

# Deep Learning Enabled Computer vision in remanufacturing And refurbishment applications : defect detection and grading for smart phones

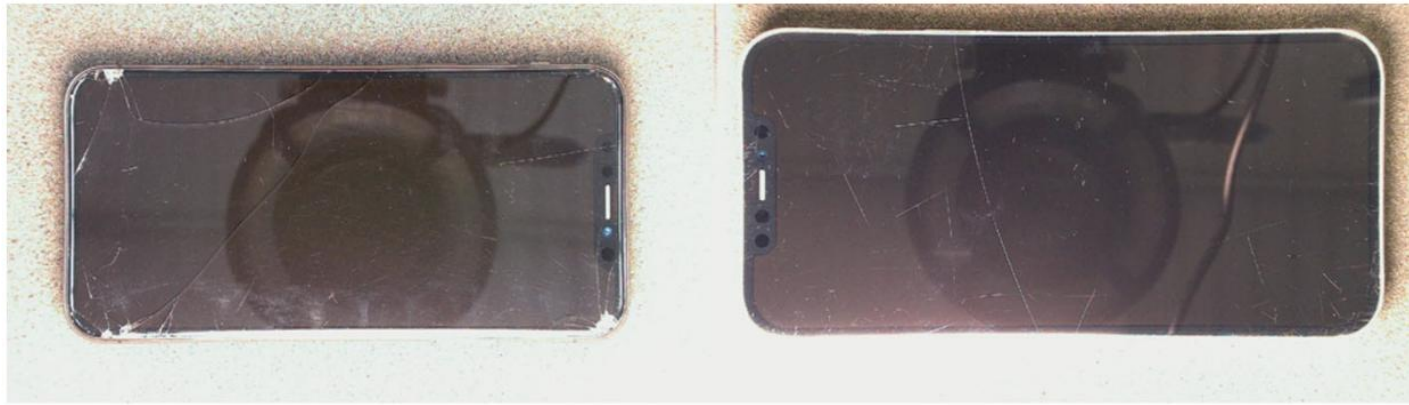
---

26.02.20

임제원



- ✓ 최근 전자폐기물이 빠르게 증가하는 상황(스마트폰, 아이패드, 태블릿 등)
- ✓ 리퍼비싱: 전자기기의 수명을 연장하고 전자 폐기물을 줄이기 위해 사용되는 절차
- ✓ 아래의 사진과 같은 스마트폰은 대부분의 기능을 복원하고 중고 사용자를 위해 리퍼비싱 될 수 있음
- ✓ 이렇게 회수된 제품들은 품질 검사를 거쳐 기능 부품을 확보하기 위해 분해할지 혹은 리퍼비싱을 거칠지 결정
- ✓ 이러한 분류 과정은 거의 작업자가 처리하며, 의사 결정 과정에서 주관이 개입될 가능성을 높임
- ✓ 분류 및 결함 식별의 일관성을 유지하기 위해서는 장기간의 수동 관찰이 필요하며 이러한 특성은 작업자에게 스트레스를 더함
- ✓ 본 논문에서는 탐지된 결함에 따라 스마트폰 재작업을 위해 컴퓨터 비전 기반 SI를 활용하여 체계적으로 결함을 식별하고 등급을 매김



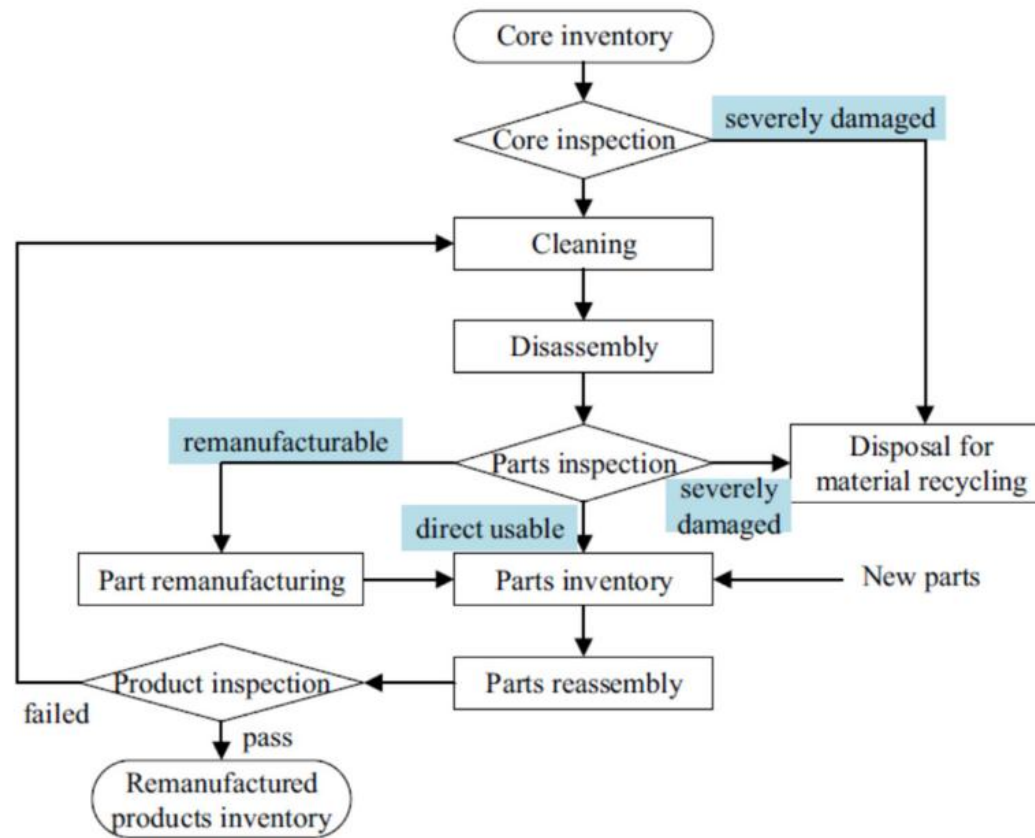
(a)

균일이 있는 스마트폰

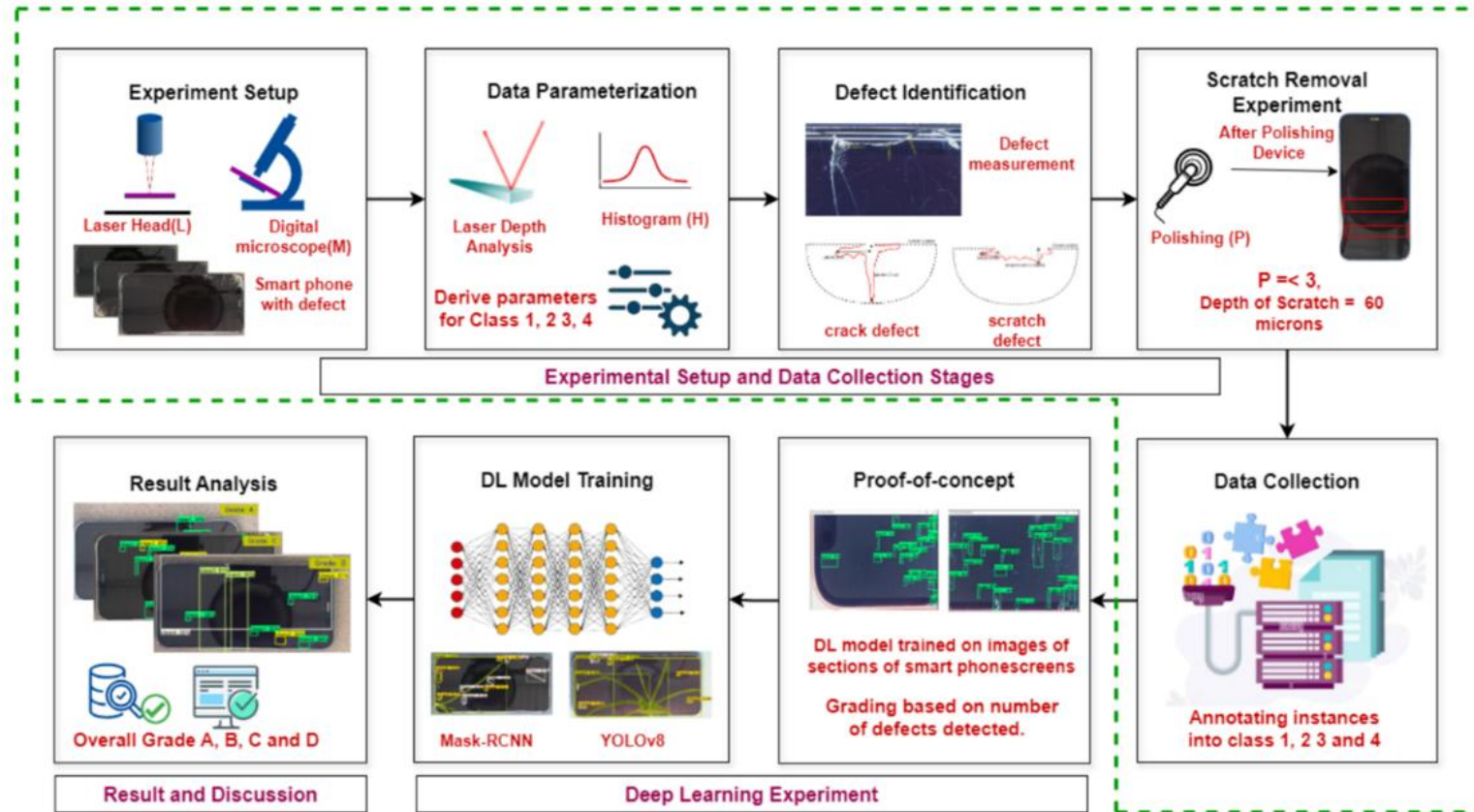
(b)

스크래치가 있는 스마트폰

- ✓ 본 실험에서 스마트폰에서 식별된 결함은 전면 스크린의 균열 및 스크래치로 제한됨
- ✓ 하지만 이는 다른 결함 유형으로 확장될 수 있다는 것은 분명하다고 주장
- ✓ 이러한 새로운 방식은 다음과 같은 목표를 달성
  - 1. 인간 작업자 의존성 및 변동성을 제거/감소시켜 등급 분류 프로세스에 통일성을 부여
  - 2. 등급 분류 프로세스의 불일치로 인한 고객 불만족으로 인한 반품 감소 (우리 과제 현황에서는 반품률이 12%라고 언급됨)
  - 3. 등급 분류 및 재작업 의사 결정 프로세스를 가속화하여 다음 단계 작업의 처리시간 단축
  - 4. 수집된 큐레이션된 데이터셋에서 딥러닝 모델의 성능을 평가
- ✓ 본 연구는 아래와 같은 기여를 함
  - 1. 스마트폰 화면의 긁힘 및 균열과 같은 결함 유형을 결정하기 위한 실험적 방법론 세트 개발
  - 2. 반품된 스마트폰 등급 판정을 위한 딥러닝 모델 훈련 및 데이터 수집  
(저자들이 아는 한 본 연구의 데이터셋은 결함의 정도에 기반하여 긁힘 및 균열을 분류한 유일한 데이터셋)
  - 3. 새로운 데이터셋을 기반으로 수집된 데이터셋으로 훈련된 객체 탐지 및 분할을 위한 딥러닝 모델 세트 개발하여 베이스라인 제공
  - 4. 해당 시장 표준에 따라 스마트폰을 A,B,C,D 등급으로 분류하기 위한 제안된 기준을 도입



전체 프레임워크 : 재고 투입부터 핵심 검사, 세척, 분해 등 스마트폰의 재제조 과정을 보여줌  
본 문헌에서 제안한 모델은 핵심 검사인 Core inspection에 해당하여 제품의 등급을 평가

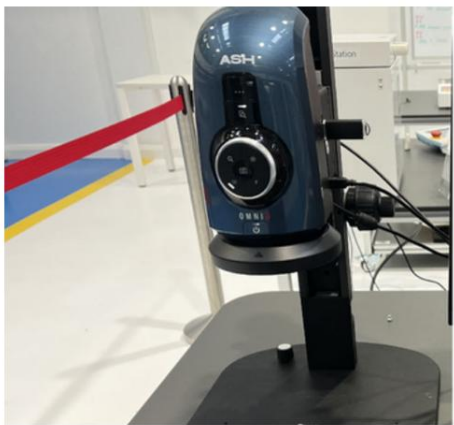


**Fig. 3 Development process followed in this experiment to arrive at the feasible solution.** Each stage of the development process is addressed in the following sections

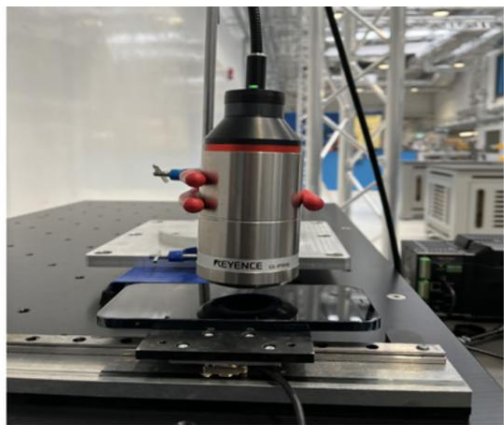


## ❖ Experimental setup for data parameterization

- ✓ 본 연구에서는 수집된 데이터셋을 Mask R-CNN과 YOLOv8을 사용하여 훈련
- ✓ 이미지 데이터 수집
  - ✓ 이미지 데이터 샘플은 디지털 현미경(a)을 사용하여 수집됨 (이미지 반사를 줄이기 위해 확산 조명을 사용했지만, 완전 제거 x)
- ✓ 데이터 매개변수화
  - ✓ 스마트폰 화면의 스크래치 범위를 결정하기 위해 스팟형 레이저 헤드(b)를 사용하여 수집
  - ✓ 이 장비는 주로 스마트폰 화면의 스크래치 깊이 범위를 이해하는 데 사용됨
  - ✓ 레이저 헤드를 사용한 깊이 측정은 유리 두께를 넘는 깊은 균열은 측정이 불가능 하여  
결함의 깊이 측면은 제거하고, 스크래치의 깊이 측면만 측정함 (균열이 있을 경우 무조건 D등급을 주며, 스크래치의 개수와 깊이 사용)



(a)



(b)



(a)



(b)

## ❖ Data parameterization

- ✓ 외관상 결함을 스마트폰의 등급으로 매핑하기 위한 파라미터화 필요
- ✓ 단순히 측정된 스크래치의 깊이를 선형 파라미터로 사용하는 것은 폭, 빛 반사 등 여러 문제로 인해 정확하게 나타낼 수 없음
- ✓ 저자는 이를 해결하기 위해 머신러닝 관행(사전 정의된 분류 체계)에 기반하여 파라미터화를 진행하였다고 설명
- ✓ 주변 조명 하에서 결함 영역의 강도는 결정된 네 가지 클래스를 구별하는 기준으로 고려함 (클래스 설명은 후술)

**Table 5** Criteria for grading based on number of defect class detections

Number of Detection for each class				Predicted Grade
Class 4 >0				Grade D
Class 4 = 0	Class 3 >0			Grade D
Class 4 = 0	Class 3 =0	Class 2 >0		Grade C
Class 4 = 0	Class 3 =0	Class 2 = 0	Class 1 >= 10	Grade B
Class 4 = 0	Class 3 =0	Class 2 = 0	Class 1 < 10	Grade A

등급을 매기는 기준

## ❖ Defect identification

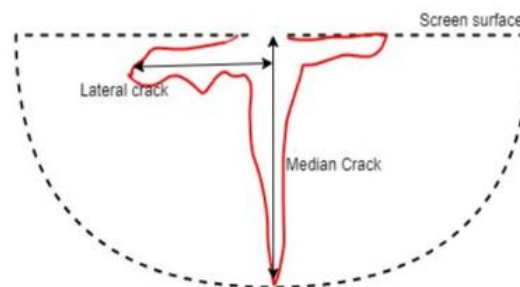
- ✓ 스마트폰 화면에서 식별된 두 가지 결함은 '균열'과 '스크래치'
- ✓ 균열: 연마에 부적합하여 교체 또는 폐기처리 진행
- ✓ 스크래치: 연마로 제거하여 중고시장에 판매 가능
- ✓ (a): 균열 | (b): 스크래치



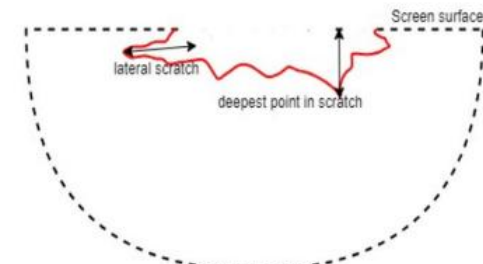
(a)



(b)



(a)



(b)



## ❖ Scratch removal experiment

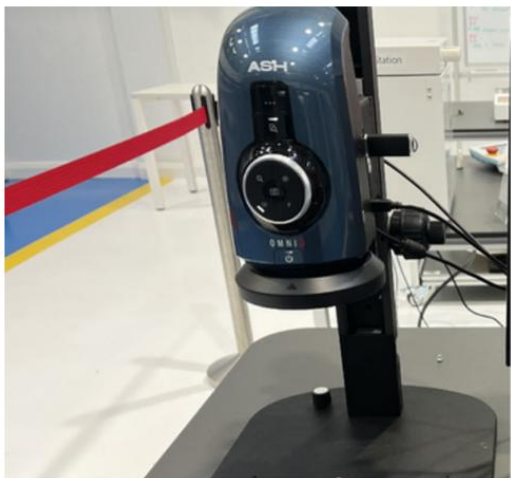
- ✓ 스마트폰 화면 스크래치 깊이가 등급 개선을 위해 필요한 연마 작업 횟수를 결정하는 요인임을 확인  
(폴리싱: 미세한 흠집이나 스크래치를 제거하여 외관 품질을 개선하는 재작업 공정)
- ✓ 2차원 이미지로는 스크래치의 깊이를 정확히 측정하기 어렵기 때문에 레이저 측정 장비를 사용하여 스크래치 깊이 측정
- ✓ 표준 폴리싱 주기가 스크래치 깊이에 미치는 영향을 파악하고, 스크래치 제거에 필요한 폴리싱 횟수를 결정하는 데 필요한 데이터 수집
- ✓ Table2는 폴리싱 횟수에 따른 처리 가능한 깊이를 표현 (폴리싱을 1번 하면 20micron의 깊이까지 제거 가능)

**Table 2** Table showing depth of scratches (in microns) removed per standard cycle of polishing

Number of polishing cycles	Depth of scratch removed
1	all up to 20 microns
2	all up to 40 microns
3	all up to 60 microns

## ❖ Data Collection

- ✓ 이 논문에서 데이터 수집은 딥러닝 실험의 핵심 단계 ( 그동안 활용 가능한 데이터가 없었기 때문 )
- ✓ 초기 단계는 제한된 수의 스마트폰 샘플로 수행되었으며, 비반사 조명에서의 추가적인 문제는 데이터 수집 과정에 영향을 미침 (아래 그림 (a)에 나타난 현미경을 사용한 이미지 데이터 수집, 이러한 빛반사로 인해 데이터 수집이 어려움)
- ✓ 1단계 ( Version 1 ) : 개념 증명(Proof-of-concept)을 위해 제한된 수의 스마트폰(24대)를 사용하여, 화면의 1/6 부분만을 캡처한 이미지를 수집. 이 데이터셋은 단일클래스(결함 있음)로 구성되었으며, 총 106개의 결함 샘플 이미지가 포함됨
- ✓ 2단계 ( Version 2 ) : 더 정확한 등급 분류를 위해 전체 스마트폰 화면을 캡처하는 방식으로 데이터를 수집. 이 단계에서는 115대의 스마트폰에서 총 246개의 결함 샘플 이미지를 확보했으며, 결함의 종류(스크래치, 균열)와 심각도에 따라 다중 클래스로 분류 진행 학습, 테스트, 검증 ( 70 : 15 : 15 ) 비율로 분할되었음



(a)

**Table 3** Descriptions of the two datasets

Dataset	Description	# Phones	# Images	# Samples	Resolution
Version 1	Proof of concept dataset, single class (defective), one-sixth of screen surface capture per image	24	147	106	1920 × 1080
Version 2	Multi-class, full screen captured	115	289	246	1920 × 1080

## ❖ Data Collection

- ✓ 데이터 주석 ( Annotation ) : 수집된 이미지는 Labelme 소프트웨어를 사용하여 폴리곤 방식으로 주석처리됨.

데이터 주석 : 어떤 부분이 결함인지 AI 모델에게 알려주는 것

폴리곤 : 결함 영역이 어떤 모양인지 정확하게 표시하기 위해 사용하는 방법 ( 도형으로 표시 )

- ✓ 결함은 Table 4에 정의된 4가지 클래스로 분류되었으며, 이러한 정보는 딥러닝 모델 훈련에 필요한 형식으로 변환됨

( 자세한 변환은 사용 모델에 따라 모두 상이함 , YOLOv8의 경우 .txt 파일을 요구 )

- ✓ 저자들은 자체적으로 구축한 데이터셋이 화면 스크래치와 균열을 결함 심각도별로 분류한 유일한 데이터셋이라고 강조함

- Class1: 미세한 스크래치 (낮은 광도를 띠며, 연마 작업 (polishing) 1회로 제거 가능한 수준)
- Class2: 더 깊고 긴 스크래치 (높은 광도를 띠며, class1보다는 심각하지만 제거 가능성이 있음)
- Class3: 잔금 (미세한 균열로, 제거하기 어려운 수준)
- Class4: 주요 균열 또는 파손 (화면 전체를 가로지르는 심각한 균열이나 깨진 부분까지 포함한 가장 심각한 결함)

**Table 4** Classifications based on the severity of presented defects: Stage 2

Defect	Defect Description	Class
Scratch	Minor, removable scratch(low light intensity-grey colour)	Class 1
	Deeper and longer scratch lines(higher intensity-bright regions)	Class 2
Crack	Hairline crack(difficult)	Class 3
	Major crack lines, including shattered edges	Class 4

## ❖ Data Collection

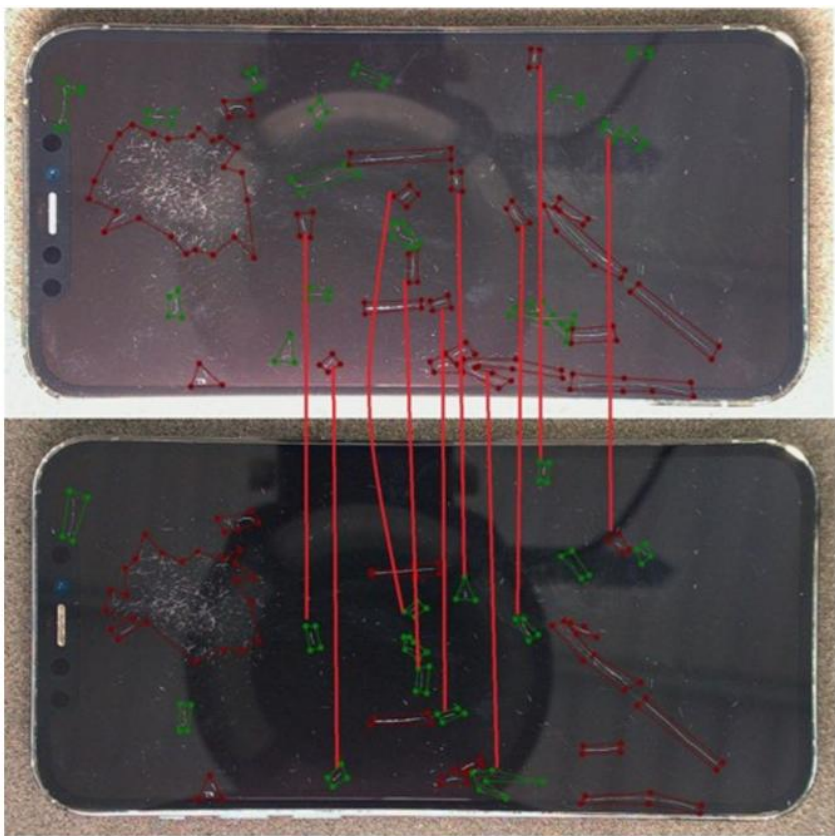
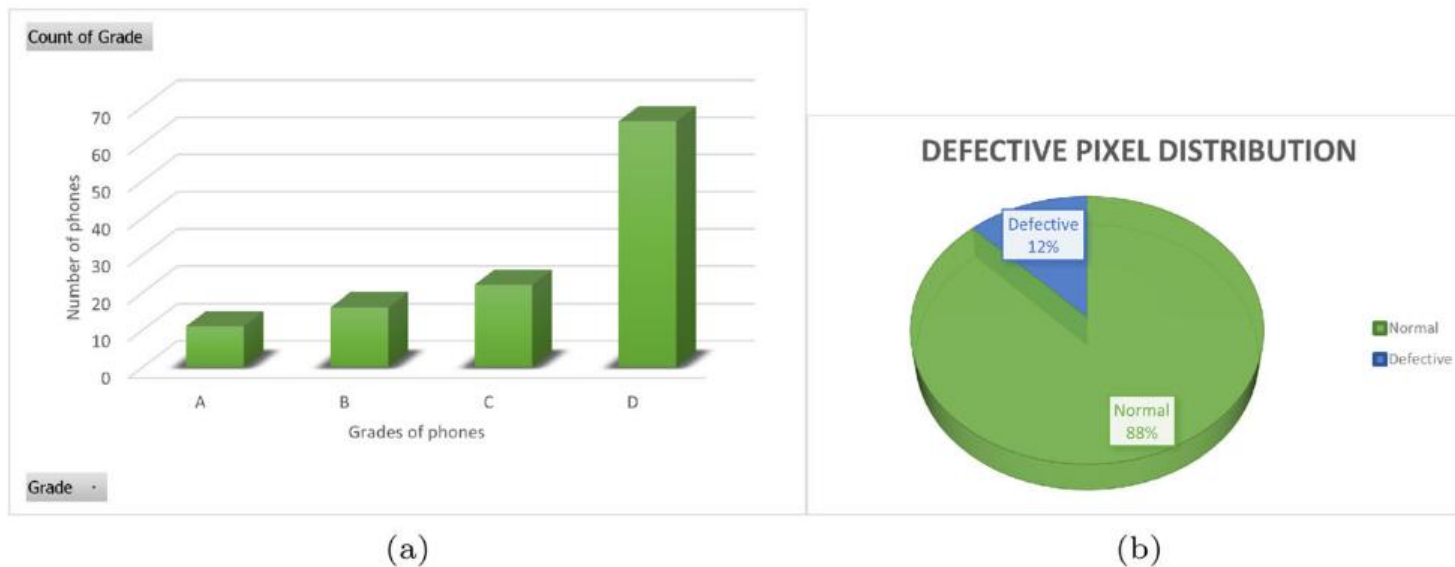


Fig. 7 Illustration of importance in uniform illumination in data capture for training and detection. The red lines connect scratches labeled as two different classes due to changes in illumination. The lower image exhibits poor lighting and higher levels of reflection, leading to mislabeling of the samples

- ✓ 데이터 주석 에서 폴리곤 방식에서 데이터 캡처 균일 조명의 중요성
- ✓ 상단 이미지  
: 스크래치와 균열을 나타내는 다양한 모양(초록색 삼각형, 빨간색 점과 선)으로 주석 표시  
녹색 - 스크래치 / 빨간색 - 균열
- ✓ 하단 이미지  
: 상단 이미지와 동일한 스마트폰이지만, 조명으로 인한 차이가 보임
- ✓ 하단 이미지는 낮은 빛과 높은 반사율로 인한 라벨링 오류를 나타냄

## ❖ Data Collection



**Fig. 8 Dataset distributions.** (a) shows the number of phones vs. grade distribution used for data collection and Figure(b) shows the distribution of high-intensity pixels representing defective areas

- ✓ (a) : 데이터 수집에 사용된 휴대폰의 등급 분포 시각화  
등급A 15대 / 등급 B 20대 / 등급 C 28대 / 등급 D 71대
- ✓ (b) : 불량 픽셀 분포로, 이미지 내에서 정상 픽셀과 불량 픽셀의 비율  
정상 88% / 불량 12% → 불량 픽셀이 상대적으로 적은 비율을 차지



# III Deep learning experiment

## ❖ Proof-of-concept experiment

- ✓ 앞서 언급한 데이터셋 Version1을 사용하여 진행
- ✓ 적은 스마트폰 수와 1/6의 화면에도 불구하고, 이 실험에서는 높은 정확도가 탐지됨  
(mIoU : 0.8314 / 정밀도 : 0.8 / 정확도 : 0.8)

mIoU : mean Intersection over Union의 약자로 예측 영역과 실제 영역의 비율 | 결함 영역의 테두리를 얼마나 정확하게 감지했는가

정밀도 : 모델이 결함이라고 판단한 영역 중에서 실제로 결함이 맞을 확률

정확도 : 모델이 화면 전체를 보고 총 몇 번의 예측을 맞췄는지

- ✓ 우측의 그림 9는 과도한 스크래치가 있는 휴대폰 화면으로, 범위 별 굵힘 정도를 수치로 표현한것을 확인 가능
- ✓ 그림 10은 1/6 화면에서의 4개 등급에 대한 결함 감지

Table 6 Criteria for grading based on number of scratches (single defect class)

Number of detections	Predicted Grade
count <10	Grade A
count >=10 and <30	Grade B
count >=30	Grade C

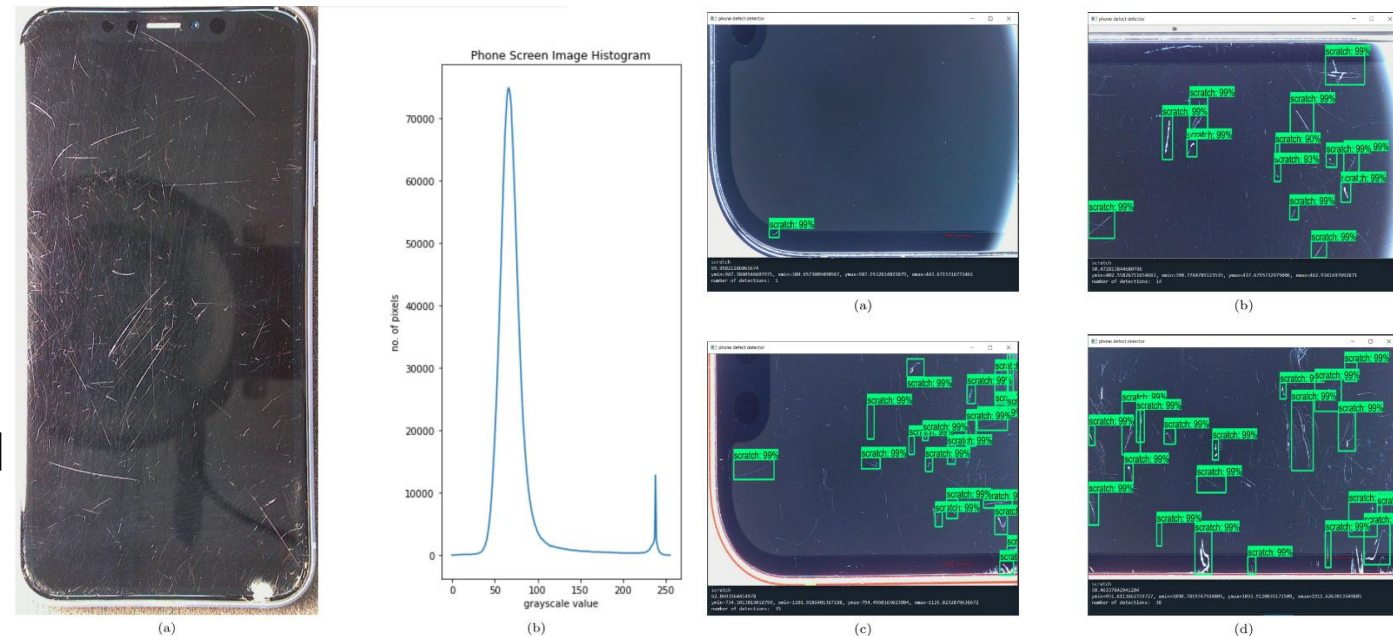


Fig. 9 Intensity variations in an image due to scratches. (a) represents a smartphone with excessive scratches, and (b) is the histogram of the image showing pixel intensity variations in the phone screen image. Fig. 10 Different grades of phone damage. (a) is Grade 1, (b) is Grade 2, and (c) and (d) are Grade 3 as per the proof-of-concept experiment



## ❖ DL model training

- ✓ Proof-of-concept experiment 단계에서는 TensorFlow 기반의 Mask R-CNN을 사용
- ✓ 논문에서 사용된 딥러닝 모델 중 하나로, 객체 감지 및 인스턴스 분할에 효과적으로 알려짐
- ✓ 그 외 YOLOv7n, YOLOv8n, YOLOv8x, YOLO11x에 대해서도 학습을 진행 (Version 2 데이터셋)

## ❖ Mask R-CNN

- ✓ 탐지에 대한 정확도는 낮은 편이며, Version 2 데이터셋에서는 베이스라인 모델로 사용됨
- ✓ 하단의 Table 7에서 Masck-RCNN의 성능을 확인 가능 (Recall : 재현율)

**Table 7** Table detailing the accuracy achieved in each stage of phone grading experiment against the number of image data collected Mask R-CNN

#	no. of images	IoU	Precision	Recall
Version 1	106	83.14	0.8	0.8
Version 2	246	73.43	0.7	0.68

# III Deep learning experiment

**Table 8** Baseline results for Segmentation on YOLOv7, YOLOv8n, YOLOv8x, YOLO11n and YOLO11x deep learning models

Model	Class	Box(P)	Box (R)	Box (mAP50)	Mask(P)	Mask (R)	Mask (mAP50)
YOLOv7	all	0.435	0.4	0.389	0.341	0.326	0.271
	class1	0.288	0.247	0.172	0.297	0.266	0.17
	class2	0.441	0.392	0.369	0.356	0.327	0.282
	class3	0.483	0.417	0.471	0.191	0.167	0.0714
	class4	0.527	<b>0.545</b>	<b>0.545</b>	0.522	0.545	<b>0.559</b>
YOLOv8n	all	0.45	0.315	0.297	0.368	0.23	0.214
	class1	0.346	0.181	0.202	0.346	0.181	0.185
	class2	0.51	0.297	0.328	0.502	0.293	0.312
	class3	0.398	0.417	0.296	0.0796	0.0833	0.0178
	class4	0.546	0.364	<b>0.362</b>	<b>0.546</b>	0.364	<b>0.344</b>
YOLOv8x	all	0.539	0.295	0.315	0.331	0.216	0.204
	class1	0.367	0.132	0.169	0.321	0.128	0.137
	class2	0.526	0.343	0.359	0.392	0.281	0.262
	class3	0.62	0.25	0.323	0	0	0.00868
	class4	<b>0.642</b>	0.455	0.41	<b>0.608</b>	0.455	0.41
YOLO11n	all	0.399	0.312	0.325	0.317	0.25	0.246
	class1	0.409	0.207	0.219	0.37	0.213	0.204
	class2	0.545	0.343	0.378	0.489	0.34	0.358
	class3	0.297	0.333	0.305	0.0662	0.0833	0.0301
	class4	0.347	0.364	0.398	0.341	0.364	0.393
YOLO11x	all	0.357	0.38	0.358	0.281	0.266	0.226
	class1	0.415	0.236	0.235	0.384	0.197	0.193
	class2	0.481	0.415	0.386	0.404	0.33	0.302
	class3	0.26	0.417	0.329	0.0578	0.0833	0.0196
	class4	0.271	0.455	<b>0.481</b>	0.278	0.455	<b>0.388</b>
YOLO8x	all	0.501	0.334	0.355	0.356	0.234	0.228
	class1	0.378	0.197	0.186	0.321	0.178	0.123
	class2	0.522	0.359	0.411	0.445	0.32	0.274
with Aug	class3	0.401	0.417	0.354	0.158	0.167	0.0707
	class4	<b>0.704</b>	0.364	0.47	0.5	0.273	0.443

Bold entries are significant results in the experiment

## ❖ YOLO Model Result Table

- ✓ Class : 결함 정도
- ✓ Box : 객체 감지 작업에 대한 성능 지표로, 결함을 얼마나 잘 탐지하는가  
→ 결함이 어디에 있는지
- ✓ Mask : 객체 분할 작업에 대한 성능 지표로,  
정확한 영역을 얼마나 잘 분할하는가  
→ 결함이 어떤 모양인지

P : Precision, 탐지된 객체 중 실제인 비율

R : Recall, 실제 객체 중 탐지에 성공한 비율

mAP50 : 다양한 IoU 임계값으로, 전반적인 성능을 나타냄

- ✓ 각 값은 높을수록 좋음

# III Deep learning experiment

## ❖ Confusion Matrix

- ✓ YOLOv8의 Confusion Matrix
  - ✓ Class 1,2는 잘 분류하지만 3,4에서는 약한 모습
  - ✓ 데이터셋 인스턴스가 적기도 하며 (그림 14), 배경 영역을 결함으로 잘못 분류하는 경향이 짙음
  - ✓ 그림 14의 b는 정밀도-재현율 곡선으로, 곡선 아래의 면적이 클수록 좋다고 평가함
- 이 그림은 데이터셋의 클래스 분포가 고르지 않다는 점과, 이러한 불균형이 성능에 미치는 영향을 시각화

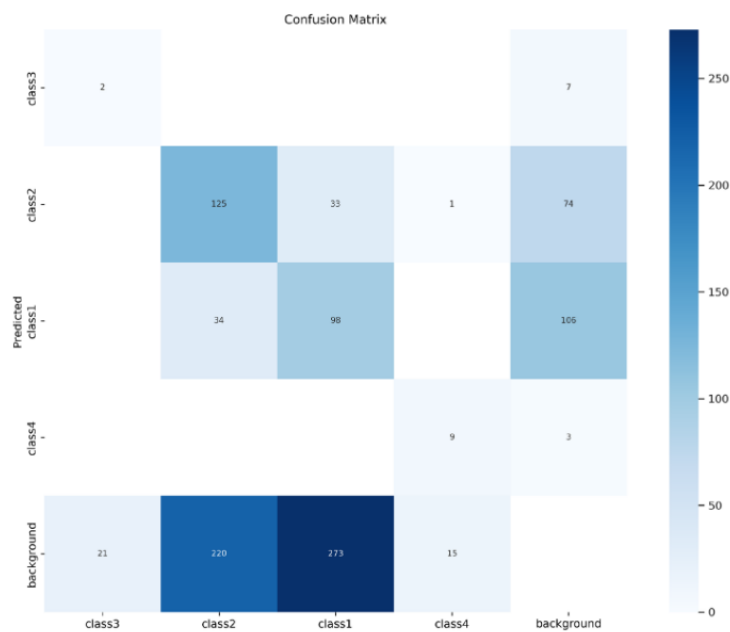


Fig. 12 Confusion matrix from detection using YOLOv8. It is interesting to note that classes 1 and 2 have higher prediction accuracy than classes 3 and 4. Classes 1 and 2 have higher occurrences yet are more prone to subjectivity in grading by humans. Precise detection using computer vision eliminates subjectivity and highly improves consistency in grading, thereby reducing the returns due to customer dissatisfaction

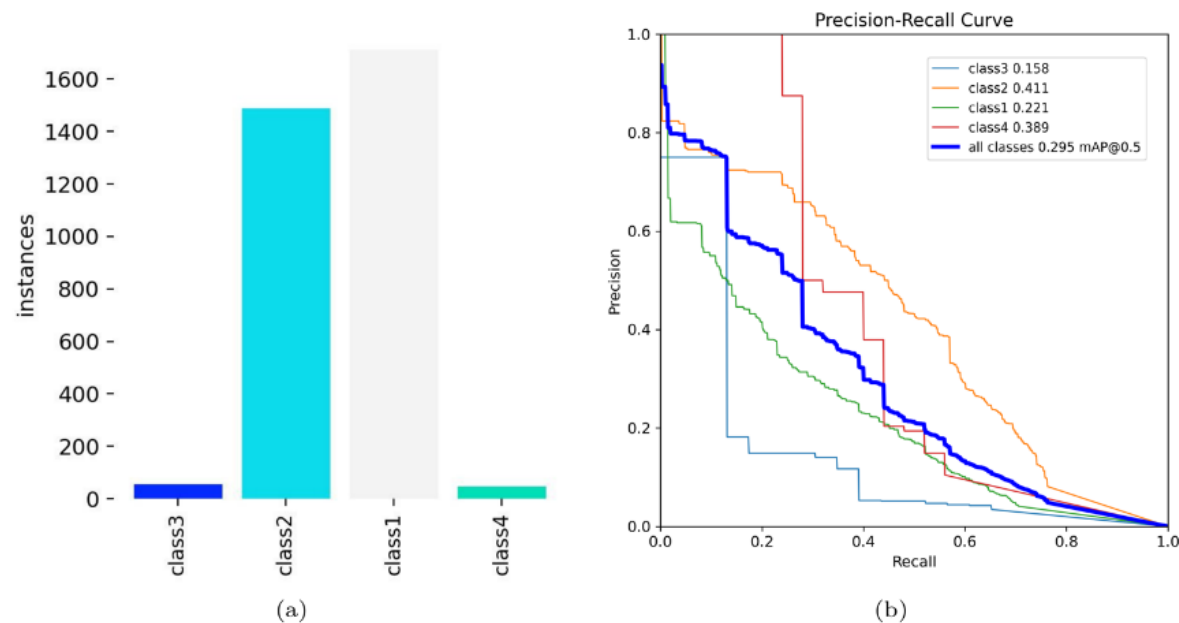


Fig. 14 Class imbalance in the dataset. (a) shows the number of instances under each label, and (b) shows the Precision-Recall curve from detection results, with the lower precision and recall for Class 3 possibly due to the class imbalance

# III Deep learning experiment

## ❖ Pre-fine-tuning using the Roboflow cracks dataset

- ✓ YOLO11을 기반으로 학습하기 이전에 Roboflow의 균열 분할 데이터셋으로 YOLO 모델을 사전 파인튜닝 진행
- ✓ 이후, 자체 스마트폰 결함 탐지 데이터셋으로 학습
- ✓ Roboflow의 균열 분할 데이터셋으로 사전 파인튜닝을 하고 학습하는것이, 그냥 학습하는 것보다 더 좋은 성능을 나타냄
- ✓ Table 9는 결함 탐지(결함이 어디 있는지), Table 10은 결함 분할(결함이 어떤 모양인지)에 대한 각각의 성능 지표

**Table 9** Table showing YOLO11n detection results on phone defect dataset, with and without pre-fine-tuning using the Roboflow cracks segmentation dataset

Class	Instances	Box(P)	Box(R)	Box(mAP50)	Box(mAP50-95)
all	633	0.409	0.345	0.355	<b>0.128</b>
class1	304	0.435	0.23	0.237	0.0799
class2	306	0.51	0.363	0.385	0.143
class3	12	0.129	0.333	0.307	0.0666
class4	11	0.561	0.455	0.492	0.224
After pre-fine-tuning with Cracks dataset					
all	633	0.358	0.339	0.316	<b>0.133</b>
class1	304	0.284	0.168	0.139	0.0408
class2	306	0.449	0.33	0.309	0.108
class3	12	0.206	<b>0.417</b>	<b>0.394</b>	<b>0.144</b>
class4	11	0.493	<b>0.443</b>	0.424	<b>0.238</b>

Bold entries are significant results in the experiment

**Table 10** Table showing YOLO11n segmentation results on phone defect dataset, with and without pre-fine-tuning using the Roboflow cracks segmentation dataset

Class	Box(P)	Box(R)	Box(mAP50)	Mask(P)	Mask(R)	Mask(mAP50)
all	0.399	0.312	0.325	0.317	0.25	0.246
class1	0.409	0.207	0.219	0.37	0.213	0.204
class2	0.545	0.343	0.378	0.489	0.34	0.358
class3	0.297	0.333	0.305	0.0662	0.0833	0.0301
class4	0.347	0.364	0.398	0.341	0.364	0.393
After pre-fine-tuning with Cracks dataset						
all	0.57	0.333	0.383	0.393	0.285	0.278
class1	0.424	0.164	0.201	0.329	0.247	0.191
class2	<b>0.689</b>	0.206	0.318	<b>0.543</b>	0.265	0.308
class3	0.386	0.417	0.383	0.0376	0.0833	0.0295
class4	<b>0.78</b>	<b>0.545</b>	<b>0.632</b>	<b>0.662</b>	<b>0.545</b>	<b>0.583</b>

There is a huge increase in the Precision, Recall, and mAP50 values for class 4, which represent the long cracks on the phone screen. These crack lines are very similar to the road and wall surface cracks in the Roboflow crack segmentation dataset

Bold entries are significant results in the experiment

## ❖ Conclusion

- ✓ 본 연구는 스마트폰 화면의 스크래치 및 균열과 같은 결함을 자동으로 감지하고 심각도에 따라 등급을 분류하는 딥러닝 모델을 개발함
- ✓ 이러한 자동화된 시스템은 수작업으로 인한 주관성을 줄이고, 일관성 있는 등급 분류를 통해 고객 불만족으로 인한 반품을 감소
- ✓ YOLOv7, YOLOv8, YOLO11, Mask R-CNN과 같은 다양한 딥러닝 모델을 사용하여 성능을 평가했으며, 특히 YOLOv8x 모델은 8ms의 낮은 추론 시간과 70.4%의 높은 정밀도를 달성했습니다.
- ✓ 미세한 균열(Class 3) 및 심각한 균열(Class 4)과 같이 데이터셋에서 상대적으로 적은 비율을 차지하는 결함에 대한 성능 향상을 위해 로드 균열 세분화 데이터셋으로 사전 미세 조정하는 실험을 수행했으며, 이는 Class 4 결함의 정밀도를 크게 향상시키는 결과를 가져왔습니다.

## ❖ Future work

### ✓ 데이터셋 확장 및 개선:

현재 데이터셋의 클래스 불균형 문제를 해결하기 위해 더 많은 고해상도 이미지와 다양한 결함 유형 및 심각도를 포함하는 데이터셋 필요

### ✓ 점진적 학습(Incremental Learning) 도입:

재제조 환경은 새로운 제품 모델과 결함 유형이 지속적으로 추가되므로, 모델을 완전히 재훈련하는 대신 새로운 데이터를 점진적으로 학습할 수 있는 점진적 학습(Incremental Learning) 또는 지속적 학습(Continual Learning) 시스템 도입이 필수

### ✓ 폴리싱 사이클 예측 통합: 스크래치 제거를 위한 폴리싱(polishing) 횟수 예측 기능을 딥러닝 모델에 통합하는 연구가 필요합니다. 이를 위해 폴리싱 시간과 결함 제거 효율성에 대한 추가적인 데이터 수집 및 전문가 인터뷰가 요구됩니다.

### ✓ 새로운 결함 유형 및 스마트폰 모델 확장: 본 연구는 스마트폰 화면의 스크래치와 균열에 집중했지만, 향후 다른 유형의 결함(예: 디스플레이 번인, 내부 부품 결함)이나 다양한 제조사 및 모델의 스마트폰으로 연구를 확장할 수 있습니다.