0.429	0.585	0.677	0.596	0.648	0.629
	Horse	0.626	0.551	0.625	0.673	0.630	0.611	**0.755**
	Ship	0.749	0.742	0.758	0.759	0.783	0.707	**0.844**
	Truck	0.759	0.589	0.665	0.731	0.623	0.608	**0.784**
	Average	0.648	0.555	0.618	0.648	0.640	0.616	**0.707**

Unsupervised Anomaly Detection Using Style Distillation

- Experimental results (Segmentation)

TABLE 2. Pixel-wise AUROC of each method for unsupervised anomaly segmentation on the MVTec-AD dataset. The highest AUROC for each category is highlighted in bold.

Category		AnoGAN [8]	CNN Feature Similarity [28]	AE [4]	AE (ours)	SDN	OE-SDN
Textures	Carpet	0.54	0.72	0.87	0.95	**0.96**	**0.96**
	Grid	0.58	0.59	0.94	**0.97**	**0.97**	**0.97**
	Leather	0.64	**0.87**	0.78	0.85	0.83	0.85
	Tile	0.50	**0.93**	0.59	0.81	0.88	0.85
	Wood	0.62	**0.91**	0.73	0.79	0.82	0.82
	Average	0.58	0.80	0.78	0.87	**0.89**	**0.89**
Objects	Bottle	0.86	0.78	0.93	0.94	**0.96**	0.95
	Cable	0.78	0.79	0.82	**0.84**	**0.84**	**0.84**
	Capsule	0.84	0.84	0.94	0.92	**0.97**	**0.97**
	Hazelnut	0.87	0.72	0.97	0.97	**0.98**	**0.98**
	Metal nut	0.76	0.82	0.89	0.89	**0.95**	0.93
	Pill	0.87	0.68	0.91	**0.94**	0.92	0.93
	Screw	0.80	0.87	0.96	0.97	**0.98**	0.97
	Toothbrush	0.90	0.77	0.92	**0.98**	**0.98**	**0.98**
	Transistor	0.80	0.66	**0.90**	0.88	0.89	0.89
	Zipper	0.78	0.76	0.88	0.87	0.89	**0.91**
	Average	0.83	0.77	0.91	0.92	**0.93**	**0.93**

Summary

- Unsupervised Anomaly Detection의 대표적인 방법론, 데이터 셋 소개
- Convolutional Autoencoder를 이용한 Unsupervised Anomaly Detection 소개 및 문제 정의
- 문제를 해결하기 위한 Unsupervised Anomaly Detection Using Style Distillation 논문 리뷰
    - Blurry 한 Output을 Style Distillation을 통해 해결하자!
    - 더 나은 Representation을 배우기 위해 Outlier Exposure Regularization을 이용하자!