

반도체 웨이퍼 데이터 증강에 따른 CNN 모델 성능 평가

CUAI 7기 DA1팀

박성호(전자전기공학부), 박준상(전자전기공학부), 양지훈(전자전기공학부), 조효원(전자전기공학부),
황지민(전자전기공학부)

[요약]

반도체 제조 공정의 높은 정밀성과 비용 때문에, 실시간 불량 예측과 신속한 대응이 필수적이다. 본 연구는 웨이퍼 센서 데이터를 활용하여 불량 유무와 유형을 실시간으로 예측하고 맞춤형 대응을 가능하게 하는 모델을 제안한다. 데이터 부족을 극복하기 위해 원본 데이터, 정규분포 증강 데이터, GAN 생성 데이터를 비교하여 예측 정확도를 평가했다. 이를 통해 불량 발생을 최소화하고 생산 효율성을 높여 반도체 제조 공정의 비용 절감을 실현할 수 있는 방안을 제시한다.

1. 서론

반도체 산업은 국가 경제와 기술 발전에 중요한 역할을 하며, 특히 반도체 칩의 품질과 생산 수율 향상이 필수적이다. 반도체 제조 공정에서 생성되는 웨이퍼 빈 맵(WBM)은 각 칩의 결함 패턴을 시각적으로 확인하고 결함 원인을 파악하는 데 중요한 도구로 사용됩니다. 그러나 전통적으로 숙련된 기술자가 육안으로 분석하는 방식은 시간과 비용이 많이 소요되며 생산량이 증가함에 따라 정확한 분석을 유지하는 것이 점점 어려워진다.

최근 기계학습, 특히 합성곱 신경망(CNN) 기반의 모델이 WBM의 결함 패턴 분류에 효과적으로 활용되고 있다. 그러나 CNN 모델의 성능은 훈련 데이터의 양과 질에 크게 의존하며, 데이터의 불균형 문제와 데이터 부족 상황에서는 예측 정확도와 일반화 능력이 저하될 수 있다. 이러한 문제를 해결하기 위해 데이터 증강 기법의 중요성이 대두되고 있다.

본 연구에서는 CNN 모델을 사용하여 WBM의 불량 유무와 결함 패턴을 실시간으로 예측하는 다양한 방법을 평가한다. 첫째, 원본 데이터를 그대로 사용하여 CNN 모델을 학습시킨다. 둘째, 정규 분포를 기반으로 데이터 증강을 수행하여 학습 데이터의 다양성을 높인 후 모델을 학습시킨다. 셋째, GAN(Generative Adversarial Network)을 활용하여 증강된 데이터를 사용하여 모델을 학습시킨다. 모든 훈련 데이터는 28x28 픽셀 크기로 resize하여 CNN 모델의 성능을 평가하며, 각 방법에서 훈련된 모델의 예측 정확도와 일반화 능력을 비교 분석한다.

본 연구의 목표는 다양한 데이터 증강 방법을 통해 CNN 모델의 성능을 극대화하고, 반도체 제조 공정에서 불량 발생을 최소화하며 생산 효율성을 높이는 최적의 모델 학습 방법을 제시한다.

2. 본론

2.1 데이터 개요

본 연구에서 사용한 데이터는 캐글(Kaggle)에서 제공하는 WM-811K 데이터셋이다. 이 데이터셋은 실제 반도체 제조 공정에서 수집된 811,457개의 웨이퍼 맵으로 구성되어 있으며, 이 중 172,950개의 웨이퍼 맵에는 결함 패턴에 대한 라벨이 부여되어 있다. 데이터의 라벨은 결함이 없는 None 패턴과 8가지 결함 패턴 (Center, Edge-Loc, Edge-Ring, Loc, Scratch, Random, Donut, Near-Full)으로 구성되어 있다. 각 결함 패턴의 예시 이미지는 Fig. 1과 같다.

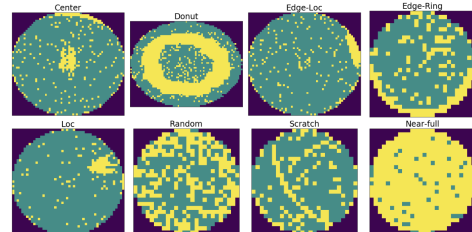


Fig 1. WM-811K 8가지 결함 패턴 예시

본 연구에서는 반도체 제조 공정에서 발생할 수 있는 다양한 결함 패턴들을 분석하고, 이를 효과적으로 분류할 수 있는 모델을 개발하는 것을 목표로 한다. 반도체 제조 과정에서 결함이 발생하는 원인을 정확히 파악하고, 이를 실시간으로 감지하여 조치하는 것은 제품 품질 향상과 생산 효율성 증대에 필수적이다. 각 결함 패턴은 특정 공정 단계에서 발생하는 문제로 인해 형성되며, 이러한 패턴들을 신속하게 분류하고 원인을 규명함으로써, 공정의 안정성을 유지하고 불량률을 최소화할 수 있다. 아래에서는 본 연구에서 다루는 주요 결함 패턴과 발생 원인에 대해 자세히 설명한다.

이 데이터셋에서 가장 큰 문제점은 데이터 불균형이다. 첫 번째로, 정상 패턴인 None 패턴이 117,945개로 라벨이 있는 데이터셋의 약 85.25%를 차지하며, 나머지

8개의 결함 패턴들은 약 14.75%를 차지한다. None 패턴을 가진 웨이퍼 맵의 수는 다른 결함 패턴들의 맵 수에 비해 압도적으로 많은 비중을 차지한다. 두 번째로, 결함 패턴들 간에도 불균형이 존재한다. 결함 패턴의 비율은 Fig. 2와 같다.

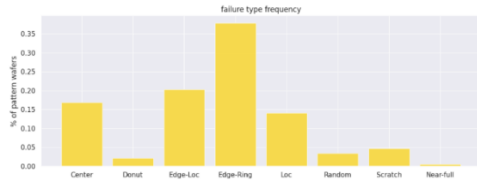


Fig 2. 8개의 결함 패턴 비율

이러한 데이터 불균형은 모델 학습 과정에서 중요한 문제를 야기할 수 있다. 정상 패턴이 과도하게 많거나, 특정 결함 패턴이 현저히 적을 경우, 모델은 빈도가 높은 유형에 과적합(overfitting) 되기 쉽다. 이로 인해 모델은 데이터의 불균형을 그대로 반영해 결함 패턴을 제대로 학습하지 못하게 될 수 있다. 따라서, 모델이 모든 결함 패턴에 고른 성능을 발휘할 수 있도록 데이터 증강을 통해 불균형 문제를 해결할 필요가 있다.

2.2 모델 소개

1) Convolution Neural Network(CNN) Classification model

본 연구에서는 대량의 웨이퍼 데이터를 반복적으로 처리할 수 있는 경량화된 CNN(Convolution Neural Network) 모델을 설계했다. 웨이퍼 이미지의 특징을 효과적으로 추출하면서도 계산 복잡도를 줄이기 위해 비교적 가벼운 모델을 채택했으며, 전체 구조는 Fig. 3와 같다.

이 모델은 총 3개의 Convolutional layer로 구성되어 있으며, 각 layer는 16, 64, 128개의 채널을 사용한다. 모든 layer의 kernel 크기는 3x3으로 설정되었으며, 활성화 함수로는 Rectified Linear Unit(ReLU)을 사용하여 비선형성을 추가했다. Convolutional layer에서 추출된 특징은 flattening과정을 거쳐 일차원 벡터로 변환되며, 이를 통해 fully-connected layer에 입력된다.

Fully-connected layer는 두 개로 구성되어 있으며 각각 512 및 128개의 뉴런을 가지고 있다. 이 layer들은 웨이퍼 이미지의 복잡한 패턴을 학습하여 최종적으로 9개의 클래스에 대한 확률을 계산한다. 최종 출력은 Softmax 활성화 함수를 사용하여 각 클래스에 대한 확률 분포를 산출한다.

모델의 성능을 향상시키기 위해 가중치를 초기화하거나 기존의 학습된 가중치를 불러오는 메커니즘이 포함되었다. 이를 통해 학습된 모델은 안정적인 성능을 유지하면서도, 필요에 따라 동일한 구조의 모델을 반복 학습하거나 가중치를 재사용할 수 있다. 이 과정에서

Adam Optimizer를 사용하여 가중치 업데이트가 이루어지며, 손실 함수는 Categorical Cross-Entropy로 설정되어 다중 클래스 분류 작업에 적합하도록 설계하였다.

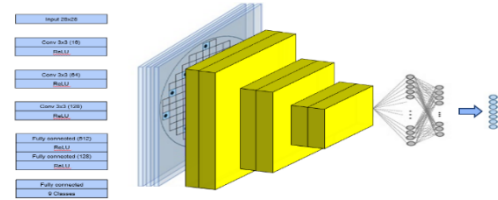


Fig 3. Convolution Neural Network Classification Model

2) Conditional AutoEncoder (CAE)

AutoEncoder는 입력 데이터를 저차원 잠재 공간으로 인코딩하고 다시 원래의 차원으로 복원하는 비지도 학습 모델이다. 입력 데이터를 압축하여 잠재 공간(latent space)에서 중요한 특성을 추출하고, 이를 통해 데이터의 특징 이해, 노이즈 제거 및 차원 축소 등에 사용된다. 인코더(Encoder)는 입력 데이터를 잠재 공간으로 변환하고, 디코더(Decoder)는 잠재 공간의 데이터를 원래의 입력 공간으로 복원하는 역할을 한다.

Convolutional AutoEncoder(CAE)는 AutoEncoder구조에 컨볼루션 레이어를 추가하여 이미지와 같은 고차원 데이터를 효율적으로 처리한다. CAE는 공간적 정보를 유지하면서 특징을 추출하는 데 강점을 가진다. CAE는 인코더를 사용해 이미지를 저차원 잠재 벡터로 변환 후, 디코더를 통해 다시 복원한다.

본 연구에서 설계한 CAE 모델은 28x28x3 Wafer 이미지 텐서를 입력으로 넣어 Convolution 및 MaxPooling layer를 통해 잠재 벡터를 생성하고, Transposed Convolution 및 Upsampling layer를 통해 복원한다. CAE 모델의 학습에는 Adam optimize, 손실 함수로는 평균 제곱 오차(MSE)를 사용하였다.

앞서 제기된 데이터 불균형 문제를 해결하기 위해 본 연구에서는 CAE를 통해 결함이 존재하는 웨이퍼 데이터의 증강을 진행했다. 학습된 CAE의 인코더를 통해 잠재 공간으로 인코딩하여 데이터의 주요 특징이 잠재 벡터로 압축되었으며, 인코딩 된 잠재 벡터에 무작위 노이즈를 추가하여 변형된 잠재 벡터를 생성하였다. 노이즈가 추가된 잠재 벡터를 CAE의 디코더에 입력하여 원래의 이미지와 유사하지만 변형된 새로운 웨이퍼 데이터를 복원하였다. Fig 4의 왼쪽은 원본 데이터, 오른쪽은 CAE를 통해 증강한 데이터다. 이로써 데이터 증강을 통해 데이터 불균형을 완화하고자 각 결함 패턴의 빈도를 대략 4000개로 조정하였다.

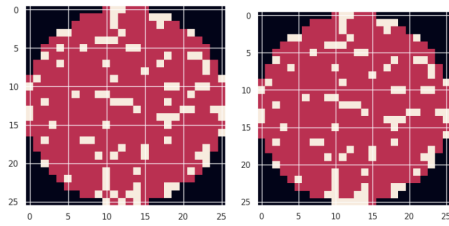


Fig 4. 원본 데이터 및 CAE를 통해 증강한 데이터

3) Data Augmentation by GAN

본 연구에서는 GAN 모델을 사용하여 데이터를 증강하는 방법을 구현하며, 이 과정에서 DCGAN과 CGAN을 적용한다.

CAE와 비교했을 때, GAN은 기존 데이터와 전혀 다른 새로운 데이터를 생성할 수 있다는 점에서 본 연구에서는 GAN을 활용하였다.

GAN은 생성자(Generator)와 판별자(Discriminator)라는 두 개의 신경망이 상호 경쟁적으로 학습하는 생성 모델로, 생성자는 실제 데이터와 유사한 가짜 데이터를 생성하며, 판별자는 실제 데이터와 가짜 데이터를 구별하는 역할을 한다.

3-1) DCGAN

DCGAN은 CNN의 특징 추출 능력을 활용하여 더욱 정교한 이미지를 생성한다. Convolutional layers의 활용으로 인해 DCGAN은 고해상도의 정교한 이미지를 생성할 수 있다. 하지만 DCGAN은 이미지의 세부 사항이나 속성을 제어할 수 없다.

DCGAN을 통해 데이터를 증강시켰다. 하지만 Fig 5에는 Fig4처럼 불량 패턴을 파악할 수 없었다. Fig5는 출력이 0, 1사이값을 가지도록 만들어졌지만 원본 데이터는 [0, 1, 2]로 이루어져 있었다. 코드를 수정하여 Fig6을 얻을 수 있었지만 loss가 커져 지나치게 단순화되는 오류가 발생했다.

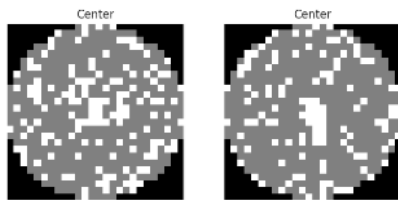


Fig 4. 원본 데이터의 center 유형



Fig 5. DCGAN으로 생성한 center 유형

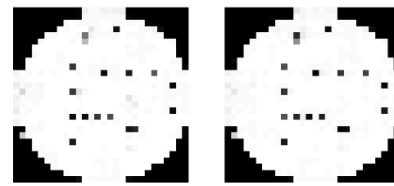


Fig 6. DCGAN으로 생성한 center 유형2

3-2) CGAN

CGAN(Conditional Generative Adversarial Network)은 GAN에 조건(Condition)을 추가하여 특정 조건에 맞는 데이터를 생성할 수 있도록 확장된 모델이다.

응용가능성 측면에서 DCGAN은 일반적인 이미지 생성에 적합하며, CGAN은 특정 조건에 맞는 이미지나 데이터를 생성하는데 적합하다. 따라서 DCGAN은 고품질의 이미지를 생성하는데 강점이 있고, CGAN은 제어 가능한 이미지 생성에 강점이 있다. DCGAN의 문제를 해결하기 위해 CGAN 모델을 사용하였다.

WM-811K 데이터셋은 총 9개의 결함 라벨을 포함하고 있으며, 기본적인 GAN을 활용할 경우, 각 라벨마다 별도의 생성자와 판별자를 학습해야 하는 번거로움이 있다. 이를 해결하기 위해 본 연구에서는 노이즈와 one-hot encoding으로 표현된 라벨을 결합하여 하나의 생성자와 판별자 쌍으로 모든 라벨의 데이터를 증강하는 방식을 채택하였다.

Keras 코드 예제를 참고하여 학습한 결과, 생성자와 판별자의 손실이 불균형하게 나타났으며, 판별자가 과도하게 학습되는 현상이 발생하였다. 이를 개선하기 위해 판별자의 구조를 단순화하고, 생성자의 레이어를 수정하여 정교화하였다. 그 결과, 상대적으로 정교화되었지만 Donut 및 Random과 같은 복잡한 패턴의 결함을 생성하는 데 어려움이 있었으며, 모드 붕괴 현상이 여전히 발생하였다. 이러한 문제의 원인으로는 웨이퍼 데이터의 1bit 이미지 구조로 인한 복잡한 패턴 표현의 어려움과 CGAN 생성자가 선형 연산을 주로 수행하여 복잡한 비선형 패턴을 생성하기 어려운 점이 지적될 수 있다.

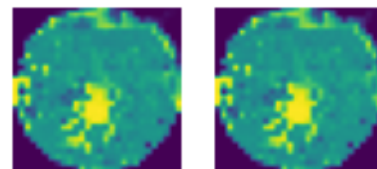


Fig 7. DCGAN으로 생성한 center 유형

생성자와 판별자의 레이어 수정을 통해 상대적인 학습 난이도 차이를 조정하였으나, 모드 붕괴를 완전히

부록 1) 반도체 불량유형에 따른 원인

1) Center 패턴	5) Scratch 패턴
원인: Center 패턴의 결함은 웨이퍼 박막 표면에 발생한 요철이나 굴곡을 화학적 혹은 기계적으로 평탄화하는 CMP(Chemical Mechanical Polishing) 공정에서 발생할 수 있다. CMP 공정에서의 불균일한 압력이나 연마 패드의 마모 상태가 결함을 유발할 수 있다.	원인: Scratch 패턴은 주로 웨이퍼 표면에 물리적 스크래치가 발생한 것으로, 핸들링 과정이나 물리적 접촉으로 인한 손상으로 인해 발생할 수 있다. 이는 주로 웨이퍼 이송 장비의 문제 또는 공정 중의 이물질로 인해 발생한다.
2) Edge-Loc 패턴	6) Random 패턴
원인: Edge-Loc 패턴은 웨이퍼 가장자리에 위치한 결함으로, 주로 에칭(etching) 공정 또는 웨이퍼 핸들링 과정에서의 문제로 인해 발생한다. 웨이퍼의 가장자리가 제대로 처리되지 않거나, 에칭 과정에서의 불균일한 노출로 인해 결함이 생길 수 있다.	원인: Random 패턴은 특정한 형태 없이 불규칙하게 나타나는 결함으로, 공정 중의 환경적 변화나, 장비의 간헐적 문제, 또는 프로세스의 예기치 않은 변동으로 인해 발생할 수 있다. 이러한 결함은 원인을 추적하기 어려운 경우가 많다.
3) Edge-Ring 패턴	7) Donut 패턴
원인: Edge-Ring 패턴은 웨이퍼 가장자리를 따라 고리 형태로 나타나는 결함으로, 주로 에칭 공정에서 발생하는 문제로 인해 나타날 수 있다. 공정 조건의 불균일성이나, 웨이퍼 가장자리에서의 공정 가스 분포의 불균형이 원인이 될 수 있다.	원인: Donut 패턴은 원형 결함으로 주로 웨이퍼 표면의 박막이 불균일하게 증착되거나, 화학적 처리 과정에서 불균일하게 반응한 결과로 나타날 수 있다. 이는 증착 공정에서의 가스 분포 불균형이나, 웨이퍼 표면의 전처리 상태에 기인할 수 있다.
4) Loc 패턴	8) Near-Full 패턴
원인: Loc 패턴은 특정 위치에서 집중적으로 발생하는 결함으로, 제조 공정 중 특정 공정 단계에서의 문제가 원인이 될 수 있다. 예를 들어, 포토 리소그래피 공정에서의 불균일한 노출 또는 현상 공정에서의 잔여물 등이 해당 위치에서 결함을 유발할 수 있다.	원인: Near-Full 패턴은 웨이퍼의 거의 모든 영역에 걸쳐 발생하는 결함으로, 주로 전체적인 공정 조건의 불량, 예를 들어 노광 장비의 오류나, 화학적 기계적 처리 공정에서의 전반적인 실패가 원인일 수 있다.