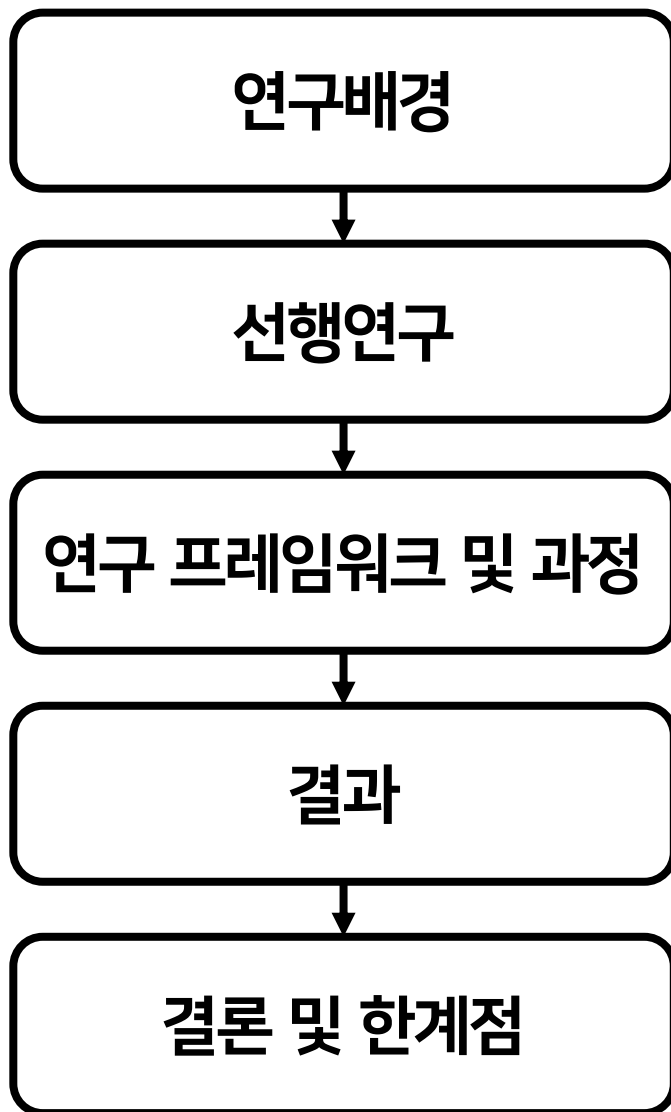

Vision Transformer와 유사도 기반 반도체 혼합결함유형 예측

20191182 산업공학과 최재준

지도교수: 조성주 교수

목차

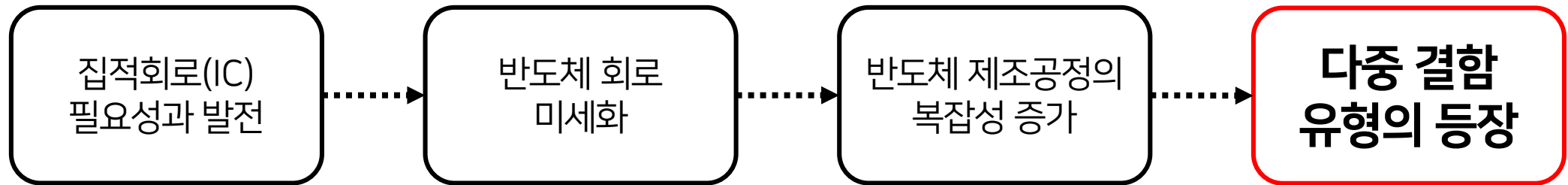


연구배경 - 새로운 결함의 등장

I 웨이퍼 빈 맵에서 혼합형 결함 패턴 탐지 및 클러스터링

- 산업시스템공학과 김희영

이러한 무작위가 아닌 결함 패턴은 할당 가능한 원인으로 인해 발생합니다. 따라서 장애의 근본 원인을 파악하기 위해서는 체계적인 결함 패턴을 감지하고 클러스터링하는 것이 중요합니다. 웨이퍼 제조 공정이 복잡해지면서 혼합형 결함 패턴(하나의 웨이퍼에서 두 가지 이상의 서로 다른 유형의 결함 패턴이 동시에 발생하는 현상)이 과거보다 빈번하게 발생하고 있습니다. 혼합형 결함 패턴을 검출하고 클러스터링하는 데에는 1) 클러스터 수 결정, 2) 복잡한 모양의 결함 패턴 클러스터링, 3) 체계적인 결함 패턴에서 무작위 결함을 분리하는 등 몇 가지 과제가 있습니다. 본 연구에서는 WBM의 혼합형 결함 패턴을 탐지하고 클러스터링하기 위한 새로운 프레임워크를 제안합니다. 체계적인 결함이 클러스터링되기 전에 탐지하기 위해 연결 경로 필터링이라는 새로운 필터링 방법을 제안합니다. 이 방법은 미리 지정된 임계값보다 긴 경로를 탐지하는 것을 목표로 합니다. 검출된 체계적 결함은 이후 무한 뒤틀림 혼합 모델을 사용하여 클러스터링됩니다. 이 모델은 복잡한 형태의 혼합형 결함 패턴을 효과적으로 군집화할 수 있으며, 또한 군집 개수를 미리 지정할 필요 없이 군집화 과정에서 자동으로 결정됩니다. 우리는 반도체 회사의 실제 데이터를 사용하여 제안된 방법을 검증했습니다. 실험 결과는 기본 클러스터 수를 추정하고 혼합형 결함 패턴을 클러스터링하는 데 제안된 방법의 효율성을 입증했습니다.

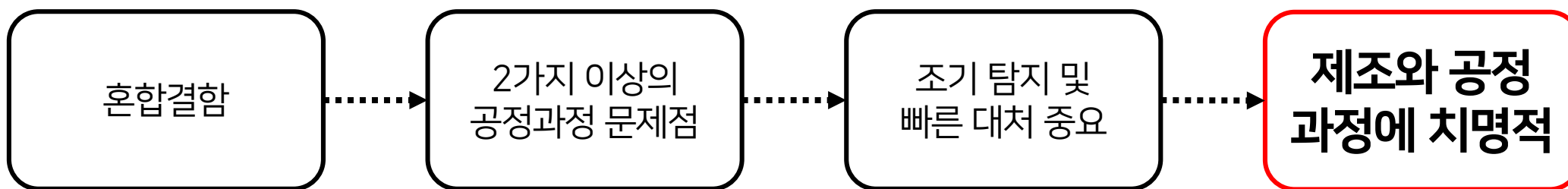


새로운 다중 결함이 등장하기에

기존에 있는 결함 데이터를 통해 새로운 다중 결함임을 인지하는 것이 필요함.

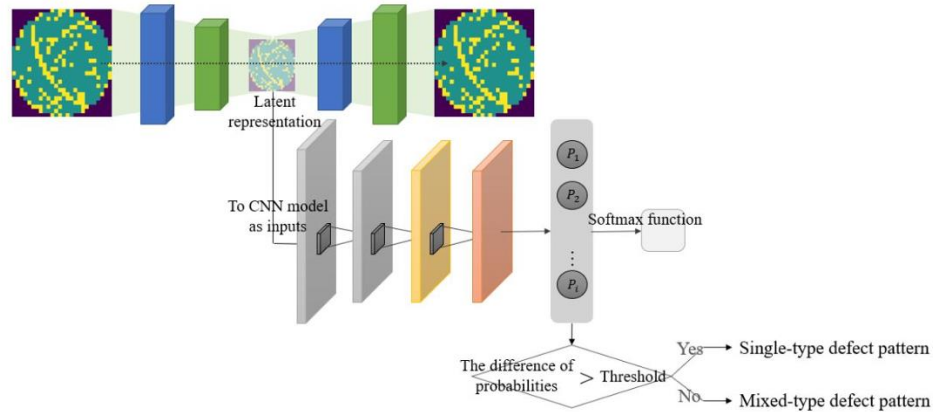
연구배경 - 혼합결함 탐지의 중요성

결함 패턴	해당 패턴의 원인
Normal	일반 패턴은 웨이퍼에 결함이 없는 것이 특징입니다.
Center	플라즈마 영역이나 박막증착, 산화 등의 문제로 인해 발생한다.
Dount	포토리스트 고형물이 재증착 되어 발생합니다.
Edge_loc	확산 공정 중 가열이 고르지 않거나 식각 중 문제로 인해 발생합니다.
Edge_ring	급속 열 어닐링 공정에서 에지의 비정상적인 온도 제어나 리소그래피 중 문제. 또는 저장 노드 공정에서 층간 정렬 불량 으로 인해 발생합니다.
Loc(Localized)	주어진 기계에서 과도한 진동 으로 인해 발생하여 충분한 입자 또는 결정질 이질성이 발생합니다.
Near_Full	웨이퍼의 다이 중 90% 이상이 불량 인 경우를 말합니다.
Scratch	배송 및 취급 과정 에서 사람의 실수나 화학적 기계적 연마 과정 에서 패드가 경화되어 발생합니다.
Random	제조 환경 요인 에 의해 발생하며 체계화되지 않습니다.



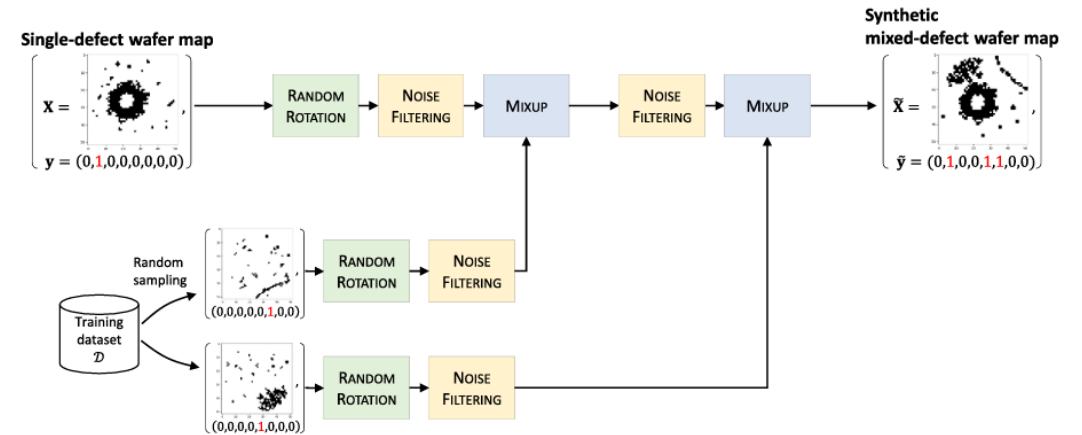
혼합결함에 대해서 기존 단일결함처럼 1가지 결함유형으로 탐지한다면,
공정의 문제점을 정확히 판단하기 어렵고 이는 제조와 공정에 치명적인 영향을 준다.
기존에 있는 결함 데이터를 통해 새로운 혼합결함을 인지하는 기술이 필요함.

선행연구 - 한계점



Byun, Y., & Baek, J. G. (2020, January). Mixed Pattern Recognition Methodology on Wafer Maps with Pre-trained Convolutional Neural Networks. In ICAART (2) (pp. 974-979).

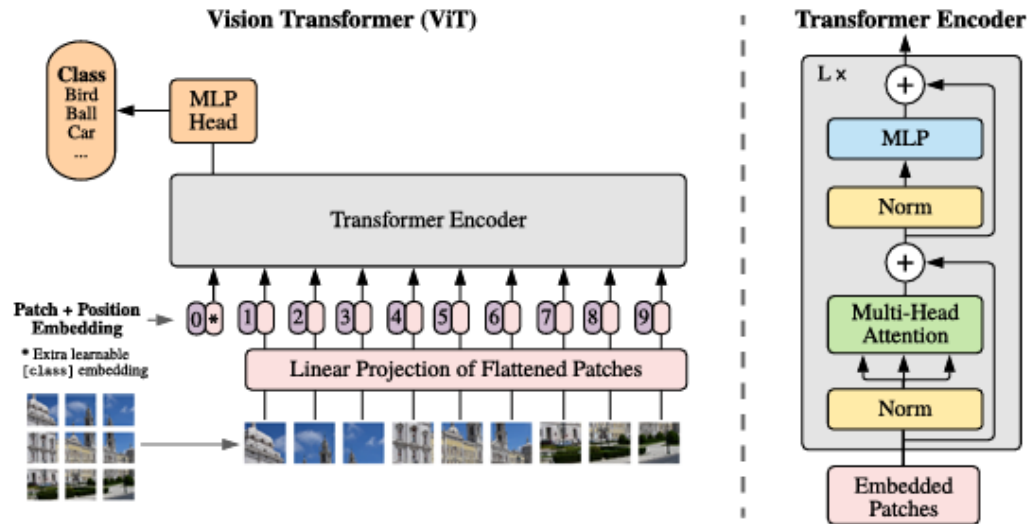
CNN + Convolutional Autoencoder + 임계값
임계값 설정에 어려움이 존재



Shim, J., & Kang, S. (2023). Learning from single-defect wafer maps to classify mixed-defect wafer maps. *Expert Systems with Applications*, 233, 120923.

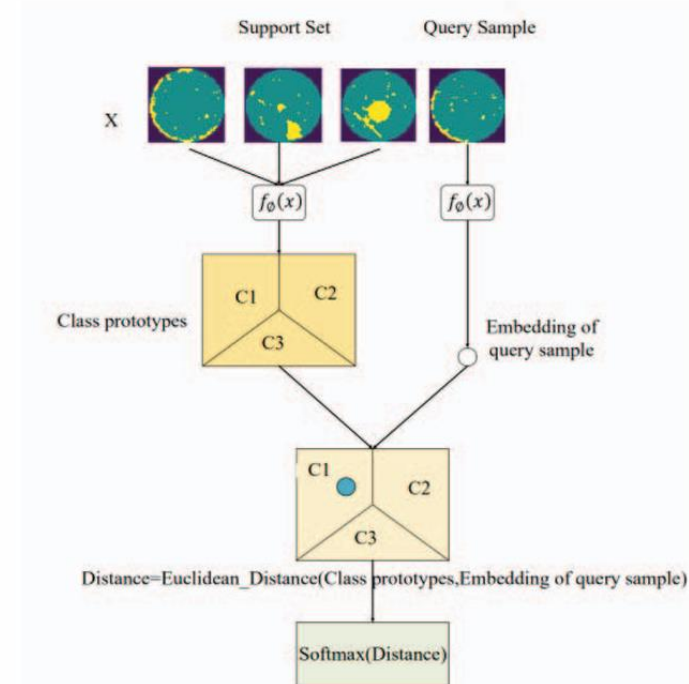
VGG-16 + Synthesis Method + Sigmoid
단일 결함을 혼합한 데이터도 모델 학습에 사용

선행연구 - 벤치마킹



Dosovitskiy, A., Beyer, L., Kolesnikov, A., Weissenborn, D., Zhai, X., Unterthiner, T., ... & Houlsby, N. (2020). An image is worth 16x16 words: Transformers for image recognition at scale. *arXiv preprint arXiv:2010.11929*.

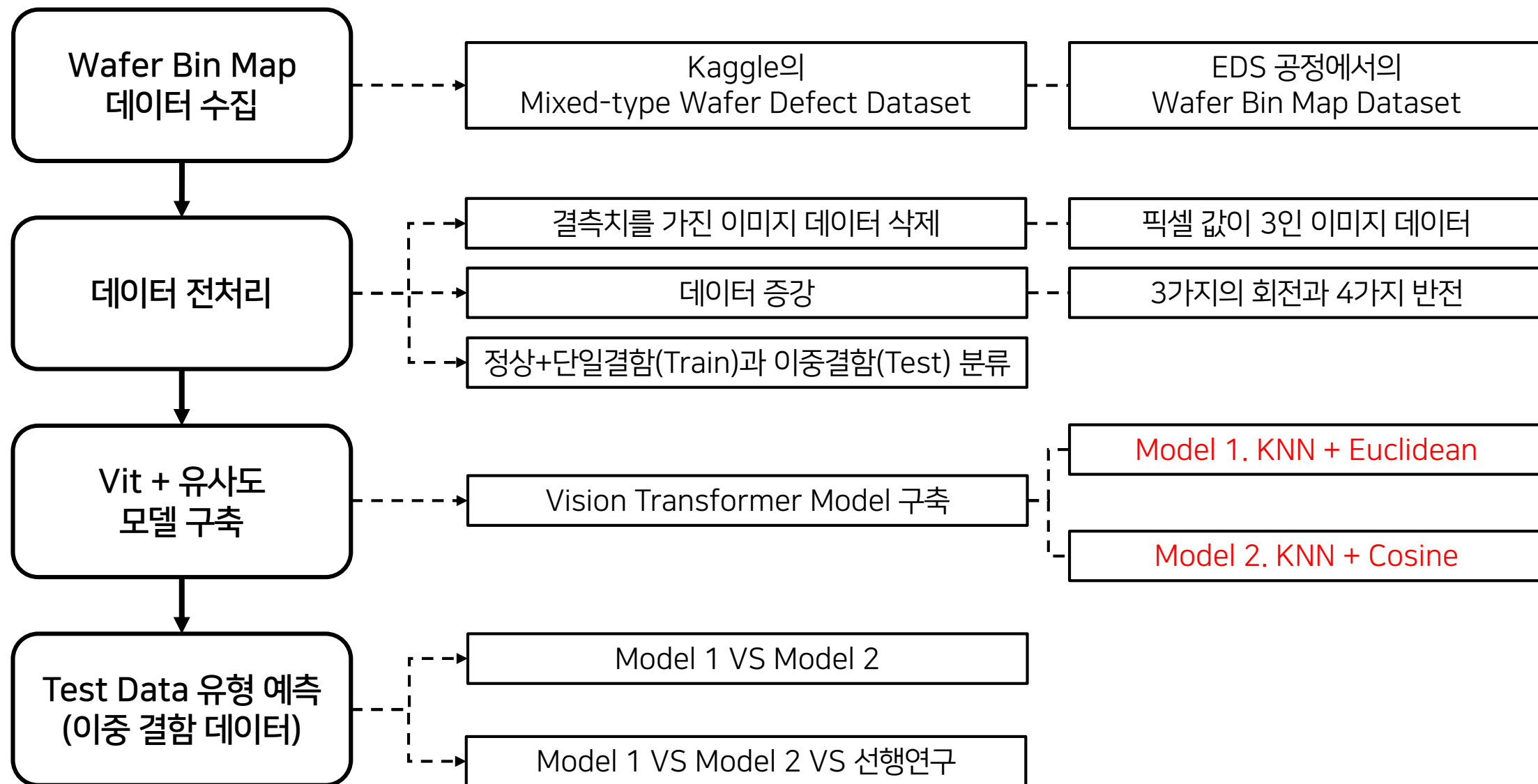
이미지 분류기법 중 Vision Transformer 사용
Vision Transformer를 활용한 Label 예측



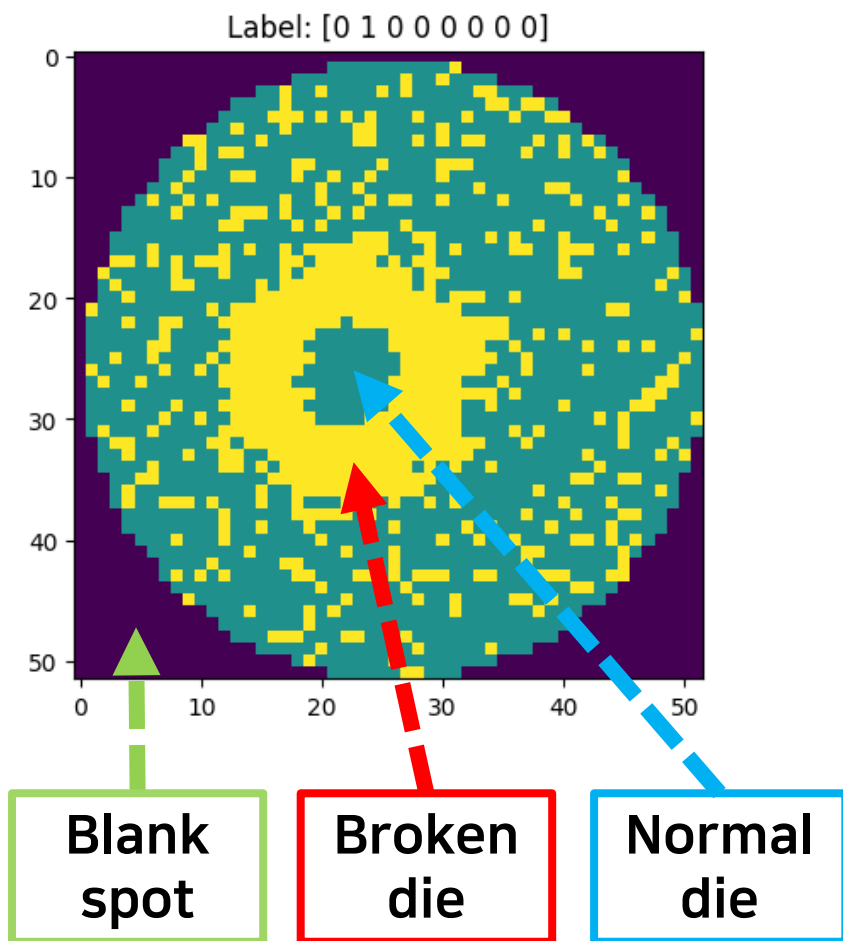
Yu, N., Chen, H., Xu, Q., & Hasan, M. M. (2021, October). Wafer map defect recognition with few shot learning based on hybrid self-attention mechanism and prototype network. In *2021 China Automation Congress (CAC)* (pp. 4128-4134). IEEE.

Few-shot Learning + Embedding + Euclidean
Embedding과 Euclidean Distance 기반
혼합유형 예측

연구 프레임워크 및 과정 - 프레임워크 소개



연구 프레임워크 및 과정 - Wafer Bin Map 데이터 수집



이미지의 사이즈 = 52*52

정상(1가지) + 단일결함(8가지)

Label	유형	개수
Normal (정상)	[0 0 0 0 0 0 0]	1000
Center	[1 0 0 0 0 0 0]	1000
Donut	[0 1 0 0 0 0 0]	1000
Edge_Loc	[0 0 1 0 0 0 0]	1000
Edge_Ring	[0 0 0 1 0 0 0]	1000
Loc	[0 0 0 0 1 0 0]	1000
Near_Full	[0 0 0 0 0 1 0]	149
Scratch	[0 0 0 0 0 0 1]	1000
Random	[0 0 0 0 0 0 1]	866

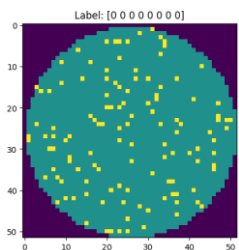
이중결함(13가지)

Label	유형	개수
Center+Edge_Loc	[1 0 1 0 0 0 0]	1000
Center+Edge_Ring	[1 0 0 1 0 0 0]	1000
Center+Loc	[1 0 0 0 1 0 0]	1000
Center+Scratch	[1 0 0 0 0 0 1]	1000
Donut+Edge_Loc	[0 1 1 0 0 0 0]	1000
Donut+Edge_Ring	[0 1 0 1 0 0 0]	1000
Donut+Loc	[0 1 0 0 1 0 0]	1000
Donut+Scratch	[0 1 0 0 0 0 1]	1000
Edge_Loc+Loc	[0 0 1 0 1 0 0]	1000
Edge_Loc+Scratch	[0 0 1 0 0 0 1]	1000
Edge_Ring+Loc	[0 0 0 1 1 0 0]	1000
Edge_Ring+Scratch	[0 0 0 1 0 0 1]	1000
Loc+Scratch	[0 0 0 0 1 0 1]	1000

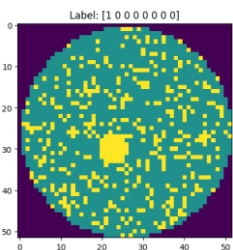
연구 프레임워크 및 과정 - Wafer Bin Map 데이터 수집

정상 + 단일 결함

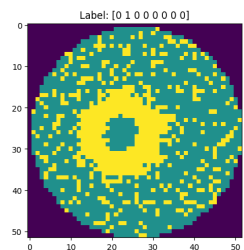
Normal



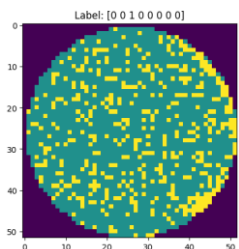
Center (C)



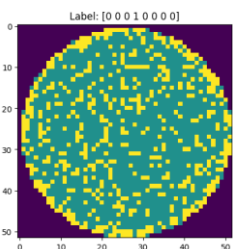
Dount (D)



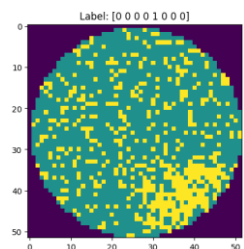
Edge_Loc (EL)



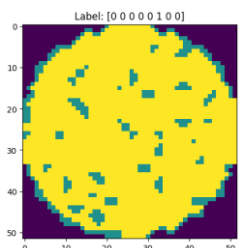
Edge_Ring (ER)



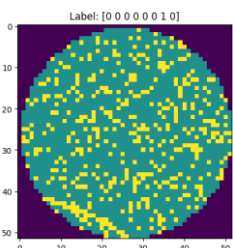
Loc (L)



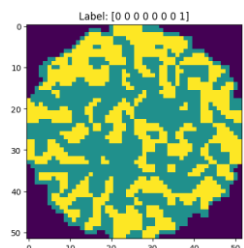
Near_Full(NF)



Scratch (S)

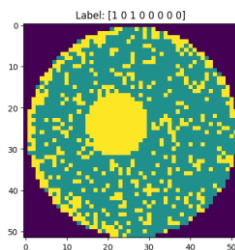


Random (R)

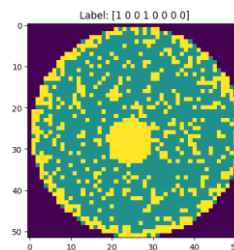


이중 결함

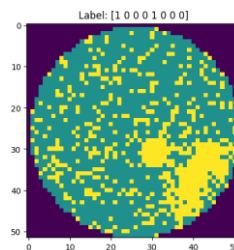
C+EL



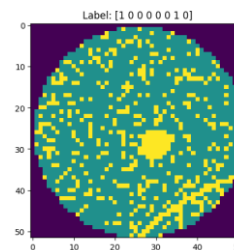
C+ER



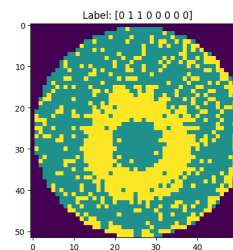
C+L



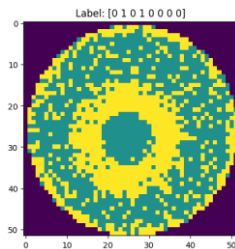
C+S



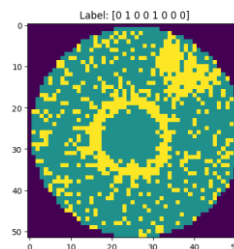
D+EL



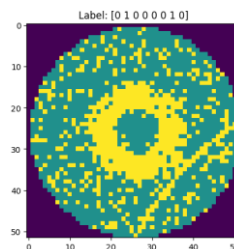
D+ER



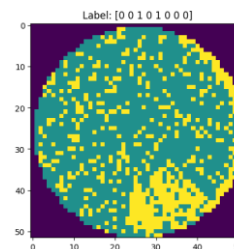
D+L



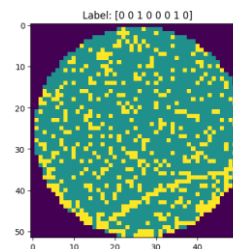
D+S



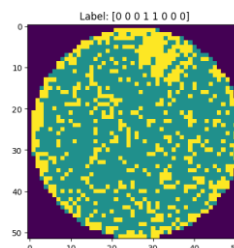
EL+L



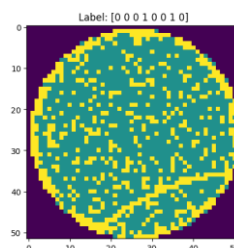
EL+S



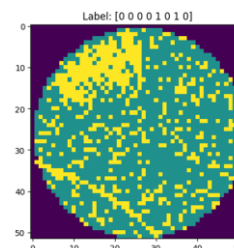
ER+L



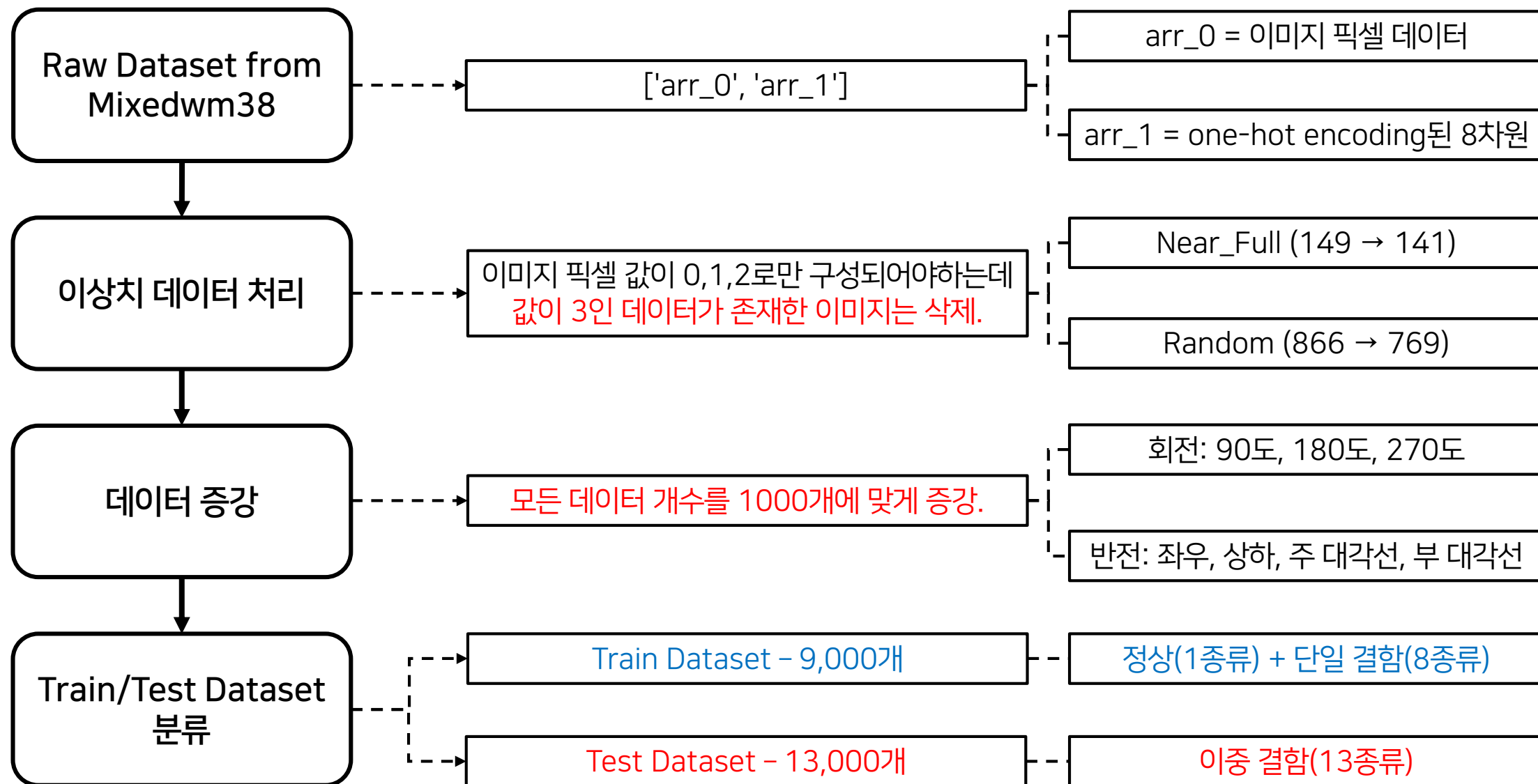
ER+S



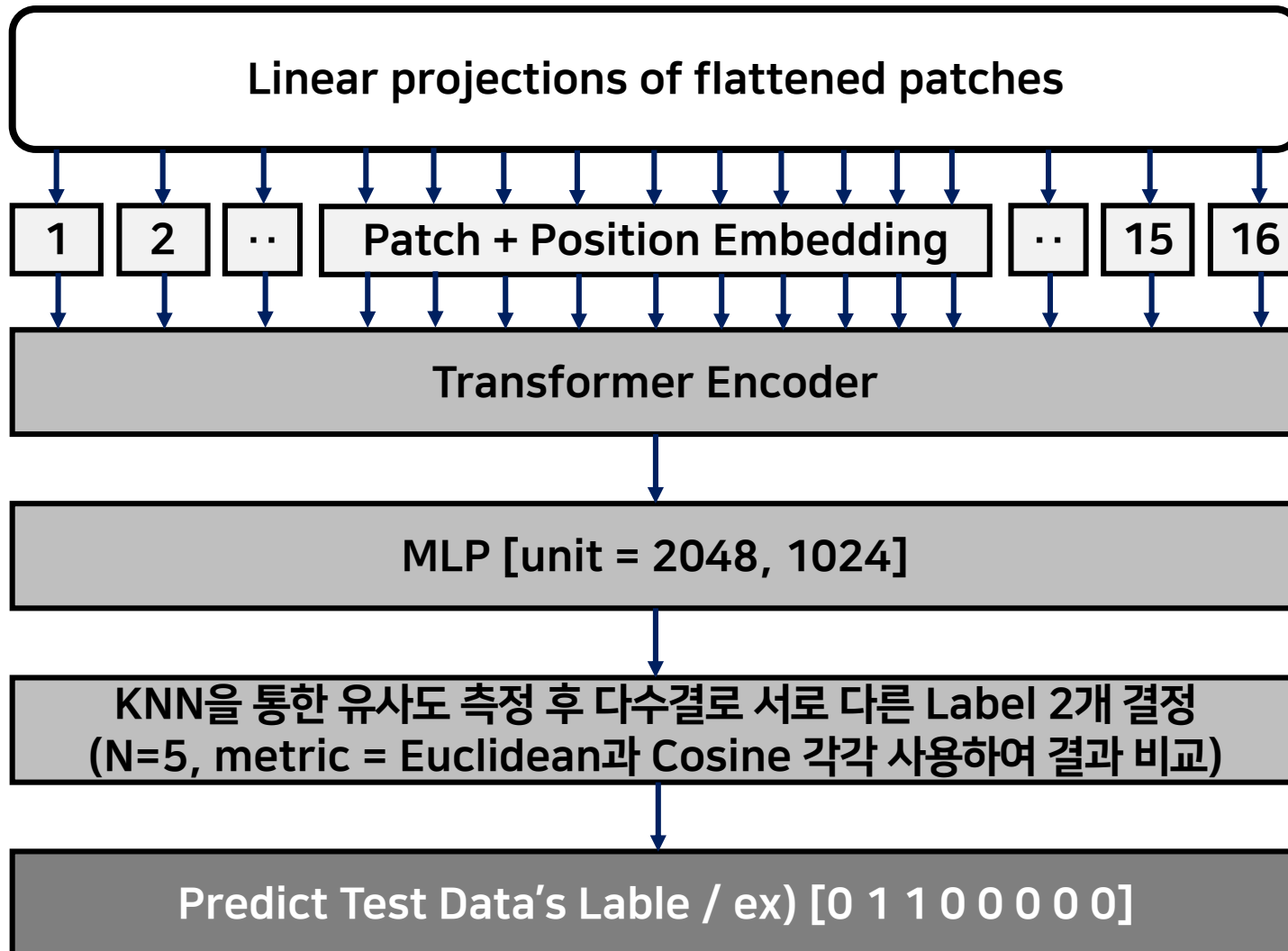
L+S



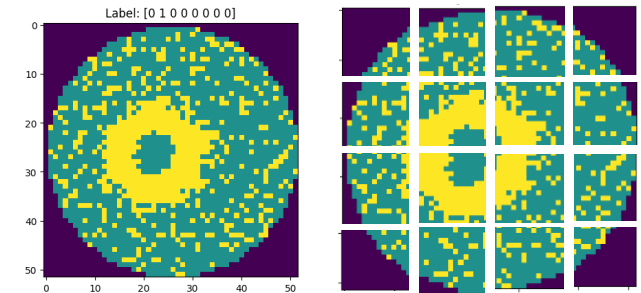
연구 프레임워크 및 과정 - 데이터 전처리



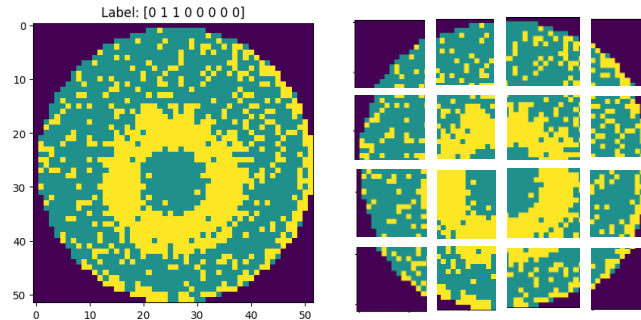
연구 프레임워크 및 과정 - Vit Model Architecture



Train Data(정상+단일결함)



Test Data(이중결함)



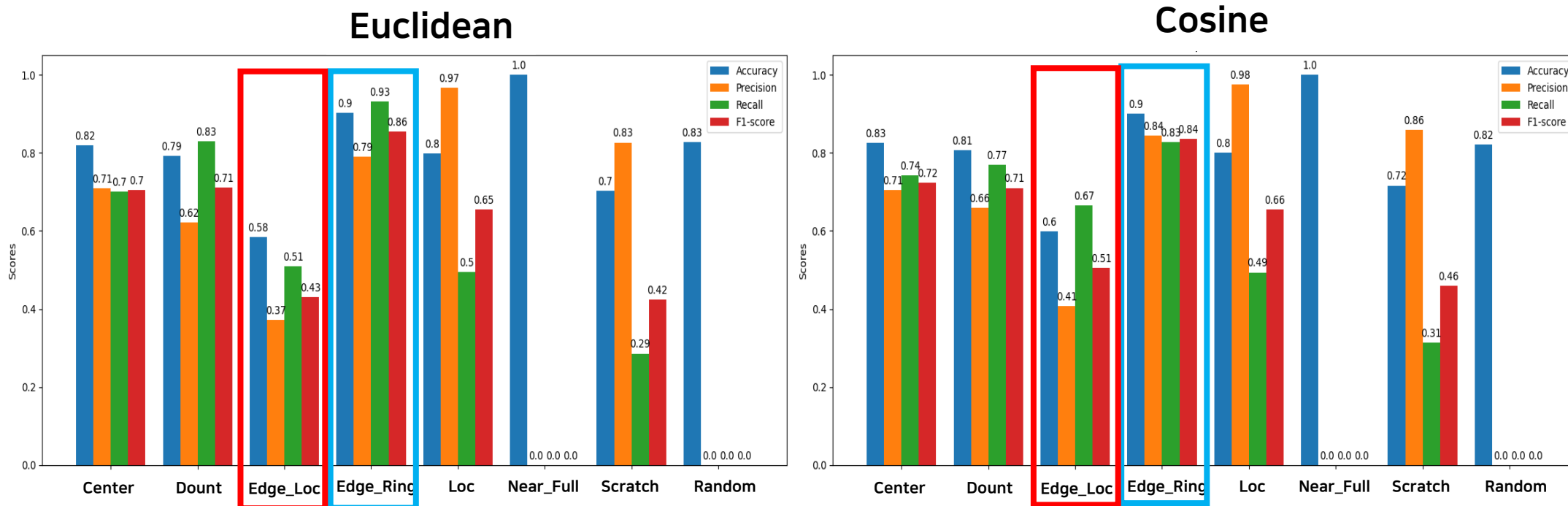
필요시, N을 1씩 추가하여 서로 다른 label 2개 도출

연구 결과 - 예측 결과

평가지표	Vit + Cosine	Vit + Euclidean
Accuracy	21.4692%	25.4154%
F1-Score	47.2479%	48.6249%
Recall	46.8894%	47.6562%
Precision	53.5908%	55.6471%

Label	유형	Vit + Cosine	Vit + Euclidean
Center+Edge_Loc	[1 0 1 0 0 0 0 0]	10.6%	61.6%
Center+Edge_Ring	[1 0 0 1 0 0 0 0]	12.5%	0.0%
Center+Loc	[1 0 0 0 1 0 0 0]	38.4%	40.4%
Center+Scratch	[1 0 0 0 0 0 1 0]	12.9%	16.8%
Donut+Edge_Loc	[0 1 1 0 0 0 0 0]	4.5%	75.0%
Donut+Edge_Ring	[0 1 0 1 0 0 0 0]	19.6%	0.0%
Donut+Loc	[0 1 0 0 1 0 0 0]	6.1%	52.0%
Donut+Scratch	[0 1 0 0 0 0 1 0]	8.0%	1.0%
Edge_Loc+Loc	[0 0 1 0 1 0 0 0]	52.1%	0.0%
Edge_Loc+Scratch	[0 0 1 0 0 0 1 0]	34.5%	50.0%
Edge_Ring+Loc	[0 0 0 1 1 0 0 0]	1.0%	0.0%
Edge_Ring+Scratch	[0 0 0 1 0 0 1 0]	0.0%	0.0%
Loc+Scratch	[0 0 0 0 1 0 1 0]	87%	83.7%

연구 결과 - 예측 결과



- Good

- 1) Edge_Ring에서 긍정적인 수치를 보여주며, Euclidean에서 Edge_Ring이 상대적으로 높다.
- 2) Near_Full과 Random에서는 혼합결함이 없었는데, 타 유형에 비해 좋은 결과를 보여줬다.

- Bad

- 1) Edge_Loc에서 굉장히 취약한 모습을 보여주고 있다.
- 2) Loc와 Scratch에서 Recall(재현율)이 Precision(정밀도)에 비해 상대적으로 낮다. → 탐지능력에 문제발생

연구 결과 - 선행연구와의 비교

Label	유형	Vit + Cosine	Vit + Euclidean	선행연구 (CNN + Convolutional Autoencoder + 임계값)
Center+Edge_Loc	[1 0 1 0 0 0 0 0]	10.6%	61.6%	59%
Center+Edge_Ring	[1 0 0 1 0 0 0 0]	12.5%	0%	69%
Center+Scratch	[1 0 0 0 0 0 1 0]	12.9%	16.8%	40%
Edge_Loc+Scratch	[0 0 1 0 0 0 1 0]	34.5%	50.0%	41.6%
Edge_Ring+Scratch	[0 0 0 1 0 0 1 0]	0%	0%	64.4%

Vit + Euclidean 모델이 **Center+Edge_Loc, Edge_Loc**는 선행연구보다 더 좋은 수치 기록했다.

하지만 **그 외의 결합유형은** 예측결과의 차이가 2개의 유형의 차이에 비해 매우 큰 격차가 존재했다.

결론 및 한계점 그리고 향후 연구

시사점

- 기존 동일 컨셉을 가진 연구보다 Edge-Loc 관련 결함유형에서 더 높은 수치의 예측률을 보여주었다.
- 해당 주제의 경우 CNN을 주로 사용하는데 Vision Transformer로 구현해보았다.

한계점

- 특정 결함은 어느 정도 탐지하지만 대다수의 결함유형을 탐지하는데 있어서 좋은 성능을 보이지 못했다.
 - Model 구현에 있어서 파라미터들의 최적값을 찾지 못했다.
- 특히, Scratch와 Edge_Loc에서 가장 낮은 수치가 나왔는데, 해당 두 결함유형은 타 결함에 비해 굉장히 두껍게 형성되지않고 얇게 형성되는 데이터임을 확인했다. 이를 통해 얇게 형성되는 결함 유형 탐지에는 부적합하다.

향후 연구방향

- 고차원인 Embedding값을 차원 축소의 중요성
- 학습되지않는 데이터에 대한 예측 방법론 (GAN을 활용한 가상 데이터, 다른 Modality 데이터)

해당 연구주제의 시사점

- 기존에 없던 혼합 결함유형을 탐지 및 예측은 미세화 및 복잡성이 증가하는 제조공정에서 빠른 진단과 대처가 매우 중요하기에 해당 주제의 연구들이 필요하다.
- 새로운 유형에 대한 정확하고 빠른 탐지 및 예측은 앞으로의 제조공정에서 비용·시간등의 여러 측면에서의 손실을 최소화할 것이다.

감사합니다.

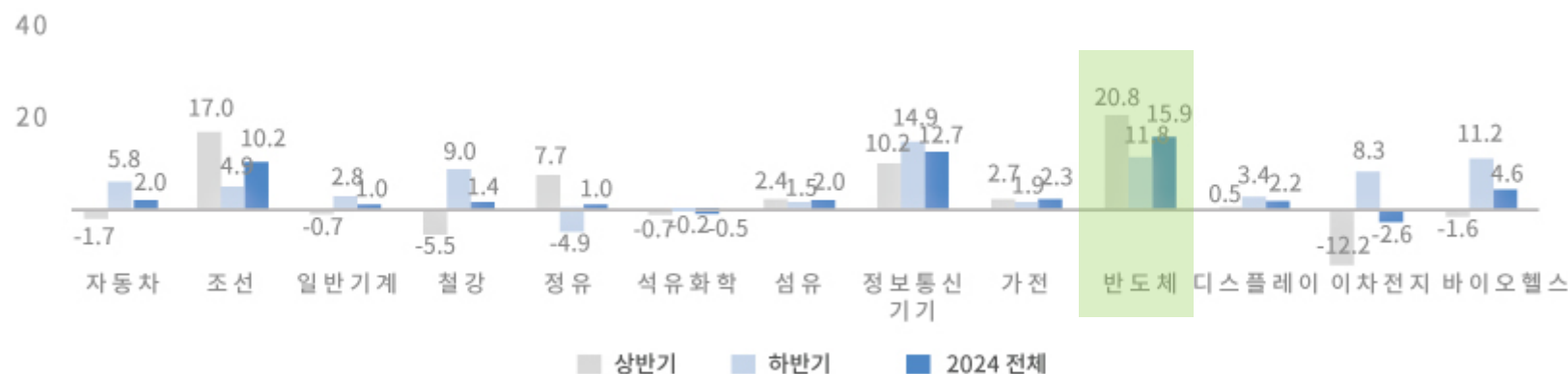
Reference

- [1] Chien, C. F., Hsu, S. C., & Chen, Y. J. (2013). A system for online detection and classification of wafer bin map defect patterns for manufacturing intelligence. *International Journal of Production Research*, 51(8), 2324-2338.
- [2] Jeong, Y. S., Kim, S. J., & Jeong, M. K. (2008). Automatic identification of defect patterns in semiconductor wafer maps using spatial correlogram and dynamic time warping. *IEEE Transactions on Semiconductor manufacturing*, 21(4), 625-637.
- [3] Kim, J., Lee, Y., & Kim, H. (2018). Detection and clustering of mixed-type defect patterns in wafer bin maps. *IEEE Transactions*, 50(2), 99-111.
- [4] Kim, H. K., & Shim, J. (2024). Generalized zero-shot learning for classifying unseen wafer map patterns. *Engineering Applications of Artificial Intelligence*, 133, 108476.
- [5] Byun, Y., & Baek, J. G. (2020, January). Mixed Pattern Recognition Methodology on Wafer Maps with Pre-trained Convolutional Neural Networks. In *ICAART* (2) (pp. 974-979).
- [6] Shim, J., & Kang, S. (2023). Learning from single-defect wafer maps to classify mixed-defect wafer maps. *Expert Systems with Applications*, 233, 120923.
- [7] Yu, N., Chen, H., Xu, Q., & Hasan, M. M. (2021, October). Wafer map defect recognition with few shot learning based on hybrid self-attention mechanism and prototype network. In *2021 China Automation Congress (CAC)* (pp. 4128-4134). IEEE.
- [8] Wang, R., & Wang, S. (2023). Similarity searching for fault diagnosis of defect patterns in wafer bin maps. *Computers & Industrial Engineering*, 185, 109679.
- [9] Zhao, J., & Tang, L. (2023, June). Metavit-Trans: A Framework for Mixed-Type Defect Detection of Wafers with Vision Transformer Combined with Meta-Learning and Transfer Learning. In *2023 China Semiconductor Technology International Conference (CSTIC)* (pp. 1-3). IEEE.
- [10] Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A. N., ... & Polosukhin, I. (2017). Attention is all you need. *Advances in neural information processing systems*, 30.
- [11] Dosovitskiy, A., Beyer, L., Kolesnikov, A., Weissenborn, D., Zhai, X., Unterthiner, T., ... & Houlsby, N. (2020). An image is worth 16x16 words: Transformers for image recognition at scale. *arXiv preprint arXiv:2010.11929*.

Appendix

연구배경 - 반도체 산업

2024년 13대 주력산업의 수출 증감률 전망 단위 : %, (전년동기비)



IT 및 혁신 의약 제품
수요 증가와 기저효과

ICT 신제품과 수출용
중간재 수요 증가로
내수 회복 전망

수출과 내수 회복으로
IT신산업군의
생산 확대 기대

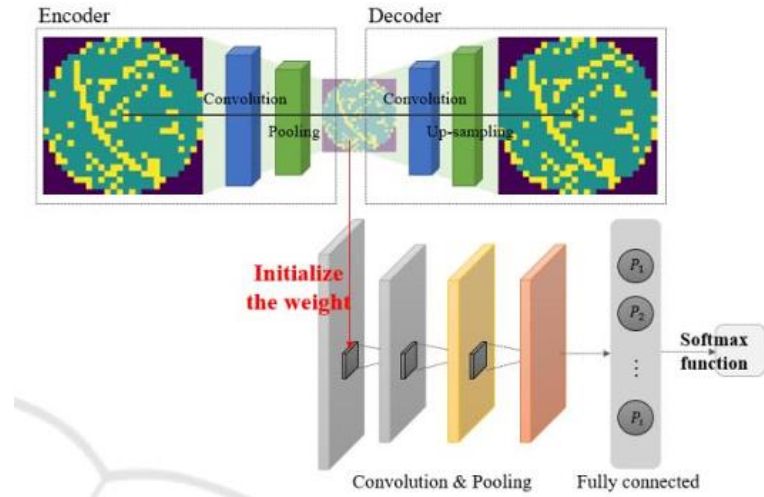
2024년 13대 주력산업의 생산 증감률 전망 단위 : %, (전년동기비)



출처: KOTRA [산업 전망] 2023년 한국 산업 전망

Mixed Pattern Recognition Methodology on Wafer Maps with Pre-trained Convolutional Neural Networks

Yunseon Byun and Jun-Geol Baek
School of Industrial Management Engineering, Korea University, Seoul, South Korea



- Convolutional Autoencoder

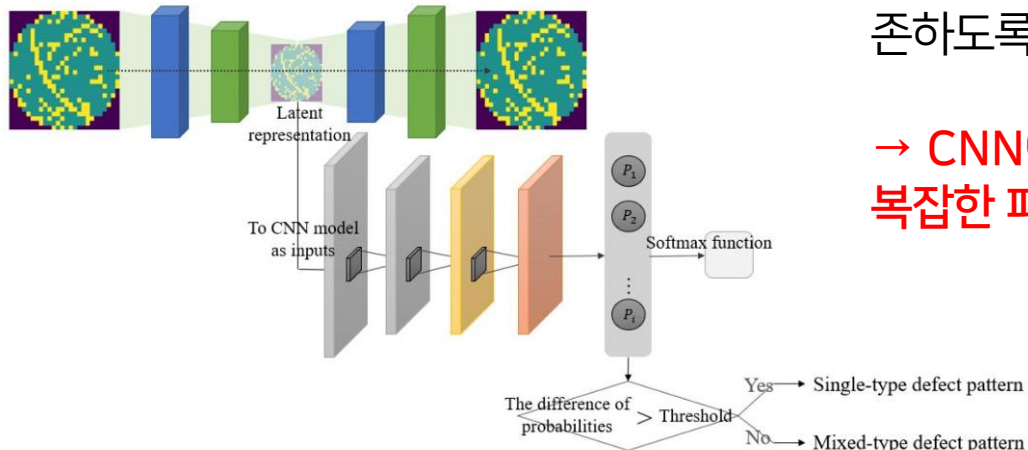
- 1) Convolutional Layer로 필터를 통해 특징을 감지.
- 2) Pooling Layer로 중요 특징을 인지 및 차원 축소.

- Encoder & Decoder

- 1) Encoder: 이미지 특징 정보를 저차원으로 압축
- 2) Decoder: Encoder를 기반으로 다시 이미지 재구성

→ Wafer Map에서 결함 패턴을 인식하는 데 중요한 공간 특징을 보존하도록 초기 가중치 조정 및 픽셀 좌표 정보 보존. 성능 향상에 도움.

→ CNN이 중요한 공간 세부 정보를 잃지 않고 웨이퍼 맵에 존재하는 복잡한 패턴을 효과적 학습 및 분류.



Mixed Pattern Recognition Methodology on Wafer Maps with Pre-trained Convolutional Neural Networks

Yunseon Byun and Jun-Geol Baek
School of Industrial Management Engineering, Korea University, Seoul, South Korea

Table 1: Classification accuracy for the single-type data.

Pattern Type	Accuracy
Center	97%
Donut	100%
Scratch	85.5%
Random	90%
Edge-Loc	87%
Edge-Ring	98.5%
Loc	85.4%
Near-full	100%

Table 2: Classification accuracy for the mixed-type data.

Pattern Type	Accuracy
Center + Edge-Loc	59%
Scratch + Edge-Loc	41.6%
Center + Edge-Ring	69%
Scratch + Edge-Ring	64.4%
Center + Scratch	40%

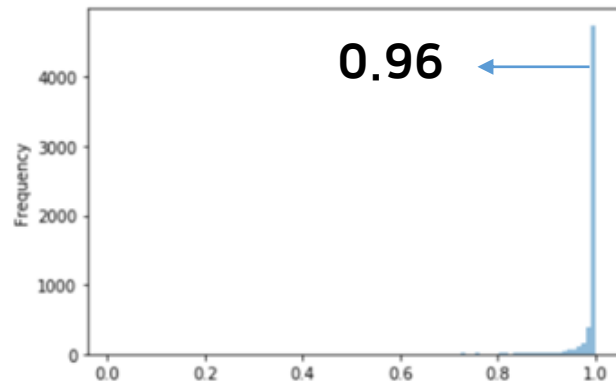


Figure 9: The histogram for the difference of probabilities.

- 단일과 혼합패턴의 구분 임계값 설정

= Maximum Class Prob – Second Maximum Class Prob
= 0.96

- 8가지의 단일결함과 5가지의 혼합결함 데이터에 테스트

- 1) 단일결함에서는 대부분 높은 수치를 보였다.
- 2) 혼합결함에서는 적합하지는 않았다.

(이유: 단일 유형 데이터와 혼합 유형 데이터의 불균형)

→ 따라서, 혼합결함인 경우에는 적합하지않았다.

Learning from single-defect wafer maps to classify mixed-defect wafer maps

Jaewoong Shim^a, Seokho Kang^{b,*}

^a Department of Industrial Engineering, Seoul National University of Science and Technology, 232 Gongneung-ro, Nowon-gu, Seoul 01811, Republic of Korea

^b Department of Industrial Engineering, Sungkyunkwan University, 2066 Seobu-ro, Jangnan-gu, Suwon 16419, Republic of Korea

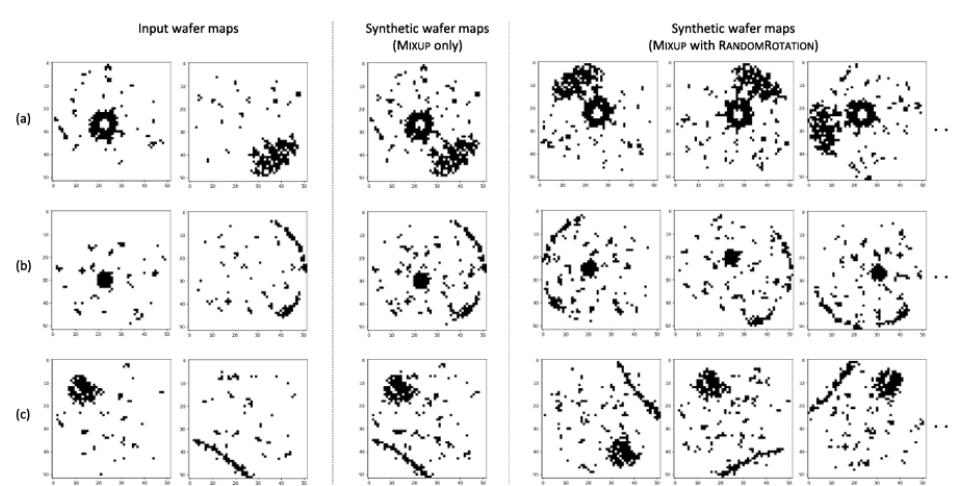


Fig. 2. Examples of synthetic wafer maps created based on Mixup only and Mixup with RANDOMROTATION.

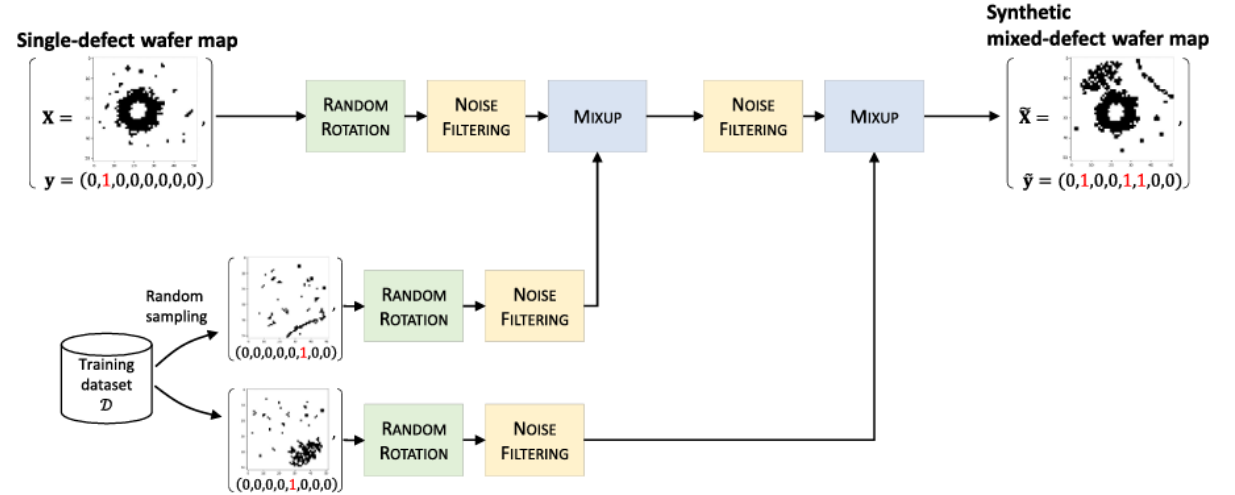


Fig. 1. Schematic description of wafer map synthesis using the proposed TRANSFORM function with $t = 3$.

- Mixup

= 각 픽셀의 위치에서 최대값만 선택 후, Mixup 방식 실행함.

- Random Rotation

= 정규분포에 따라 0~360도 회전 + 50%의 확률로 좌우반전.

- Noise Filtering

= $K=3$, $\theta=4/9$ 로 지정하여 3×3 공간 내에 있는 픽셀 평균값이 $4/9$ 이하이면 픽셀값 0.

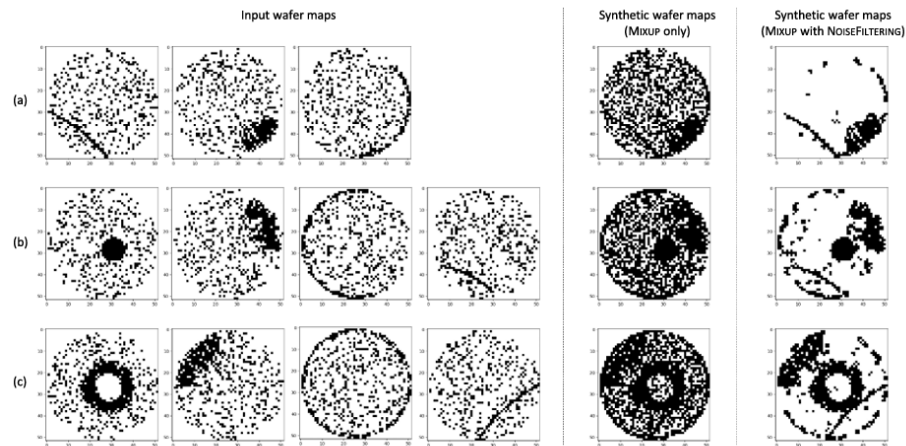


Fig. 3. Examples of synthetic wafer maps created based on Mixup only and Mixup with NOISEFILTERING.

Wafer map defect recognition with few shot learning based on hybrid self-attention mechanism and prototype network

- Positional self-attention module

= 웨이퍼 맵에서 결함 위치의 공간적 상호의존성을 학습하여, 결함 위치가 최종 네트워크 출력에 미치는 영향을 강조함.

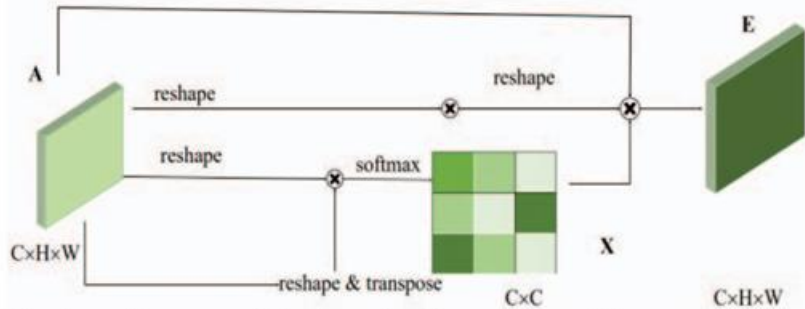


Fig. 2. Structure of the channel self-attention module

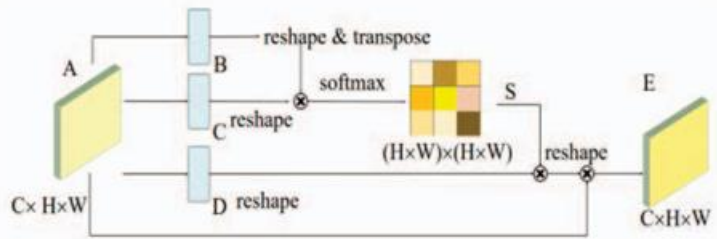


Fig. 3. Structure of the positional self-attention module

1. 3-way 1-shot
2. 3-way 5-shot
3. 5-way 1-shot
4. 5-way 5-shot

1) Wafer Map에서 결함 위치의 공간적 상호의존성 학습하여 결함 위치가 최종 네트워크 출력에 미치는 영향을 강조함.

2) Convolution Layer를 통해 차원 축소 특징 = $R^{C \times H \times W}$
(R=실수집합, C=채널 수, H=높이, W=너비)

3) A를 새로운 특징 맵 B, C로 변환

4) B와 C의 전치행렬 간의 곱셈을 수행하고, Softmax를 통해 공간적 어텐션 맵 $S \in R^{H \times W}$ 생성함.

5) 원래의 특징 맵 A와 공간적 어텐션 맵 S를 가중합하여 최종 출력 E를 생성.

- Channel self-attention module

= 채널 간의 상호의존성을 학습하여, 각 채널이 결함 특징 정보에 미치는 영향을 강조합니다.

1) A = 차원 축소된 특징 Map

2) A와 A의 전치행렬 간의 곱으로 $X \in R^{C \times C}$ 를 생성

3) 원래의 특징 맵 A와 Channel attention map X를 가중합하여 최종 출력 O를 생성.

Wafer map defect recognition with few shot learning based on hybrid self-attention mechanism and prototype network

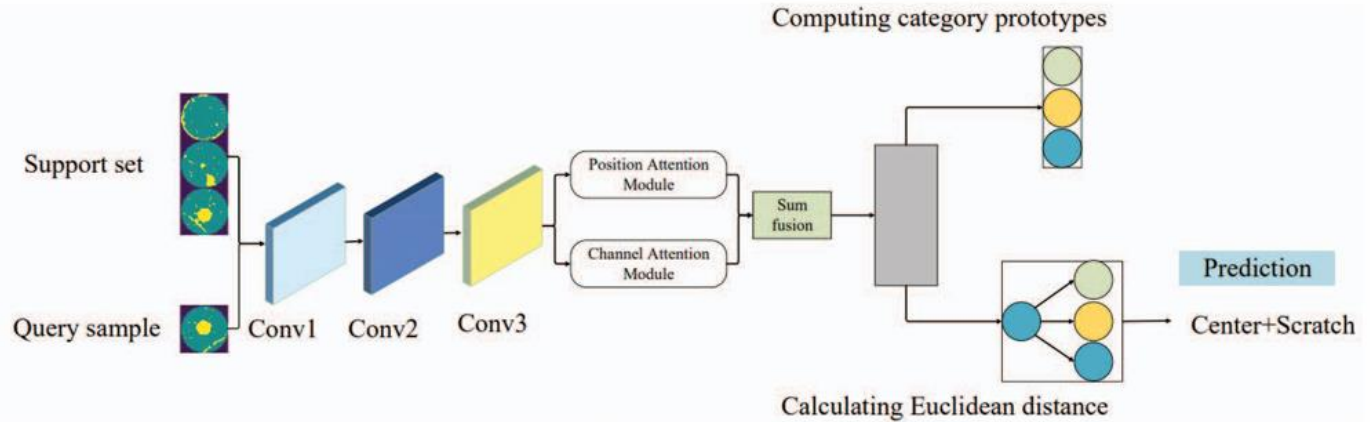
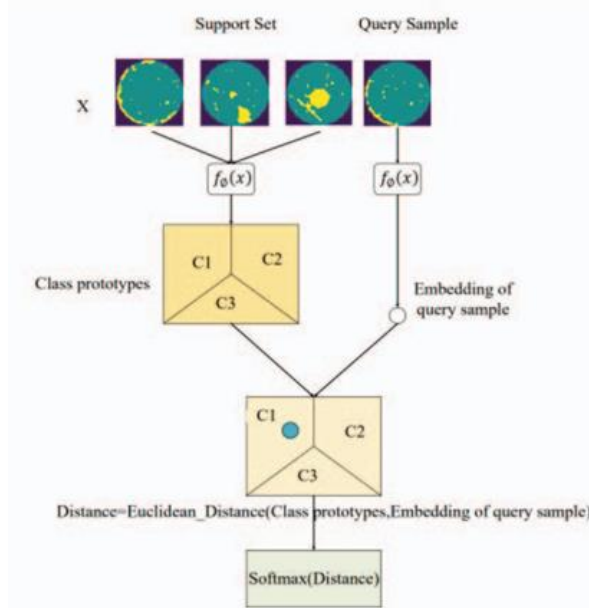


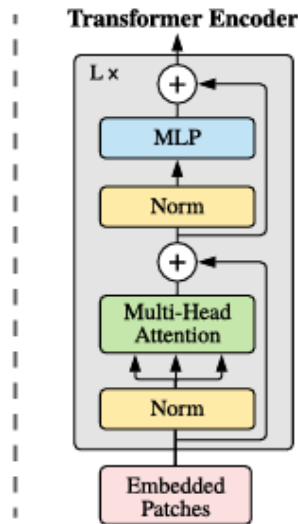
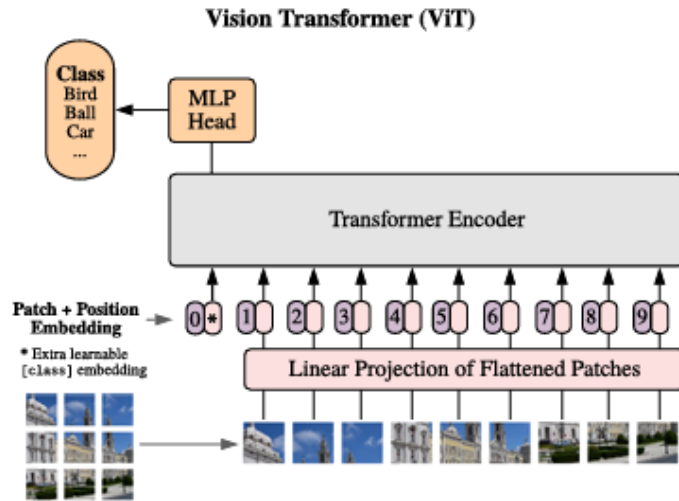
Fig. 5. Small sample wafer defect pattern detection model based on hybrid self-attention mechanism and Prototypical Network

- Model Process

- 1) 임베딩: 입력 샘플 데이터를 임베딩 공간으로 매핑하여 임베딩 특징을 추출.
- 2) 프로토타입 계산: 각 클래스의 지원 샘플 임베딩 특징의 평균을 계산하여 해당 클래스의 프로토타입을 생성.
- 3) 거리 측정: Test Sample과 각 Class(Label)의 프로토타입 간의 유클리드 거리를 계산.
- 4) 분류: Softmax를 통해 각 테스트 샘플이 어느 클래스에 속하는지 결정.

AN IMAGE IS WORTH 16x16 WORDS:
TRANSFORMERS FOR IMAGE RECOGNITION AT SCALE

Alexey Dosovitskiy^{*1}, Lucas Beyer^{*}, Alexander Kolesnikov^{*}, Dirk Weissenborn^{*},
Xiaohua Zhai^{*}, Thomas Unterthiner, Mostafa Dehghani, Matthias Minderer,
Georg Heigold, Sylvain Gelly, Jakob Uszkoreit, Neil Houlsby^{*1}
^{*}equal technical contribution, ¹equal advising
Google Research, Brain Team
{adosovitskiy, neilhoulshby}@google.com



1. 이미지를 고정된 크기의 패치로 분할하고 각 패치를 선형으로 투입.

2. 위치 임베딩을 추가하고 결과 벡터 시퀀스를 표준 Transformer Incoder에 공급.

3. 분류를 수행하기 위해 시퀀스에 추가 학습 가능한 "분류 토큰"을 추가.

Step 1. 이미지 $x \in \mathbb{R}^{C \times H \times W}$ 가 있을 때, 이미지를 $(P \times P)$ 크기의 패치 $N (=H \times W / P^2)$ 개로 분할하여 sequence $x_p \in \mathbb{R}^{N \times (P^2 \times C)}$ 를 생성.
→ (H, W) 는 원본 이미지의 해상도, C 는 채널의 수, (P, P) 는 이미지 패치의 해상도

Step 2. Trainable linear projection을 통해 x_p 의 각 패치를 flatten한 벡터 D 차원으로 변환한 후, 이를 패치 임베딩으로 사용함.

Step 3. Learnable class 임베딩과 패치 임베딩에 learnable position 임베딩을 추가.
여기서 Learnable class는 BERT 모델의 [class] 토큰과 같이 classification 역할을 수행함.

Step 4. 임베딩을 Transformer encoder에 input으로 넣어 마지막 layer에서 class embedding에 대한 output인 image representation을 도출함.

image representation = L 번의 encoder를 거친 후의 output 중 learnable class 임베딩과 관련된 부분을 의미.

Step 5. MLP에 image representation을 input으로 넣어 이미지의 class를 분류함.

- 효율적인 학습 및 확장성

1. 단순하고 효율적인 구조

= ViT는 NLP에서 성공한 Transformer 구조를 거의 변경하지 않고 사용합니다. 이는 복잡한 CNN 구조와 달리 더 단순하며, 기존의 NLP Transformer 아키텍처와 효율적인 구현을 거의 그대로 사용할 수 있음.

2. 효율적인 확장

= ViT는 매우 큰 모델로 확장할 수 있으며, 이는 더 많은 데이터와 계산 자원이 주어질 때 성능이 더욱 향상됩니다. NLP에서와 같이, 더 큰 데이터셋에서 사전 학습된 모델은 더 작은 데이터셋에서의 전이 학습에서도 우수한 성능을 보임.

- 성능 향상

1. 대규모 데이터셋에서의 성능

= ViT는 대규모 데이터셋에서 사전 학습을 통해 CNN을 능가하는 성능을 보입니다. 예를 들어, ImageNet-21k 및 JFT-300M과 같은 대규모 데이터셋에서 사전 학습된 ViT 모델이 좋은 결과를 보여줌.

2. 전이 학습에서의 우수한 성능: ViT는 대규모 데이터셋에서 사전 학습된 후, 중소형 데이터셋에서의 전이 학습에서도 CNN을 능가하는 성능을 보임. 이는 대규모 데이터셋에서 학습된 일반적인 특징들이 다양한 이미지 인식 작업에 효과적으로 적용될 수 있음을 의미함.

- 계산 자원 효율성

1. 낮은 계산 비용

= ViT는 CNN보다 학습에 필요한 계산 자원이 적다. 이는 특히 대규모 데이터셋에서 학습할 때 중요한 장점입니다. ViT는 동일한 또는 더 나은 성능을 달성하면서도 더 적은 계산 자원으로 학습할 수 있다.

2. 효율적인 학습 과정

= ViT는 대규모 데이터셋에서 사전 학습된 후, 소규모 데이터셋에서의 미세 조정을 통해 높은 성능을 달성합니다. 이는 효율적인 학습 과정을 통해 계산 자원을 절약할 수 있음을 의미한다.

연구 프레임워크 및 과정 - Wafer Bin Map 데이터 수집

Mixed-type Wafer Defect Datasets

Institute of Intelligent Manufacturing, Donghua University



Data Card Code (2) Discussion (1) Suggestions (0)

About Dataset

Motivation

Defect pattern recognition (DPR) of wafer maps, especially the mixed-type defect, is critical for determining the root cause of production defects. We collected a large amount of wafer map data in a wafer manufacturing plant. These wafer maps are obtained by testing the electrical performance of each die on the wafer through test probes. However, there are big differences between the quantity distribution of various patterns of wafer maps actually collected.

To maintain the balance between the various patterns of data, we used the generative adversarial networks to generate some wafer maps to maintain the balance of the number of samples among the patterns. Finally, about 38,000 mixed-type wafer map defect dataset is formed, which are used to identify mixed-type wafer map defect and assist the research on the causes of defects in the wafer manufacturing process. In order to facilitate researchers, students, and enthusiasts in related fields to better understand the causes of defects in the wafer manufacturing process, we public this dataset of mixed-type wafer map defects for you to research.

Citation

출처: Kaggle - Mixedwm38 Dataset (npz 파일형식)

Usability ⓘ

8.75

License

Data files © Original Authors

Expected update frequency

Annually

Tags

Computer Science

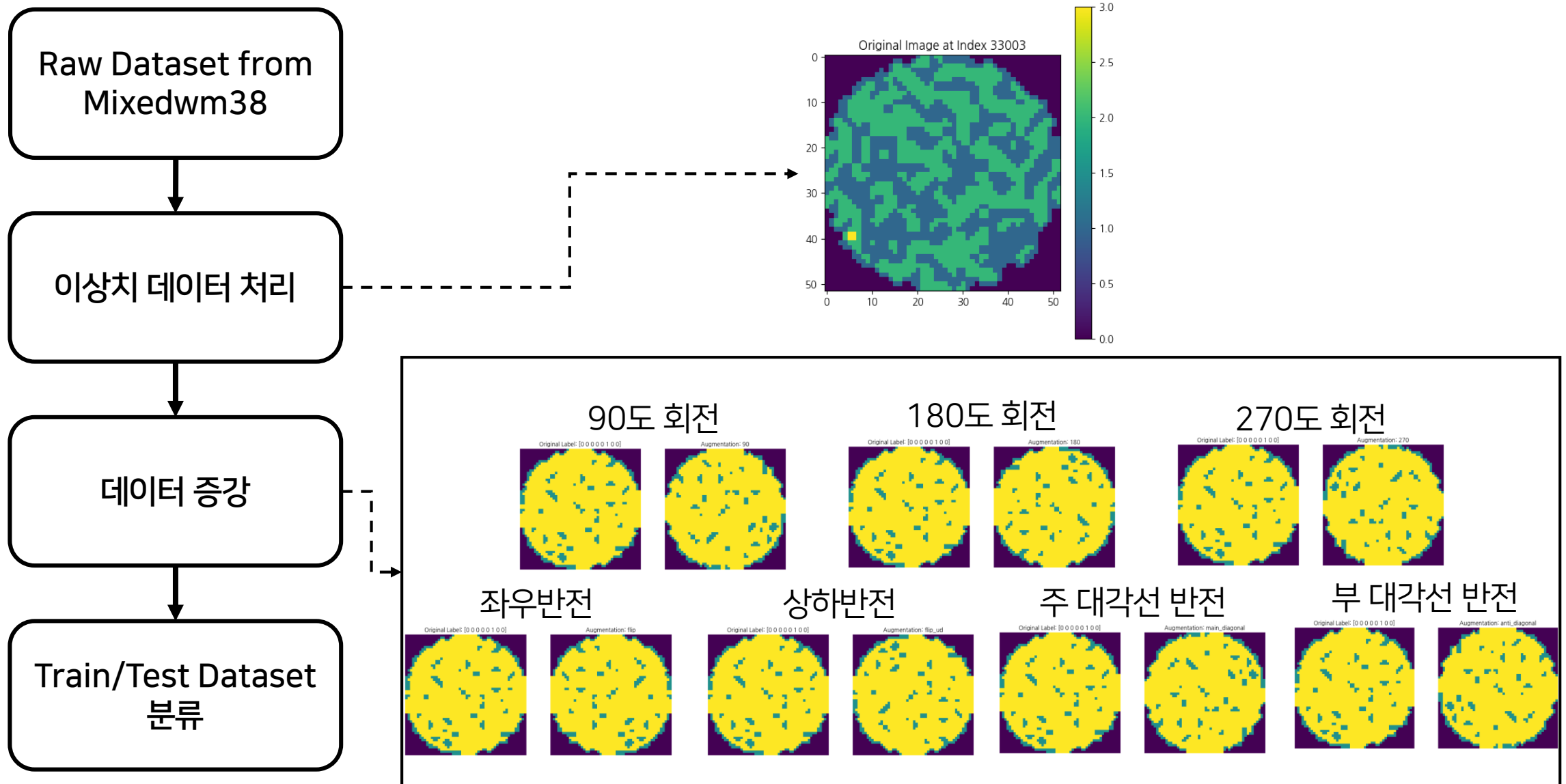
Universities and Colleges

Image

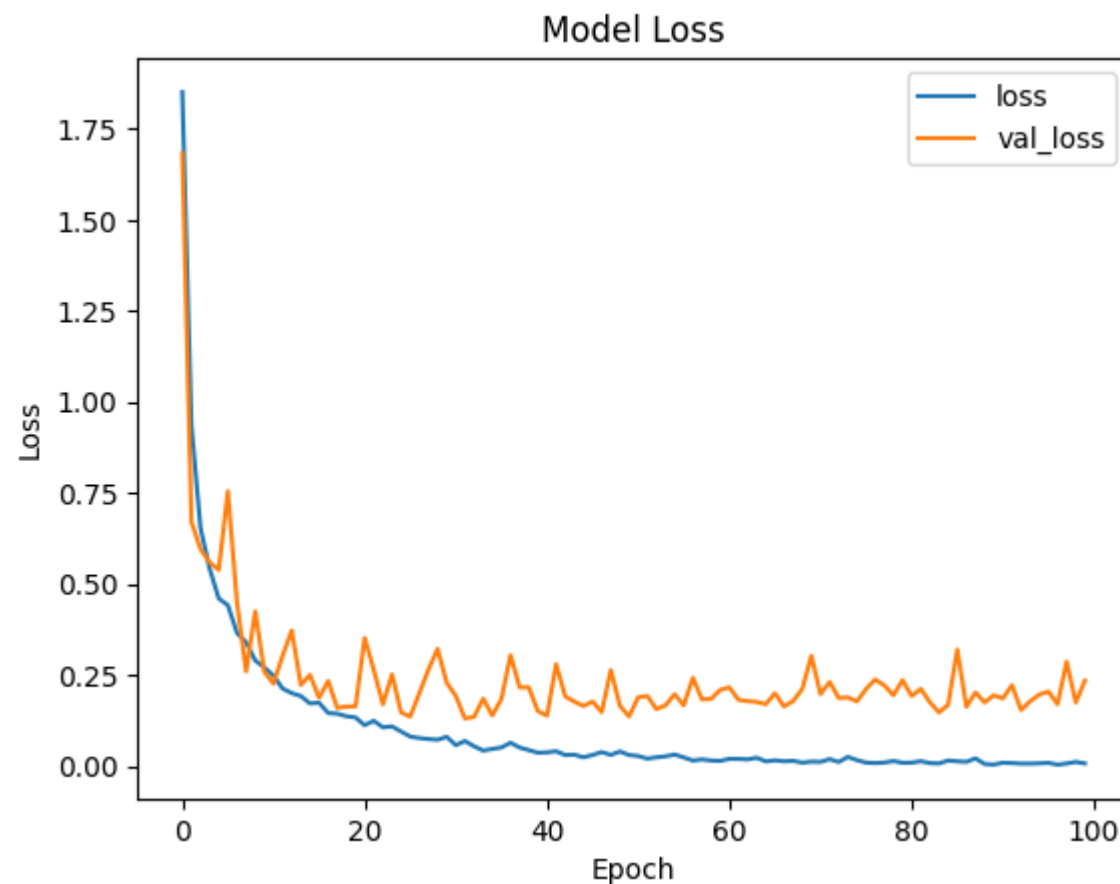
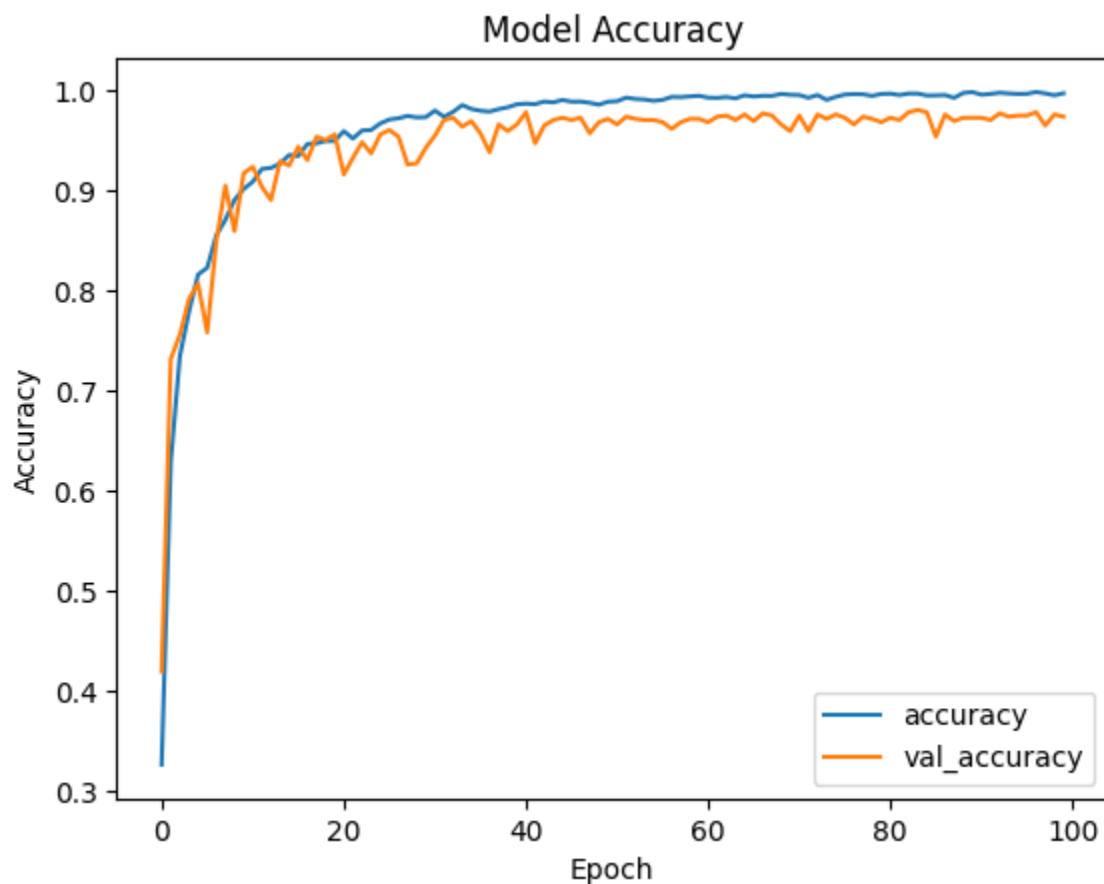
데이터 결함유형 설명 - 정상 + 단일결함

결함 패턴	텍스트 설명
Normal	일반 패턴은 웨이퍼에 결함이 없는 것이 특징입니다.
Center	센터결함(Central Defect)은 웨이퍼 중앙에 클러스터(cluster)가 생기는 것이 특징으로 플라즈마 영역이나 박막 증착, 산화 등의 문제로 인해 발생한다.
Dount	도넛 결함은 환형 클러스터가 특징이며 현상 공정 중 웨이퍼 표면에 남아 있는 용해된 포토레지스트 고형물이 재증착되어 발생합니다. 중앙 세척 단계에서 웨이퍼 중앙의 결함이 줄어들기 때문입니다.
Edge_loc	Edge loc은 웨이퍼 가장자리에 국부적인 클러스터가 나타나는 것을 특징으로 하며 확산 공정 중 가열이 고르지 않거나 식각 중 문제로 인해 발생합니다.
Edge_ring	에지 링 결함은 웨이퍼 주변에 링 모양의 클러스터가 나타나는 것을 특징으로 하며, 급속 열 어닐링 공정에서 에지의 비정상적인 온도 제어나 리소그래피 중 문제, 저장 노드 공정에서 층간 정렬 불량으로 인해 발생합니다.
Loc(Localized)	Loc 결함은 국부적인 클러스터가 특징이며 주어진 기계에서 과도한 진동으로 인해 발생하여 충분한 입자 또는 결정질 이질성이 발생합니다.
Near_Full	Nearfull Defect는 웨이퍼의 다이 중 90% 이상이 불량인 경우를 말합니다.
Scratch	스크래치 불량은 직선과 곡선이 특징이며, 배송 및 취급 과정에서 사람의 실수나 화학적 기계적 연마 과정에서 패드가 경화되어 발생합니다.
Random	무작위 결함은 패턴 중 하나에 기인할 수 없는 하나 이상의 결함이 특징이며, 제조 환경 요인에 의해 발생하며 체계화되지 않습니다.

연구 프레임워크 및 과정 - 데이터 전처리



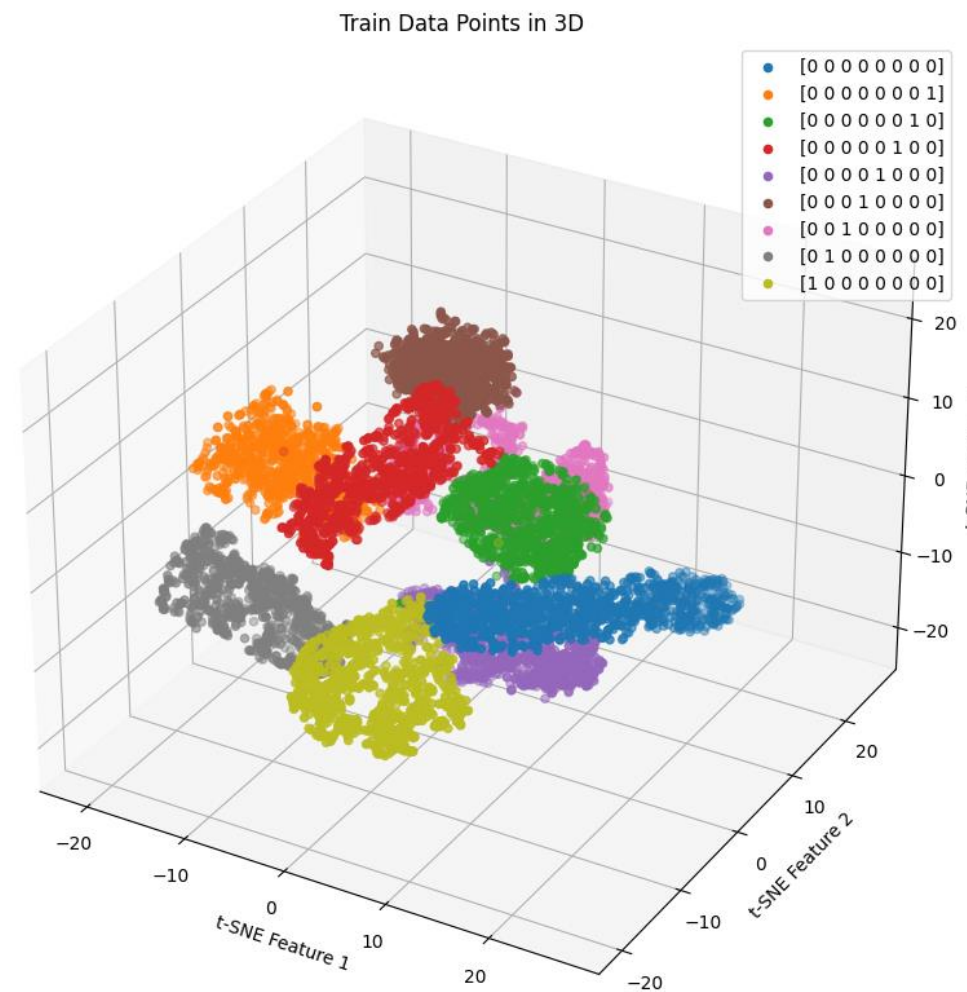
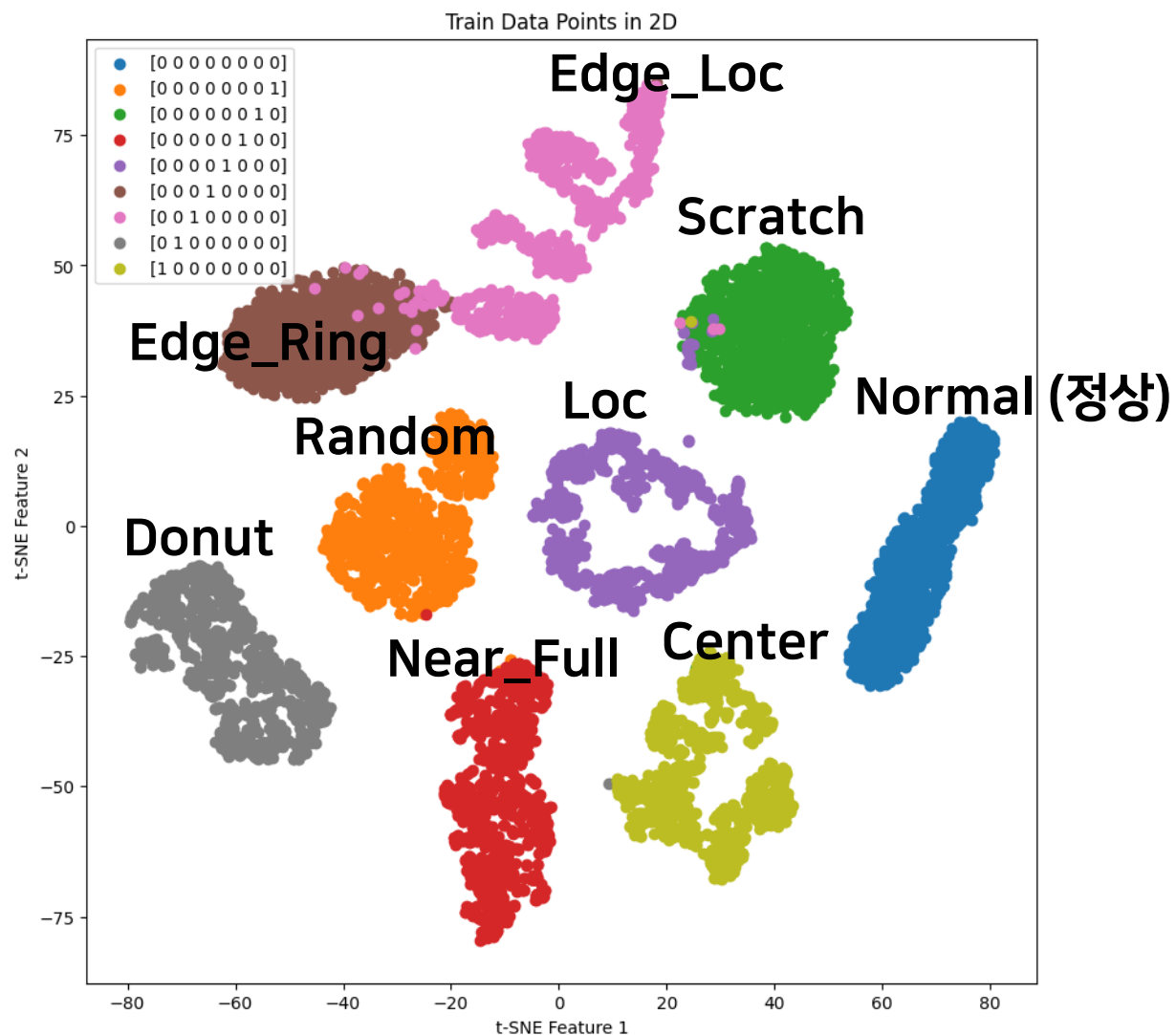
Vit 모델학습 - Train 데이터에 대한 학습



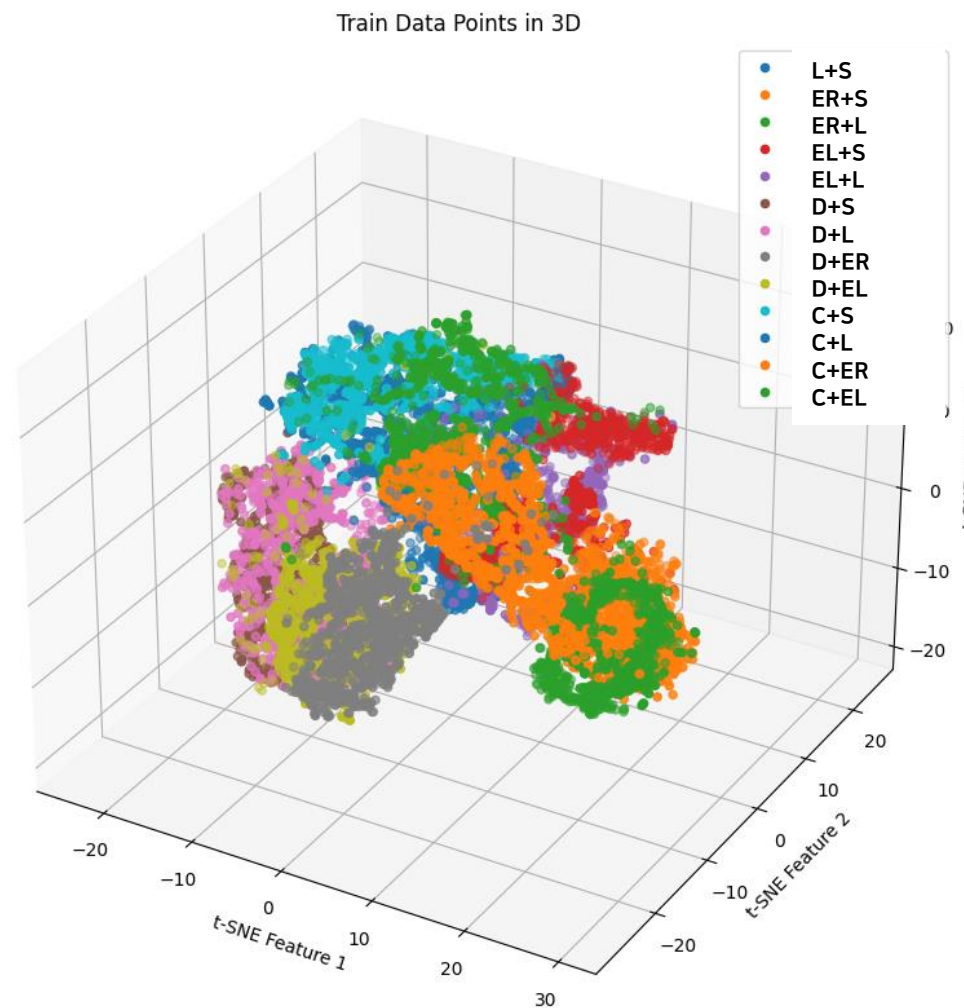
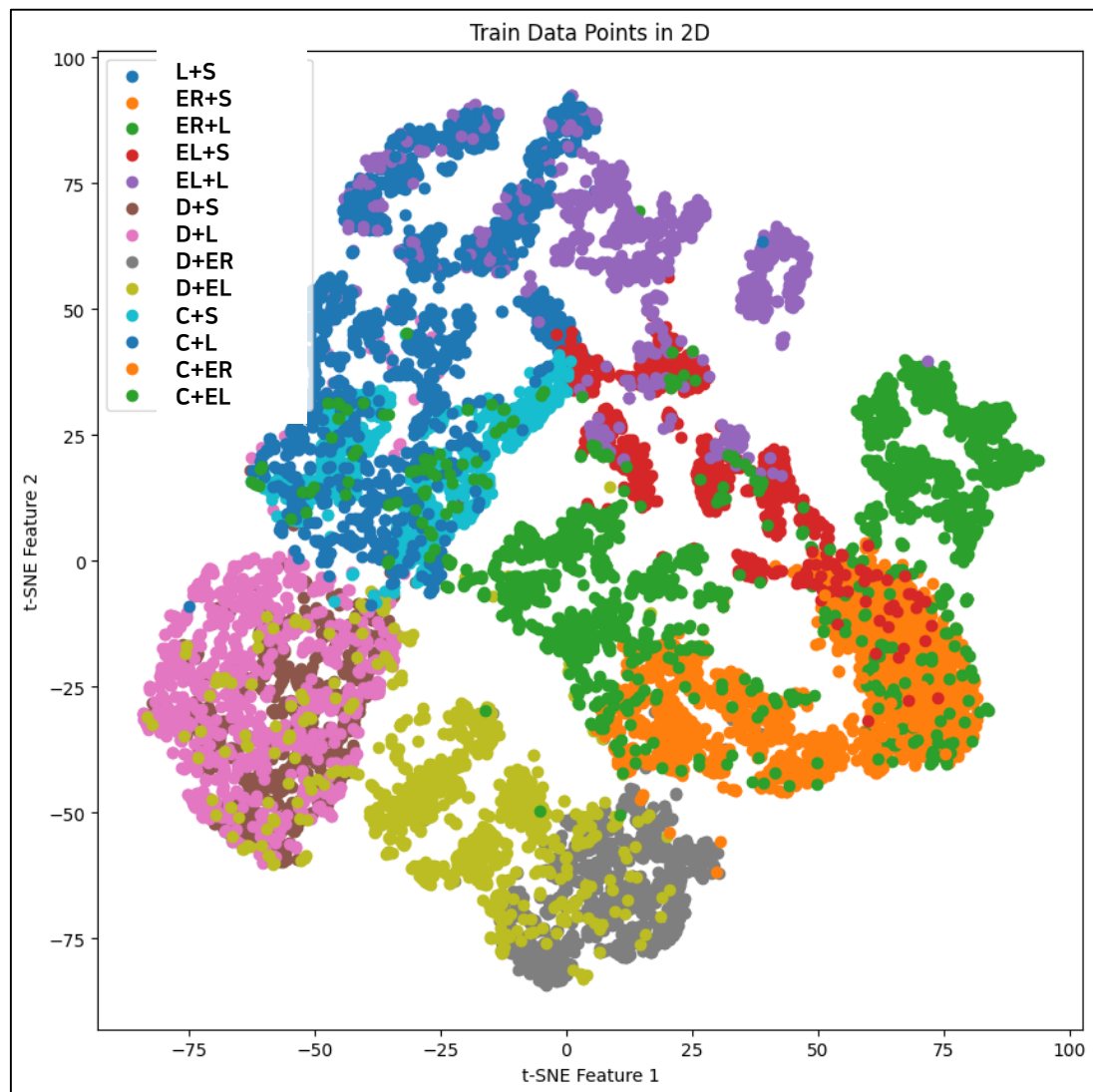
Vit Model: Epochs = 100, batch_size = 128, Validation split: 0.1, Monitor: val_loss

Epoch 32/100 64/64 [=====] - 6s 89ms/step - loss: 0.0687 - accuracy: 0.9743 - val_loss: 0.1306 - val_accuracy: 0.9711

연구 프레임워크 및 과정 - Train Data Embedding



연구 프레임워크 및 과정 - Test Data Embedding



ViT 모델 구조 - 기본 구조

- Patch Extraction

입력 이미지는 고정 크기의 패치(16×16)로 나뉘어진다. 이후 각 패치는 평탄화되어 재구성됩니다. 이후 size와 strides 파라미터를 통해 패치의 크기와 패치를 추출할 간격을 지정한다.

- Patch Embedding

각 패치는 밀집 벡터 공간으로 Projection 되고 위치 인코딩으로 보강된다. 이 단계는 모델이 원본 이미지의 공간 정보를 유지하도록 한다. Projection 층은 각 패치를 고차원 공간으로 변환하고 Position_embedding 층은 각 패치에 위치 정보를 추가한다.

- Transformer Encoder Layers

모델은 임베드 된 패치를 처리하기 위해 여러 Transformer Encoder Layer를 적용한다. 각 레이어는 Layer Normalization을 통해 입력을 정규화하여 학습 안정성을 높인다. Multi-Head Self-Attention은 입력 시퀀스의 각 위치에 대해 다른 위치를 참조하여 정보를 추출한다. MLP 블록을 통해 각 위치에 독립적으로 적용되는 두 개의 밀집 레이어를 적용한다. Residual Connection은 입력과 출력을 더하여 그라디언트(gradient) 소실 문제를 완화한다. 최종적으로는 logits 함수에 Softmax Function을 적용하여 학습 데이터의 단일 결함의 유형 클래스에 따라 확률 분포를 생성한다. 이는 해당 클래스일 확률을 의미한다.

- Multi-Layer Perceptron (MLP)

MLP Layer는 추가 학습 용량을 제공하고 모델 학습에 있어 반도체 결함의 복잡한 패턴을 학습할 수 있도록 한다. Activation Function은 GELU를 사용했고, Dropout Layer를 포함한 밀집 레이어로 구성된다.

- Training

해당 모델 학습은 지수 학습률 감소 스케줄을 설정한 Adam Optimizer를 사용했다. Checkpoint도 사용하여 Validation loss를 기준으로 최상의 모델을 저장하도록 했다. 초기 학습률은 이고 1,000 step마다 0.9로 감소하게 스케줄을 생성했다. 해당 방법을 통해 학습이 진행됨에 따라 학습률을 감소시켜 안정적인 모델로 수렴하도록 한다. batch_size는 128, epochs는 100으로 설정했다.

- Embedding Extraction

모델의 특정 레이어에서 96차원의 임베딩 벡터를 추출하여 Train Data와 Test Data의 임베딩 값에 대해 분석하도록 생성했다. 모델의 마지막 두 번째 레이어의 출력을 임베딩 벡터로 추출하도록 한다.

- KNN 유사도 계산

테스트 데이터의 추출된 96차원의 임베딩 벡터를 학습 데이터와의 유사도를 Euclidean과 Cosine으로 계산하고, 예측 결함 유형(레이블)을 생성할 수 있게 했다. KNN의 N은 5로 설정했으며 5개의 이웃 데이터에서 다수결로 2개의 서로 다른 결함 유형을 선택하도록 설정했다. 만약 5개에서 결정이 안 나면 N을 1씩 증가시켜서 결정하도록 했다.

ViT 모델 구조 - KNN과 유사도 기반 알고리즘

Algorithm 1 Calculate KNN Similarity

```
1: procedure CALCULATEKNNSIMILARITY(train_embeddings, test_embeddings, y_train, y_test, k =  
5)  
2:   results  $\leftarrow$  {}  
3:   for metric in ['euclidean', 'cosine'] do  
4:     nbrs  $\leftarrow$  NearestNeighbors(n_neighbors = k, algorithm =  
auto, metric = metric).fit(train_embeddings)  
5:     distances, indices  $\leftarrow$  nbrs.kneighbors(test_embeddings)  
6:     predictions  $\leftarrow$  []  
7:     for i, test_embed in enumerate(test_embeddings) do  
8:       knn_indices  $\leftarrow$  indices[i]  
9:       knn_labels  $\leftarrow$  y_train[knn_indices]  
10:      unique_labels  $\leftarrow$  np.unique(knn_labels, axis = 0)  
11:      current_k  $\leftarrow$  k  
12:      while len(unique_labels) < 2 do  
13:        current_k  $\leftarrow$  current_k + 1  
14:        nbrs.set_params(n_neighbors = current_k)  
15:        distances, indices  $\leftarrow$  nbrs.kneighbors(test_embeddings)  
16:        knn_indices  $\leftarrow$  indices[i]  
17:        knn_labels  $\leftarrow$  y_train[knn_indices]  
18:        unique_labels  $\leftarrow$  np.unique(knn_labels, axis = 0)  
19:      end while  
20:      label_counts  $\leftarrow$  np.sum(knn_labels, axis = 0)  
21:      top_two_labels  $\leftarrow$  np.argsort(label_counts)[-2 :]  
22:      predicted_label  $\leftarrow$  np.zeros_like(knn_labels[0])  
23:      predicted_label[top_two_labels]  $\leftarrow$  1  
24:      predictions.append({'ActualLabel' : y_test[i], 'PredictedLabel' :  
predicted_label, 'KNN Labels' : unique_labels})  
25:    end for  
26:    results[metric]  $\leftarrow$  pd.DataFrame(predictions)  
27:  end for  
28:  return results  
29: end procedure
```

$$\text{Euclidean distance} = \sqrt{\sum_{i=1}^d (a_i - b_i)^2}$$

$$\text{Cosine similarity} = \frac{\mathbf{a} \cdot \mathbf{b}}{\|\mathbf{a}\| \|\mathbf{b}\|}$$

$$\mathbf{a} \cdot \mathbf{b} = \sum_{i=1}^d a_i b_i$$

$$\|\mathbf{a}\| = \sqrt{\sum_{i=1}^d a_i^2}$$

$$\|\mathbf{b}\| = \sqrt{\sum_{i=1}^d b_i^2}$$