Multi-view Anomaly Detection

2025 Summer seminar

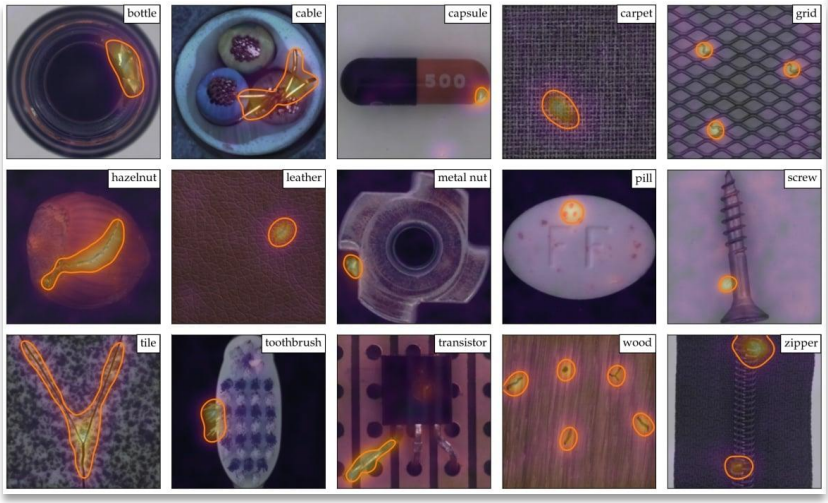
Sogang University
Vision & Display Systems Lab, Dept. of Electronic Engineering

Presented By
김건우

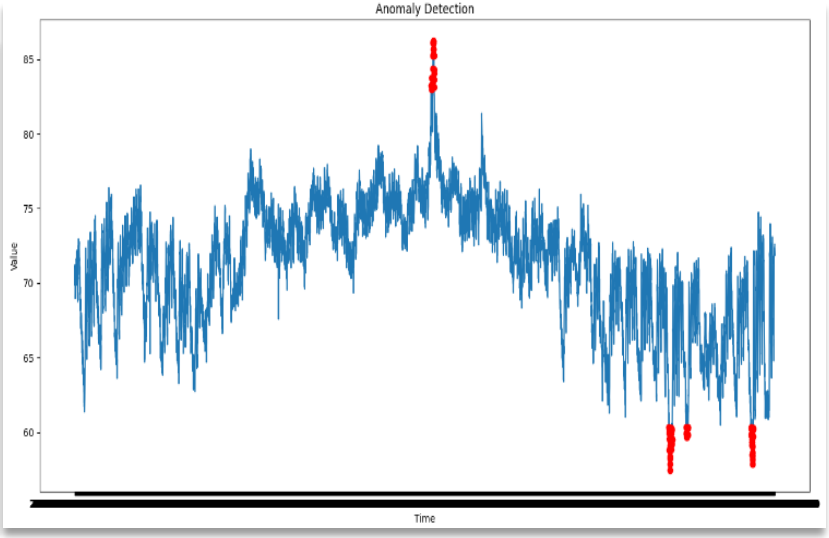
Background

Anomaly Detection(이상 탐지)

- 일반적인(Normal) 패턴에서 벗어난 이상(Anomaly)를 탐지하는 기법
- 활용 분야
  - 산업 양품 검사
  - 의료 영상
  - 보안 및 침입 탐지



< Image AD >

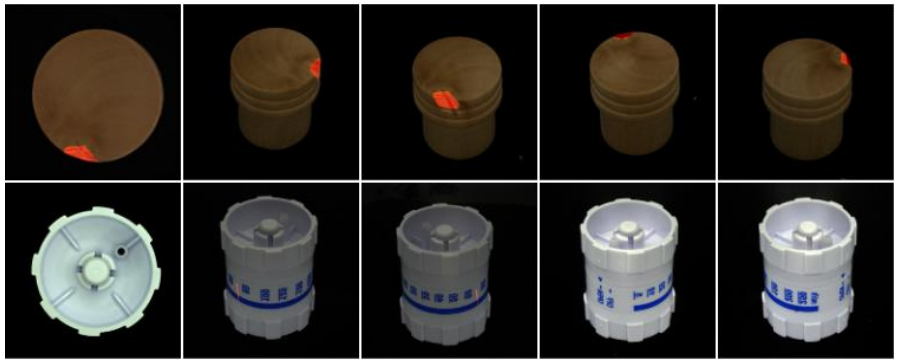


< Time Series AD >

Background

Multi-view Anomaly Detection(이상 탐지)

- 개념
  - 하나의 Sample을 다양한 view에서 수집한 데이터로부터 anomaly를 감지
- 핵심 아이디어
  - Cross-view Consistency
    - 정상이면 모든 view에서 비슷한 feature가 보여야 함
    - Abnormal은 특정 view에서 드러나지 않아 불일치
  - View Complementarity
    - 각기 다른 시점의 정보가 상호 보완적 역할을 함
    - 가려진 영역이나 시점 특이성 문제를 극복할 수 있음



Learning Multi-view Anomaly Detection

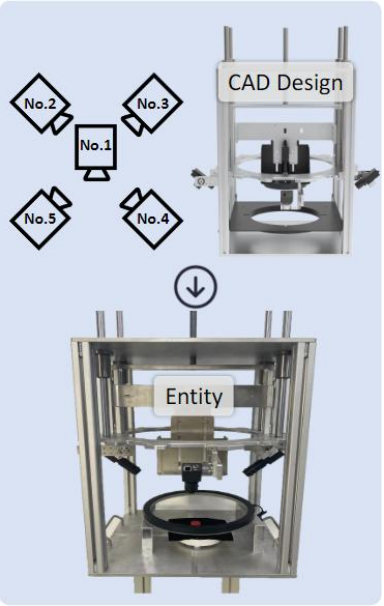
[Arxiv 2024]

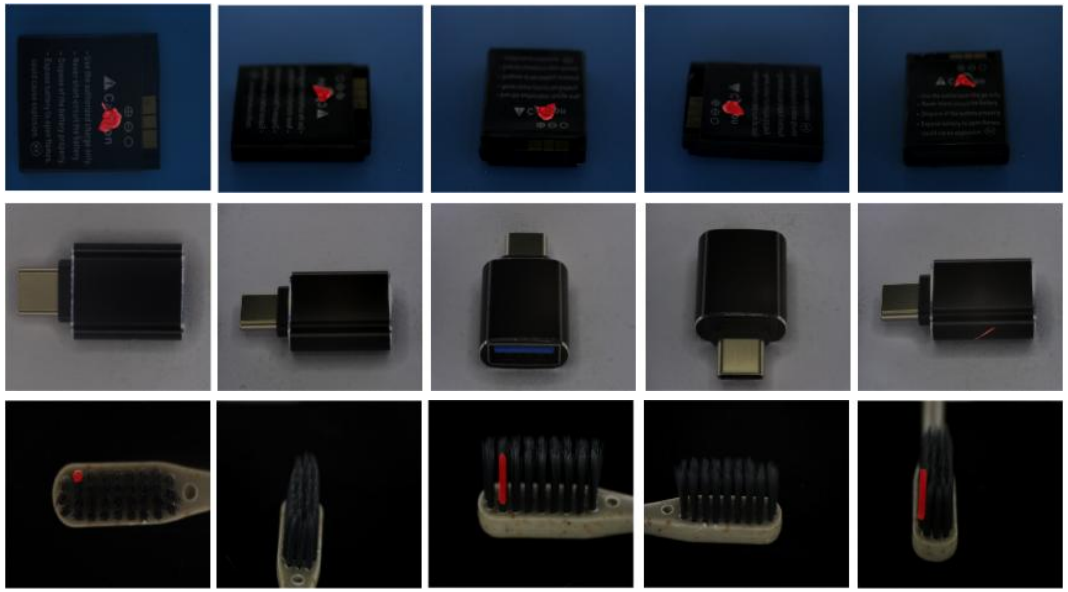
(Submitted TO IEEE TRANSACTIONS ON MULTIMEDIA)

Introduction

Real-IAD Dataset

- Multi-view anomaly detection dataset
- 기존 single view와 달리 multi-view dataset은 객체를 다양한 관점으로 보여줌
  - 어떤 시점에서는 이상이 보이지만 다른 시점에서는 정상처럼 보일 수 있음
    - ⁂ 다중 시점 간 상호 관계





Introduction

현재 2D 기반 이상 탐지 기법

- Data Augmentation method
  - Synthetic anomaly
    - 문제점: 실제 anomaly 대응 한계
- Reconstruction-based method
  - 정상만 복원 → 이상 검출
    - 문제점: 복잡한 뷰 구조 미반영
- Embedding-based method
  - Feature embedding 비교
    - 문제점: 다중 시점 결합 부재
- Fusion techniques와 attention based fusion method
  - CNN과 Attention 기반
    - 문제점: 대부분 2-view만 결합(너무 적음)
    - 시점 간의 위치/ 의미 정렬 안됨

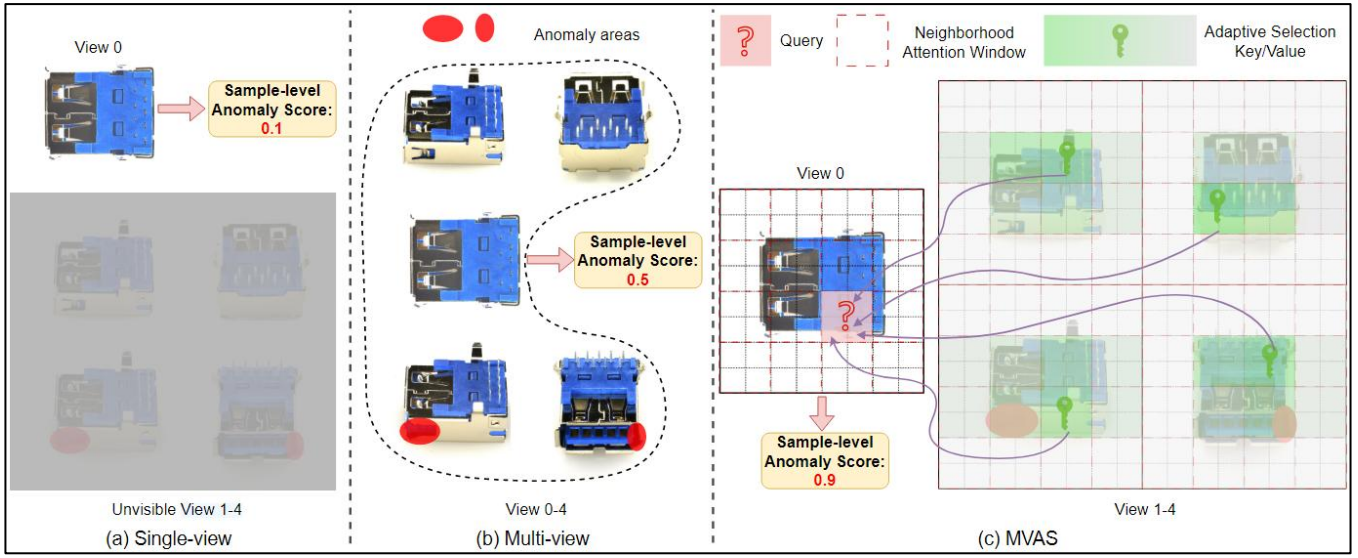
Introduction

Contribution

- 새로운 Multi-view anomaly detection (MVAD) framework 제안
  - 기존 AD연구에서 다루지 않았던 Multi-view learning 문제를 최초로 다룸
  - 각 view의 정보를 결합하여 더 정확한 이상 탐지를 수행할 수 있도록 하는 새로운 설계
- MVAS(Multi-View Adaptive Selection) 알고리즘 도입
  - 단일 시점의 윈도우마다 다중 시점 중에서 의미론적으로 가장 관련된 K개의 윈도우를 적응적으로 선택
  - 선택된 윈도우에 대해서만 attention 연산을 수행
    - 탐지 성능 향상
    - 계산 복잡도는 최소화

Introduction

Multi-View Adaptive Selection(MVAS)



- 입력 이미지의 feature map을 neighbourhood attention window로 분할
- MVAS 알고리즘이 적용
  - 단일 시점의 window와 다중 시점 window들 간의 semantic correlation matrix을 계산
  - 가장 상관도가 높은 top-K개의 다중 시점 window가 Key, Value로 선택
    - ∴ 단일 시점 window(Query)와 해당하는 key/value 간의 인접 상관 cross-attention이 가능
  - Top-K 윈도우만 단일 시점 window와 연산
    - ∴ 윈도우 크기와 top-k 수를 조절해 계산 복잡도를 최소 선형 수준까지 대폭 줄임

Method

Multi-View Adaptive Selection

- Neighborhood Attention Window
  - 전체 feature map을 작은 window 단위로 나누는 단계
    - 이 윈도우 단위에서만 attention을 수행하여 계산량을 줄이고 local 정보 집중
  - Input feature map: $\mathbf{X}_i \in \mathbb{R}^{v \times h \times w \times c}$
  - Window size $a \times a$: $\mathbf{X}_a \in \mathbb{R}^{v \times a^2 \times \frac{hw}{a^2} \times c}$
- Multi-View Windows Adaptive Selection
  - Correlation matrix: $\mathbf{A}_c \in \mathbb{R}^{a^2 \times (v-1)a^2} = \mathbf{A}_s(\mathbf{A}_m)^K$
  - Top K: $\mathbf{I}_m^K = \mathrm{TopK\_Index}(\mathbf{A}_c)$
- Neighbourhood Correlative Cross-Attention
  - Single-View(Query)와 Top-K Window(Key/Value)사이에서만 Attention 수행
    - $\mathbf{X}_s^o = \mathrm{Attention}(\mathbf{Q}_s, \mathbf{K}_m^K, \mathbf{V}_m^K)$ (Window 단위)
    - $\mathbf{X}_s^o$ 를 unpatch하여 전체 feature map 형태로 복원 $\mathbf{Y}_s^o \in \mathbb{R}^{h \times w \times c}$

Method

Computation Complexity Analysis of MVAS

- Cross-Attention의 연산량
  - $\Omega(\text{Cross-Attention}) = 2v(hw)^2c + v(hw)^2 = (2c+1)v(hw)^2 \approx O((hw)^2)$
- MVAS의 Neighbourhood Correlative Cross-Attention
  - $\Omega(\text{MVAS}) = 2v(a^2)^2c + v(a^2)^2c + 2\frac{hw}{a^2}\frac{khw}{a^2}c$

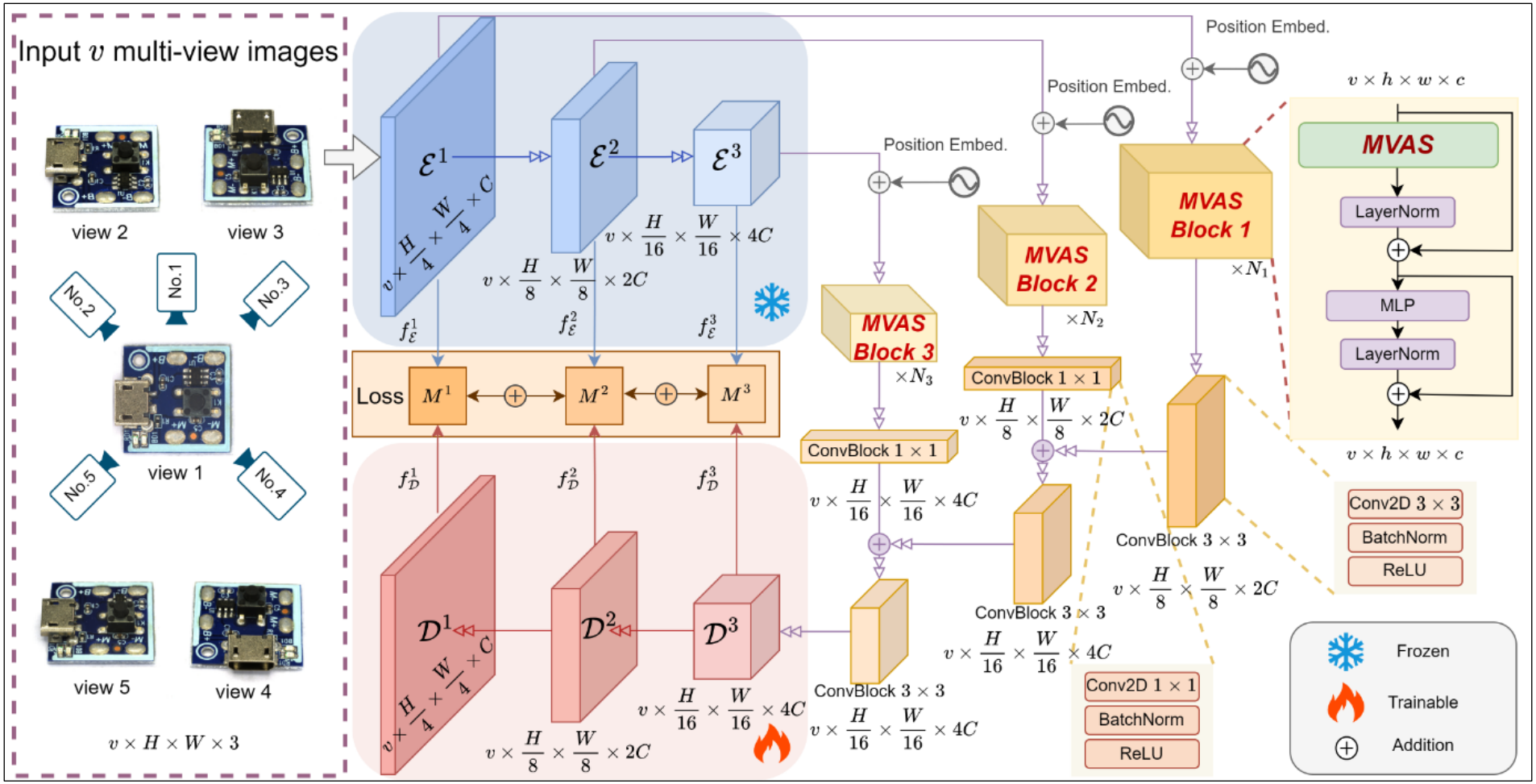
$$= 2c(va^4 + \frac{k(hw)^2}{a^4}) \geq 4c(va^4 \cdot \frac{k(hw)^2}{a^4})^{\frac{1}{2}}$$

$$= 4c(vk)^{\frac{1}{2}}(hw) \approx O((hw))$$

- 기존 Cross-Attention의 연산량은 feature의 크기 *hw*에 대해 이차 복잡도를 가짐
  - Query가 단일 view에서 오고 key/value는 모든 multi-view에서와 계산량이 매우 커짐
- MVAD의 모든 window이 아닌 top-K window에만 attention을 수행
  - 전체 연산 복잡도를 선형 수준으로 줄임

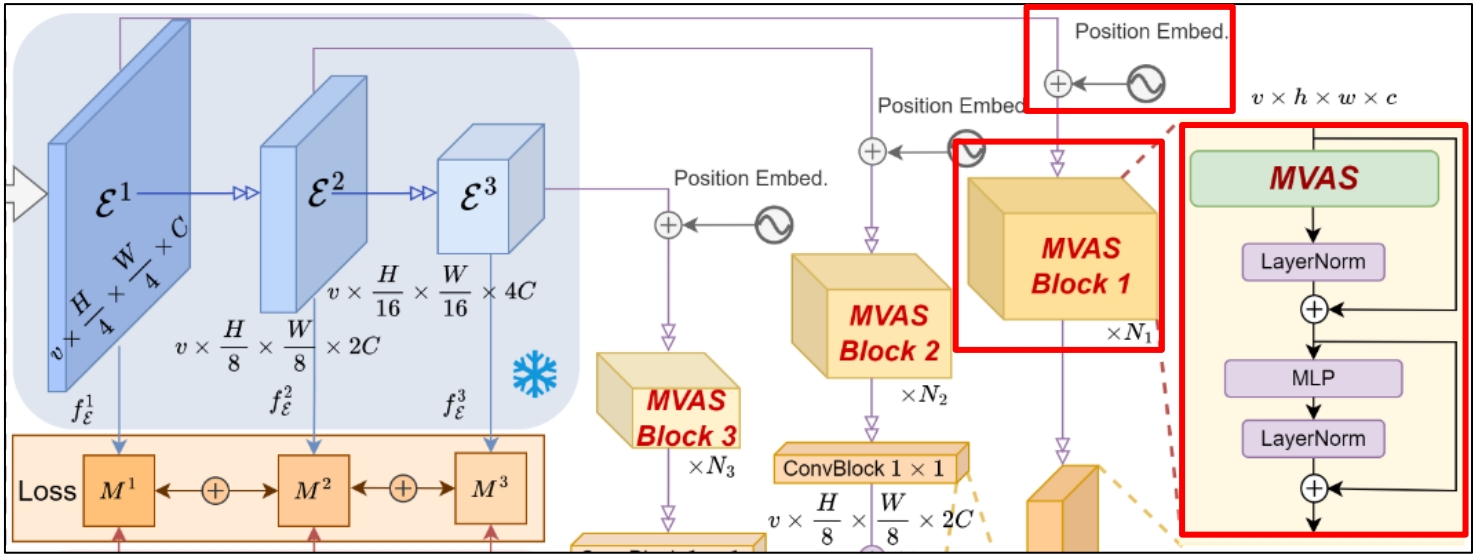
Method

Overall Architecture of MVAD



Method

Overall Architecture of MVAD

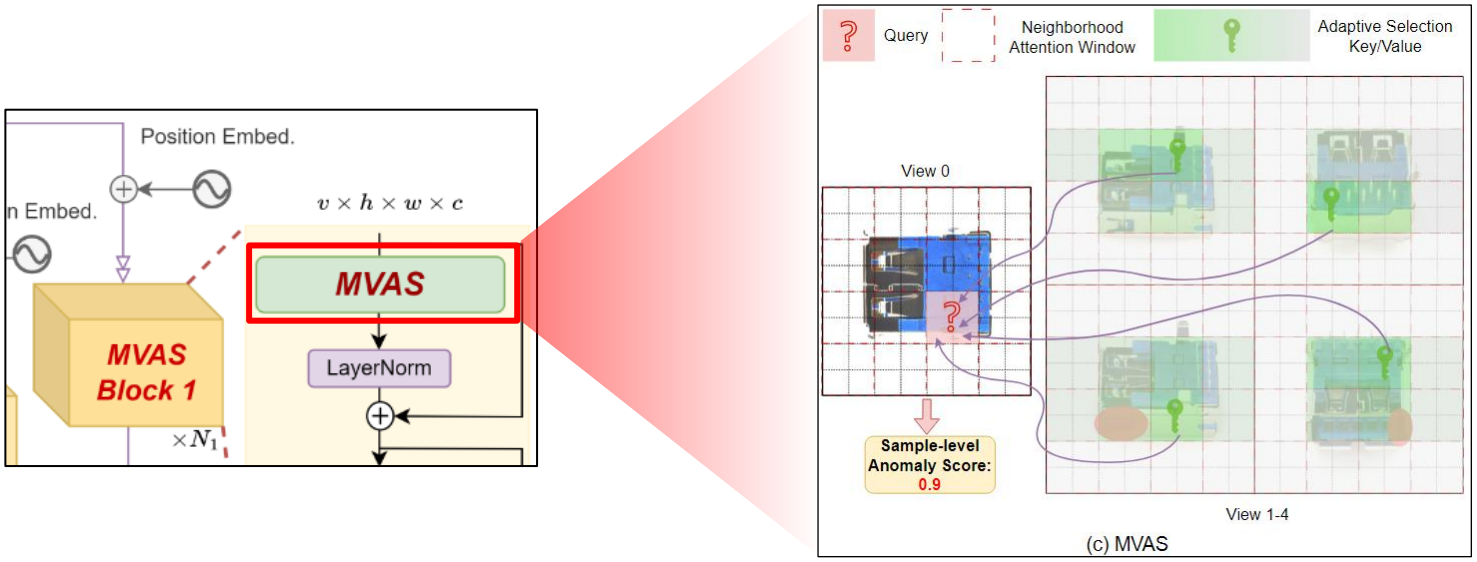


1. Pre-trained Encoder에서 추출된 feature를 Position Embed
2. Position embed가 된 feature를 MVAS Block에 넣음

- $X_o^j = \mathrm{LN^j}\left(\mathrm{MVAS^j}\left(X_i^j + \mathbb{P}\right)\right) + X_i^j$
  - X_i^j: Encoder에서 추출된 j번째 Scale의 View feature
  - $\mathbb{P}$: 위치 정보를 담은 Position Embedding
  - X_o^j: MVAS block을 거쳐 나온 multi-view fused feature

Method

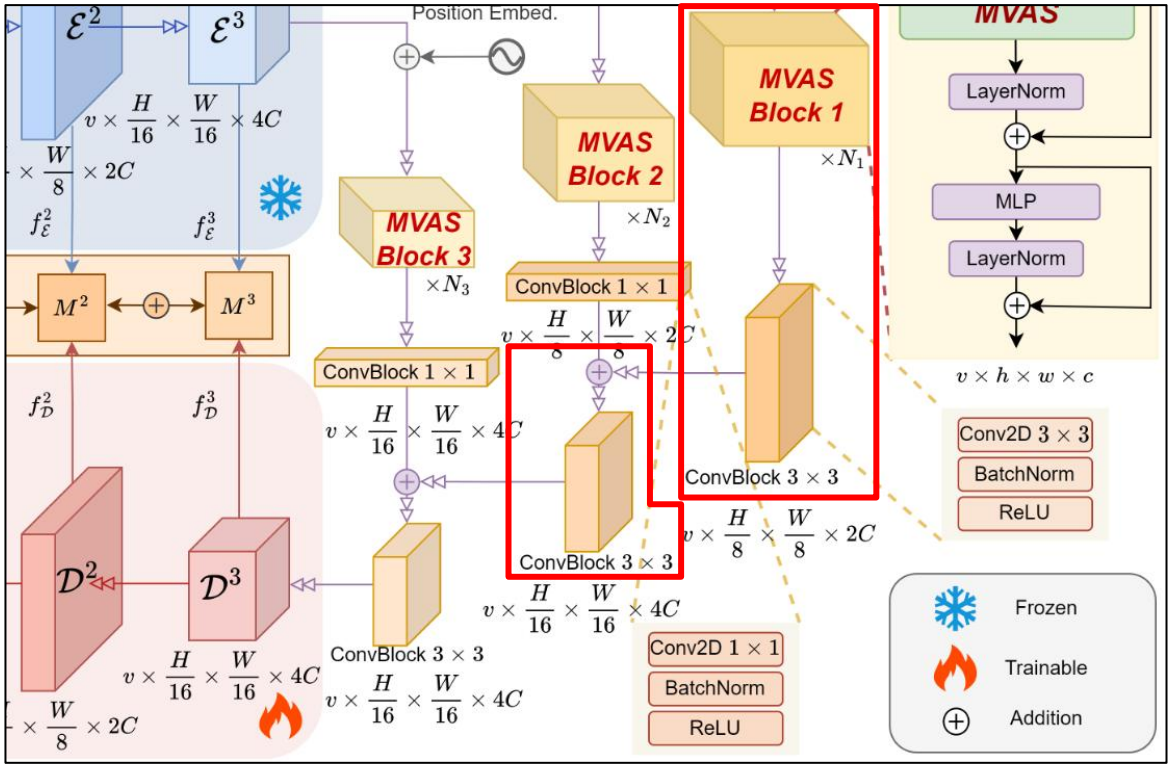
Overall Architecture of MVAD



- Semantic correlation matrix
  - View 0의 window와 View 1-4까지의 window간의 상간관계를 계산
    - ∴ 이 후 top-K 유사도를 가진 window만 사용
  - view 0의 window(Query)와 Top-k window(Key/ Value) attention 수행

Method

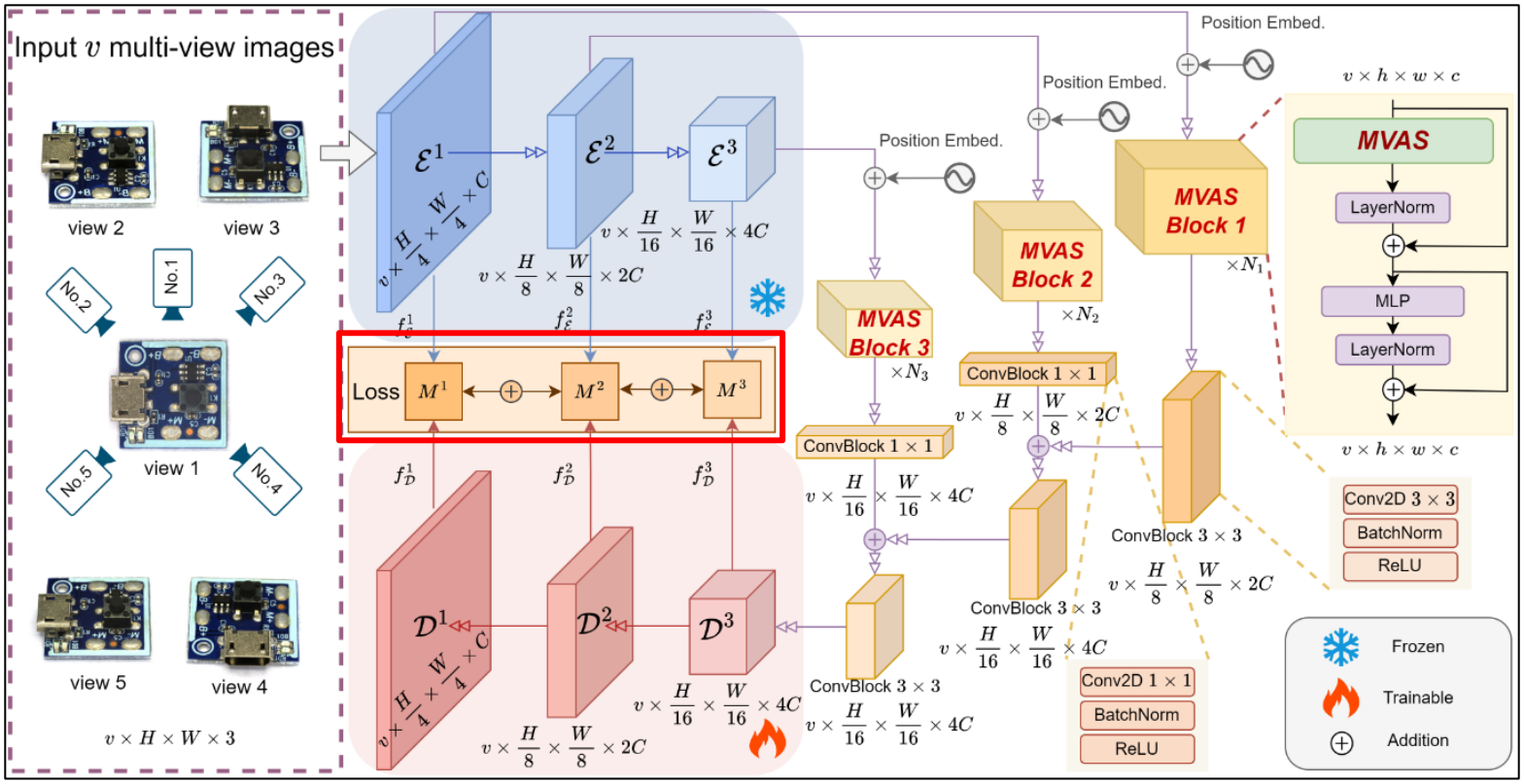
Overall Architecture of MVAD



3. MVAS 1에서 출력된 view의 feature tensor를 Conv 3x3에 통과
4. MVAS 2를 지나고 Conv 1x1 통과시킨 feature와 합침
   - MVAS Block 통과 전 각기 다른 크기의 Encoder의 채널 수를 맞춰 줌

Method

Overall Architecture of MVAD



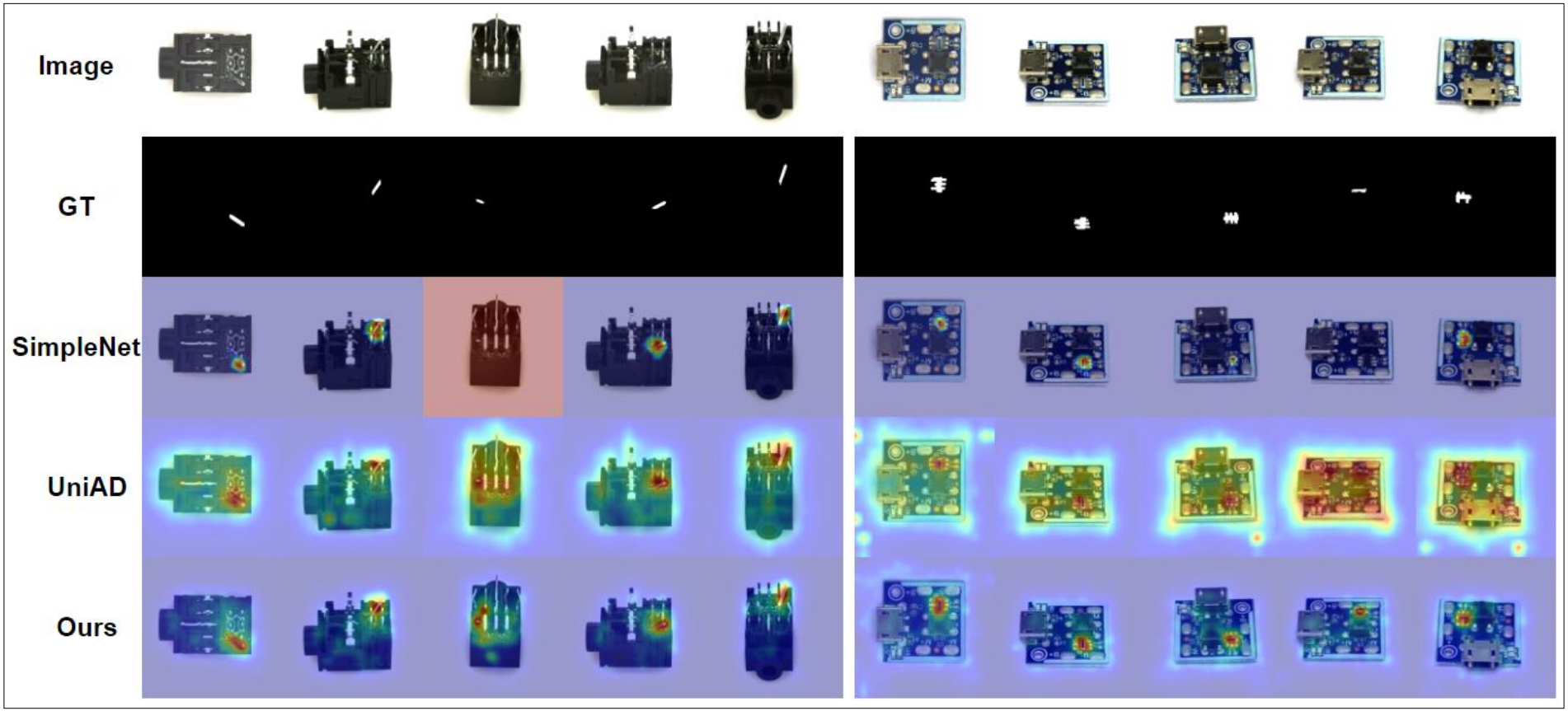
5. Multi-Scale Mean Squared Error (MSE)
   - Encoder와 Decoder Feature간 loss 계산
   - $M^1 \leftrightarrow M^2 \leftrightarrow M^3$ 간 연산적 연결

Experiment

- Implementation Details
  - 환경 세팅
    - GPU: RTX 3090 1개
    - Resize: 256 x256
    - LR: 0.005
    - Epochs: 100
    - Optimizer: Adam
- Evaluation Metrics
  - Image-level
    - Multi-class, Single-class
      - ✓AUROC, AP, F1-max
  - Sample-level
    - Multi-class, Single-class
      - ✓AUROC, AP, F1-max

Experiment

Result



Experiment

Result

Method →	Image-level multi-class AUROC/AP/F1-max			Image-level single-class AUROC/AP/F1-max		
	UniAD [45]	SimpleNet [19]	MVAD	UniAD [45]	SimpleNet [19]	MVAD
Category ↓	NeurIPS'22	CVPR'23	(Ours)	NeurIPS'22	CVPR'23	(Ours)
audiojack	**81.4/76.6/64.9**	58.4/44.2/50.9	80.3/72.8/62.8	78.7/60.8/64.0	**88.4/84.4/74.7**	86.9/82.5/70.6
bottle_cap	**92.5/91.7/81.7**	54.1/47.6/60.3	92.4/90.9/81.5	85.6/82.6/74.9	91.1/88.6/83.0	**95.6/95.4/87.0**
button_battery	75.9/81.6/**76.3**	52.5/60.5/72.4	**76.6/83.2**/75.6	65.9/71.9/74.6	88.4/89.9/83.5	**90.8/92.1/86.8**
end_cap	**80.9/86.1/78.0**	51.6/60.8/72.9	79.4/84.6/77.4	80.6/84.4/78.3	83.7/88.4/79.5	**85.8/88.8/81.8**
eraser	**90.3/89.2/80.2**	46.4/39.1/55.8	88.6/87.2/77.8	87.9/82.4/75.5	**91.6/90.1/80.1**	91.2/89.2/79.7
fire_hood	**80.6/74.8/66.4**	58.1/41.9/54.4	78.6/71.8/64.1	79.0/72.3/65.0	81.7/74.1/67.4	**84.6/77.2/69.8**
mint	67.0/66.6/**64.6**	52.4/50.3/63.7	**68.9/70.2**/64.5	64.5/63.8/63.9	77.0/78.5/68.5	**79.5/80.7/70.8**
mounts	87.6/77.3/**77.2**	58.7/48.1/52.4	**89.5/81.5**/76.5	84.1/71.2/71.0	**88.2/79.1**/76.3	87.9/75.6/**77.3**
pcb	81.0/88.2/79.1	54.5/66.0/75.5	**87.7/92.5/83.1**	84.0/89.2/81.9	89.3/93.5/84.4	**91.3/94.6/86.2**
phone_battery	83.6/80.0/71.6	51.6/43.8/58.0	**90.6/89.1/81.5**	83.7/75.9/73.5	86.8/81.9/76.0	**92.5/90.7/82.2**
plastic_nut	80.0/69.2/63.7	59.2/40.3/51.8	**84.9/77.2/69.5**	78.7/64.7/62.0	89.8/83.1/76.2	**91.3/85.7/77.5**
plastic_plug	81.4/75.9/67.6	48.2/38.4/54.6	**85.2/80.1/71.6**	70.7/59.9/61.1	87.5/83.7/74.0	**89.7/87.2/76.0**
porcelain_doll	85.1/75.2/69.3	66.3/54.5/52.1	**89.2/83.4/76.4**	68.3/53.9/53.6	85.4/76.8/68.9	**87.8/81.5/72.9**
regulator	56.9/41.5/44.5	50.5/29.0/43.9	**66.6/55.4/47.2**	46.8/26.4/43.9	81.7/68.3/63.4	**85.2/75.4/66.1**
rolled_strip_base	**98.7/99.3/96.5**	59.0/75.7/79.8	96.9/98.2/94.0	97.3/98.6/94.4	**99.5/99.7**/97.5	99.4/**99.7/97.6**
sim_card_set	89.7/90.3/83.2	63.1/69.7/70.8	**94.1/94.9/87.6**	91.9/90.3/87.7	95.2/94.8/**90.9**	**96.2/96.7**/90.2
switch	85.5/88.6/78.4	62.2/66.8/68.6	**89.1/91.6/81.8**	89.3/91.3/82.1	**95.2/96.2/89.4**	93.1/94.8/86.0
tape	**97.2/96.2**/89.4	49.9/41.1/54.5	96.8/96.1/**89.8**	95.1/93.2/84.2	96.8/95.2/89.1	**98.1/97.4/91.8**
terminalblock	87.5/89.1/81.0	59.8/64.7/68.8	**93.5/94.4/87.3**	84.4/85.8/78.4	94.7/95.1/89.4	**97.3/97.6/92.2**
toothbrush	78.4/80.1/75.6	65.9/70.0/70.1	**84.8/86.7/79.7**	84.9/85.4/81.3	**85.8/87.1**/80.5	85.5/84.2/**81.6**
toy	68.4/75.1/74.8	57.8/64.4/73.4	**79.1/84.2/78.0**	79.7/82.3/80.6	83.5/87.6/80.9	**86.5/90.2/82.6**
toy_brick	**77.0/71.1/66.2**	58.3/49.7/58.2	66.4/58.8/60.6	80.0/73.9/68.6	**81.8/78.8/70.5**	77.9/73.6/67.0
transistor1	93.7/95.9/88.9	62.2/69.2/72.1	**94.3/96.0/89.2**	95.8/96.6/91.1	97.4/98.1/93.5	**97.9/98.4/93.6**
u_block	88.8/**84.2/75.5**	62.4/48.4/51.8	**89.1**/84.0/74.2	85.4/76.7/69.7	90.2/82.8/76.8	**93.1/90.2/81.3**
usb	78.7/79.4/69.1	57.0/55.3/62.9	**90.1/90.5/81.9**	84.5/82.9/75.4	90.3/90.0/83.5	**92.8/92.1/83.9**
usb_adaptor	76.8/71.3/64.9	47.5/38.4/56.5	**78.1/72.4/66.1**	78.3/70.3/67.2	82.3/78.0/67.9	**83.8/78.7/70.8**
vcpill	**87.1/84.0/74.7**	59.0/48.7/56.4	83.7/80.9/70.5	83.7/81.9/70.7	90.3/88.8/79.6	**90.8/90.1/80.4**
wooden_beads	78.4/77.2/67.8	55.1/52.0/60.2	**84.3/83.1/73.1**	82.8/81.5/71.4	86.1/84.7/75.7	**89.5/88.9/79.3**
woodstick	**80.8/72.6/63.6**	58.2/35.6/45.2	78.0/65.3/59.8	79.7/70.4/61.8	78.3/70.3/62.3	**85.7/77.9/70.0**
zipper	98.2/98.9/95.3	77.2/86.7/77.6	**98.9/99.4/95.7**	97.5/98.4/94.2	98.7/99.2/95.6	**99.4/99.6/97.1**
Average	83.0/80.9/74.3	57.2/53.4/61.5	**85.2/83.2/76.0**	81.6/77.3/73.4	88.9/87.4/80.4	**90.2/88.2/81.0**

Experiment

Result

Method →	Sample-level multi-class AUROC/AP/F1-max			Sample-level single-class AUROC/AP/F1-max		
	UniAD [45]	SimpleNet [19]	MVAD	UniAD [45]	SimpleNet [19]	MVAD
Category ↓	NeurIPS'22	CVPR'23	(Ours)	NeurIPS'22	CVPR'23	(Ours)
audiojack	90.3/**95.2**/88.3	68.3/82.4/82.0	**91.2**/94.9/**90.7**	88.4/93.4/88.4	93.0/**96.5**/91.2	**93.1**/96.3/**91.5**
bottle_cap	97.2/98.7/93.6	51.2/70.4/80.7	**97.8**/**98.9**/**95.3**	90.8/95.2/89.7	**99.3**/**99.7**/**98.9**	98.9/99.5/96.6
button_battery	**95.5**/**95.1**/**95.0**	57.2/75.2/81.3	80.0/90.4/83.4	88.3/91.8/87.1	96.1/96.4/95.0	93.4/96.4/93.1
end_cap	**87.7**/**94.2**/86.4	55.7/73.7/81.2	86.6/93.0/**87.4**	85.3/92.0/86.7	**95.1**/**97.7**/**91.8**	91.9/96.1/89.8
eraser	**90.1**/**95.3**/**87.9**	35.8/61.6/80.9	87.2/94.3/85.2	93.2/96.8/90.1	**94.3**/**97.5**/**90.5**	91.8/96.3/88.1
fire_hood	**85.9**/**92.9**/**85.5**	54.2/69.4/80.6	81.8/90.7/82.1	89.4/94.1/90.5	**93.5**/**96.5**/**90.9**	90.2/95.1/87.2
mint	66.5/89.7/89.7	53.1/84.7/89.7	**68.1**/**91.0**/**89.7**	61.0/88.0/89.7	**86.1**/**96.7**/90.3	85.9/96.5/**91.3**
mounts	98.3/99.2/95.5	64.1/79.7/80.7	**98.8**/**99.5**/**96.1**	93.4/96.9/89.8	99.5/99.7/98.1	**99.6**/**99.8**/**98.2**
pcb	83.8/91.6/85.2	61.2/77.2/81.1	**90.0**/**95.3**/**87.8**	79.4/86.7/86.9	**91.5**/95.7/**89.1**	91.1/**95.9**/88.4
phone_battery	85.4/93.1/85.3	62.4/78.4/81.3	**92.7**/**96.7**/**90.5**	91.5/96.1/89.3	**96.3**/**98.1**/**95.0**	94.2/97.2/91.4
plastic_nut	84.7/89.5/87.6	48.2/66.3/80.0	**90.8**/**94.7**/**89.1**	86.4/88.8/91.6	96.1/97.5/94.3	**97.2**/**98.5**/**95.4**
plastic_plug	80.2/90.3/82.9	50.2/71.0/80.9	**89.1**/**94.6**/**87.6**	65.6/81.2/82.4	**96.0**/**98.2**/**93.2**	94.4/97.3/92.3
porcelain_doll	90.1/91.9/90.4	80.2/89.1/84.2	**94.9**/**97.3**/**93.6**	65.9/78.4/81.9	**96.6**/98.2/**96.2**	96.2/**98.3**/93.9
regulator	65.7/80.6/**82.1**	49.9/68.1/80.7	**73.6**/**86.8**/81.2	52.1/69.8/80.9	**96.3**/**98.5**/**93.9**	87.7/93.5/88.8
rolled_strip_base	**98.6**/**99.3**/**97.5**	65.5/80.8/80.7	97.7/98.7/95.4	97.8/99.0/95.0	**99.8**/**99.9**/**99.0**	99.6/99.8/98.9
sim_card_set	87.3/90.8/87.0	77.1/86.7/82.9	**96.2**/**97.9**/**95.0**	93.2/93.6/94.1	**99.0**/**99.5**/**98.6**	98.2/99.1/96.6
switch	91.5/96.2/88.7	66.8/82.0/81.6	**94.9**/**97.6**/**91.4**	92.9/96.4/90.7	**98.6**/**99.4**/**97.0**	96.3/98.4/92.9
tape	98.2/99.0/**96.0**	54.2/73.9/80.7	**98.4**/**99.3**/**96.0**	94.8/97.4/91.7	99.9/**100.**/99.4	**100.**/**100.**/**99.5**
terminalblock	95.7/98.2/93.1	75.4/87.8/81.4	**96.8**/**98.7**/**94.2**	82.9/92.2/83.4	98.2/99.3/**97.0**	**98.6**/**99.4**/96.5
toothbrush	**89.8**/92.9/**90.0**	71.2/83.7/81.2	87.0/**93.1**/87.5	94.3/96.6/93.2	**96.1**/**98.1**/93.3	95.8/97.0/**93.4**
toy	75.4/85.7/85.0	59.0/74.3/80.5	**86.4**/**92.3**/**86.5**	86.3/90.9/89.0	**93.8**/**96.9**/**90.2**	93.6/96.8/90.9
toy_brick	**78.6**/**84.3**/**84.2**	59.2/73.6/80.6	69.4/81.7/80.8	80.1/85.6/85.1	**87.1**/**92.6**/**88.1**	80.8/89.7/83.5
transistor1	98.4/99.2/95.7	66.8/81.6/80.4	**98.8**/**99.5**/**96.3**	96.1/97.7/94.1	**99.8**/**99.9**/98.1	**99.8**/**99.9**/**98.3**
u_block	**94.8**/**97.0**/**91.2**	50.5/75.3/80.0	93.0/96.1/90.0	91.2/94.9/90.4	98.7/99.4/96.8	98.4/99.0/96.8
usb	79.7/88.4/82.7	64.1/79.9/81.2	**92.1**/**95.9**/**89.0**	84.0/88.9/86.9	93.9/96.1/92.1	**96.3**/**97.7**/**94.9**
usb_adaptor	82.9/90.3/86.3	48.8/70.4/82.5	**91.0**/**95.5**/**90.4**	78.4/87.7/84.4	**93.8**/**97.1**/**92.6**	93.3/96.5/92.2
vcpill	80.7/89.9/83.6	64.5/78.7/81.0	88.2/94.2/85.3	79.8/88.9/83.8	**96.8**/**98.4**/**93.7**	95.9/98.1/91.8
wooden_beads	77.3/90.1/86.2	59.5/81.3/83.9	**82.2**/**92.7**/**86.3**	78.5/90.1/86.9	92.1/97.0/**91.7**	**92.7**/**97.2**/90.8
woodstick	**84.0**/**90.7**/**85.1**	58.2/72.7/80.0	76.8/87.5/81.6	81.8/87.3/84.9	80.3/90.5/81.7	**85.0**/**92.5**/**84.3**
zipper	97.8/98.6/96.8	91.9/96.0/88.8	**99.9**/**99.9**/**98.7**	94.6/96.2/93.5	**99.8**/**99.9**/**98.8**	99.6/99.8/98.6
Average	87.1/92.9/88.8	60.8/77.5/81.8	**89.0**/**94.6**/**89.5**	84.6/91.1/88.4	**94.6**/**97.3**/**93.3**	94.3/**97.3**/92.9

Experiment

Result

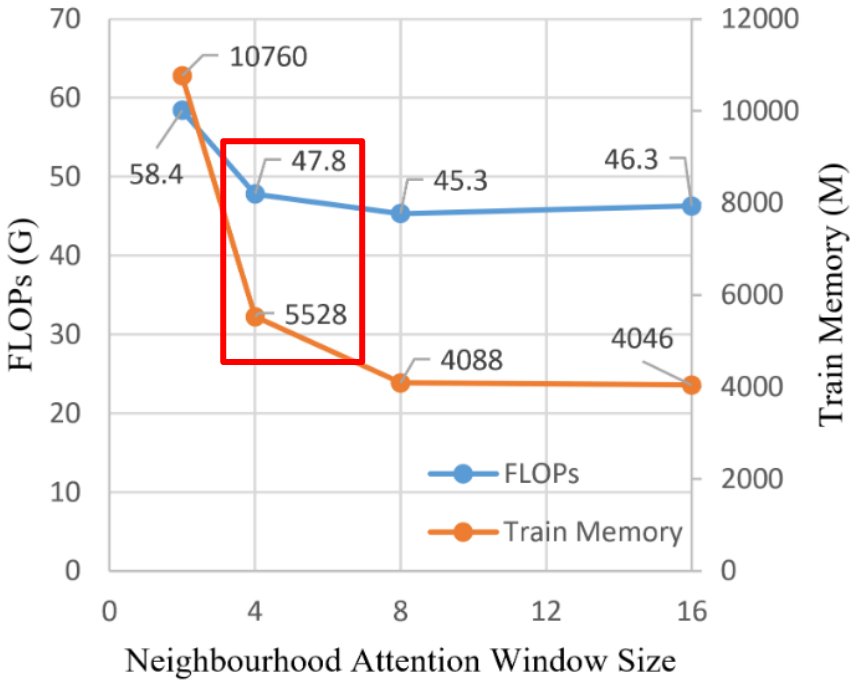


Fig. 4. Ablation studies on window size with $k = 4$

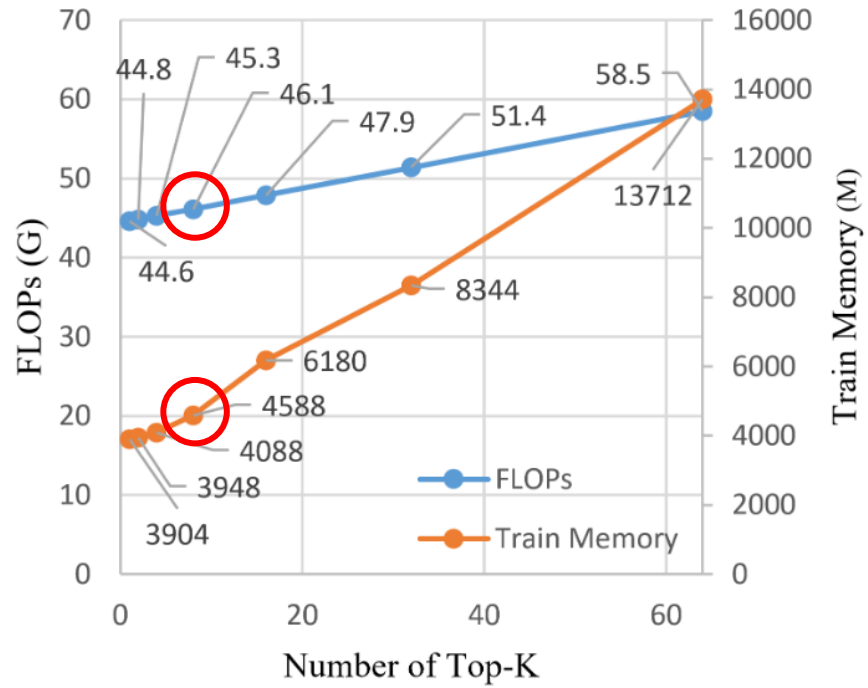


Fig. 5. Ablation studies on top-K with $a = 8$

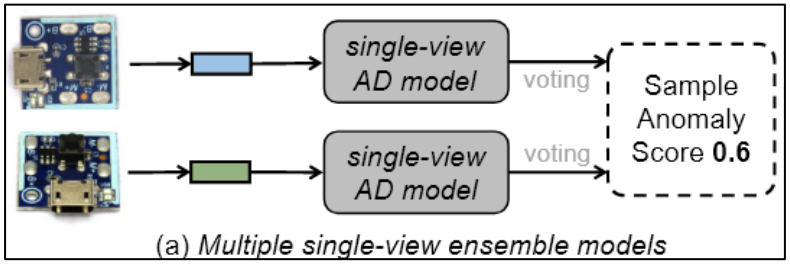
Unveiling Multi-View Anomaly Detection: Intra-view Decoupling and Inter-view Fusion

[AAAI 2025]

Introduction

IDIF (Intra-view Decoupling and Inter-view Fusion)

- 단일 시점 기반 AD는 사각지대에 숨겨진 defect를 놓칠 수 있음
- 각 view 마다 개별 모델로 학습시키고 그 결과를 voting 하는 방식(fig. a)
  - 계산 비용 多, 다양한 view 간의 잠재적 상관관계와 보완 정보를 무시하게 됨
    - ⁘ 이는 feature anomaly를 식별하는데 매우 중요
- 실제 생산 라인에서는 작업자들이 여러 각도에서 이상 영역을 점검
  - 다양한 시점에서의 관찰은 같은 영역이 조명, 깊이가 서로 다르게 보임
    - ⁘ 서로 다른 시점에의 이상영역은 서로 보완적임
- 하나의 모델로 다중 시점 이상을 탐지하면 이러한 보완적 특성을 활용하여 탐지 신뢰도를 높일 수 있음

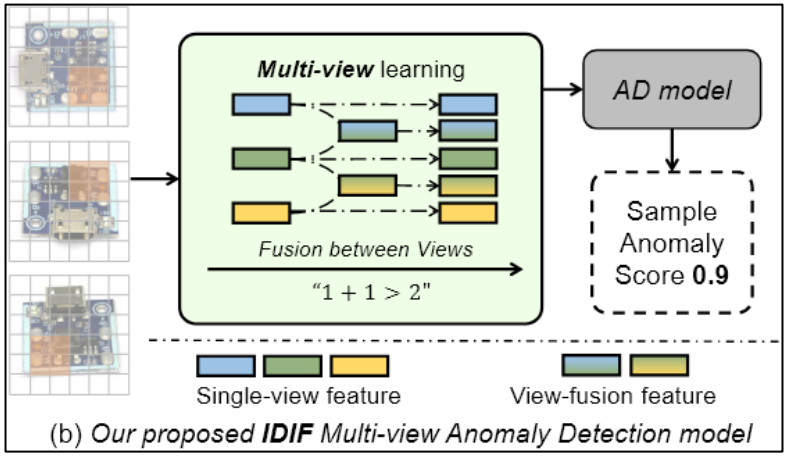


(a) *Multiple single-view ensemble models*

Introduction

Contribution

1. IDIF: Multi-view anomaly detection framework
   - Intra-view Decoupling (ID)
     - 각 view에서 겹쳐지는 common view와 각 view만 가지는 고유한 specific view를 구별
   - Inter-view Fusion (IF)
     - 각 view만이 가지는 고유한 specific view를 3D voxel 형태로 결합
2. View-wise Dropout strategy
   - Missing view 상황을 만드는 robust한 학습 방식 도입



(b) *Our proposed **IDIF** Multi-view Anomaly Detection model*

Intra-view Decoupling

Consistency Bottleneck (CB)

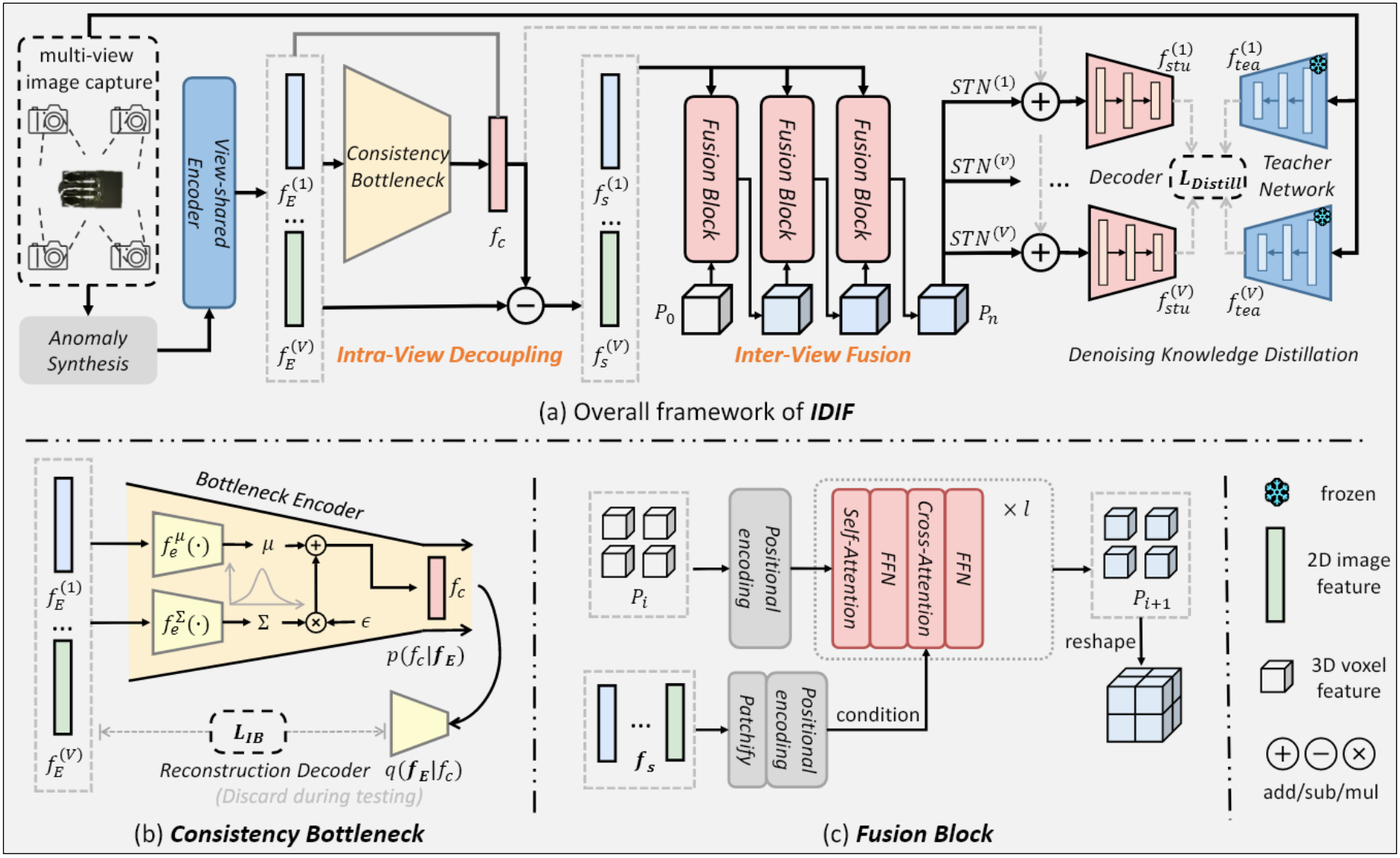
- 핵심 아이디어
  - 각 시점에서 추출된 feature를 view-common과 view-specific으로 분해
    - 시점 간 공유되는 feature는 압축하고 고유한 feature는 남겨 후속 fusion에 활용
- 주요 구성
  1. Common feature(view common)
     - 모든 view의 feature를 저차원 공간으로 압축(정보 병목 → Consistency Bottleneck)
     - Mutual information을 최대화하여 공통된 정보가 f_c에 포함되도록 함
  2. View-specific feature 추출
     - 각 시점의 원래 feature에서 f_c를 빼서 View-specific feature를 추출
       - ✓ $f_S^{(v)} = f_E^{(v)} - f_c$
- 목적
  - View-common은 fusion 과정에서 제거하여 과적합 방지, 고유성 보존
  - 시점마다 다르게 나타나는 이상을 효과적으로 포착

Inter-view Fusion

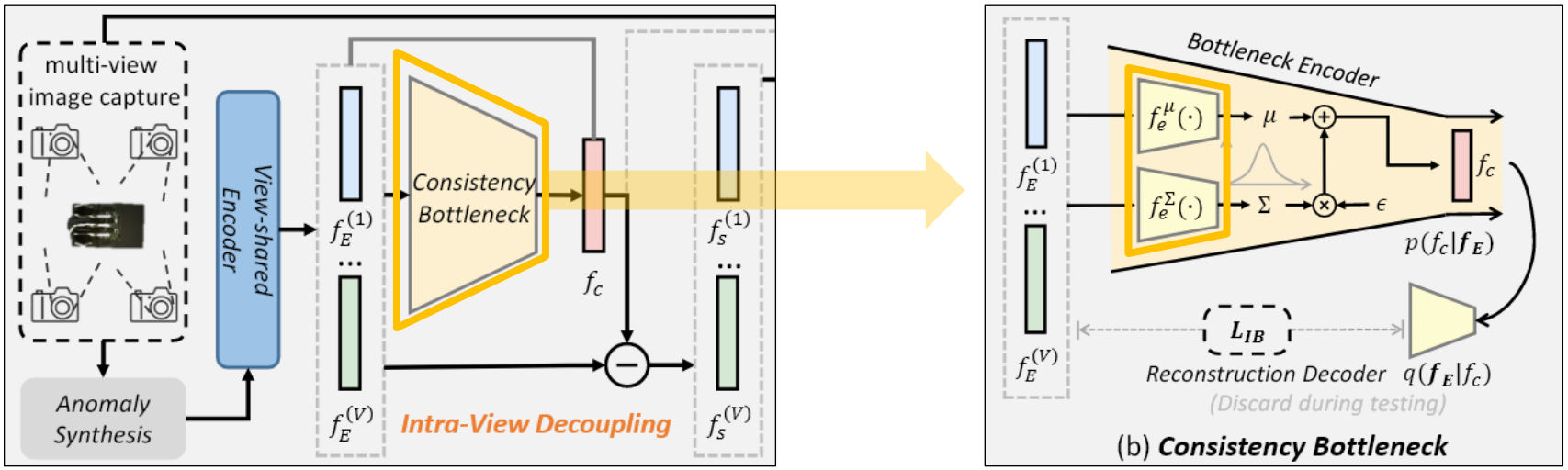
Implicit Voxel Construction (IVC)

- 핵심 아이디어
  - 각 시점에서 얻은 View-specific feature를 3D voxel에 fusion (공간적 구조 반영)
  - View 간 정보를 입체적으로 결합해서 더 풍부한 표현을 만듦
- 주요 구성
  1. 3D Voxel 프로토 타입
     - Learnable 샘플 독립적인 3D 구조와 초기 상태
     - Voxel은 view-specific feature와 cross-attention을 통해 업데이트
  2. Fusion Blocks
     - Self-attention → Voxel 자체 정보 보존
     - Cross-attention → 각 시점의 고유 정보와 voxel 간 상호작용
     - 반복적으로 수행되며 점진적으로 Voxel 정보가 통합
  3. 2D 재투영 (STN)
     - 학습 가능한 각도를 통해 3D voxel을 다시 2D 평면으로 투영
     - 각 시점의 view-specific decoder에 전달

Overview


multi-view image capture
Anomaly Synthesis
View-shared Encoder
Consistency Bottleneck
Intra-View Decoupling
Fusion Block
Fusion Block
Fusion Block
Inter-View Fusion
Decoder
Teacher Network
Denoising Knowledge Distillation
(a) Overall framework of IDIF
Bottleneck Encoder
Reconstruction Decoder
(Discard during testing)
(b) Consistency Bottleneck
Positional encoding
Self-Attention
FFN
Cross-Attention
FFN
reshape
Patchify
Positional encoding
condition
frozen
2D image feature
3D voxel feature
add/sub/mul
(c) Fusion Block

Overview



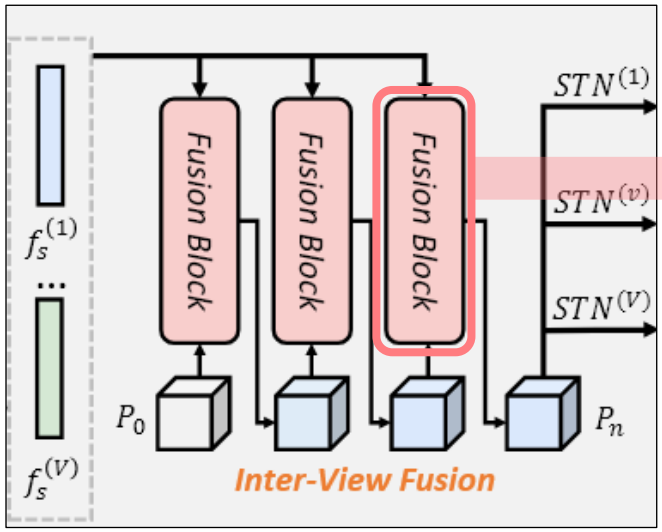
- CB(Consistency Bottleneck)
  1. Multi-view의 feature들을 정규 분포 파라미터로 변환

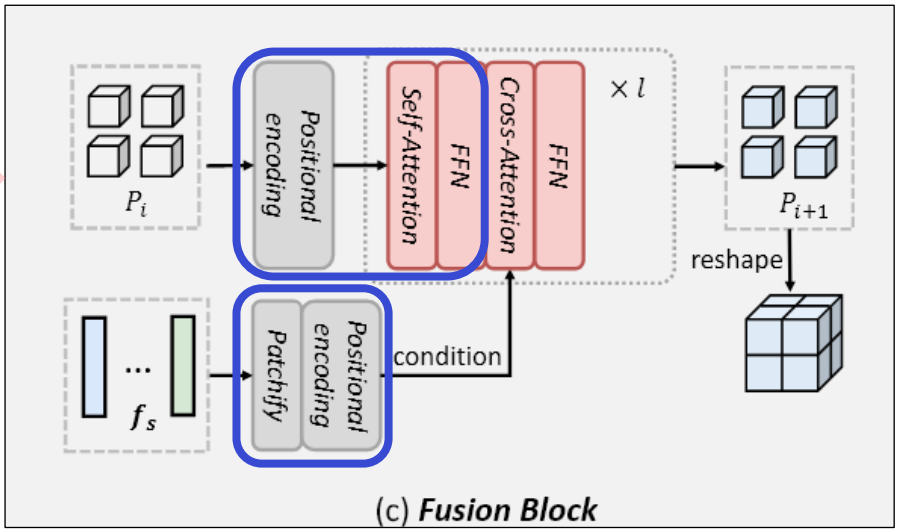
     ∴ $f_e^{\mu}(\cdot)$: 평균(μ) 추출기, $f_e^{\Sigma}(\cdot)$: 분산($\sum$) 추출기

     ✓입력 feature들을 이용해 잠재 공간에서 정규 분포를 구성
  2. 평균 분산을 기준으로 정규 분포 $\mathcal{N}(\mu, \Sigma)$로 부터 샘플링해서 f_c (common feature)를 구함
  3. 전체 feature에서 f_c (common feature) 제외한 $f_E^{(v)}$를 view-specific feature로 분류

     ∴ $f_S^{(v)} = f_E^{(v)} - f_c$

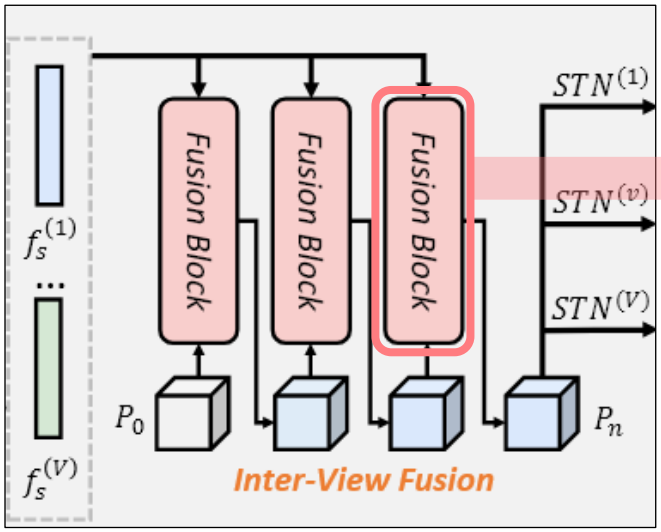
Overview

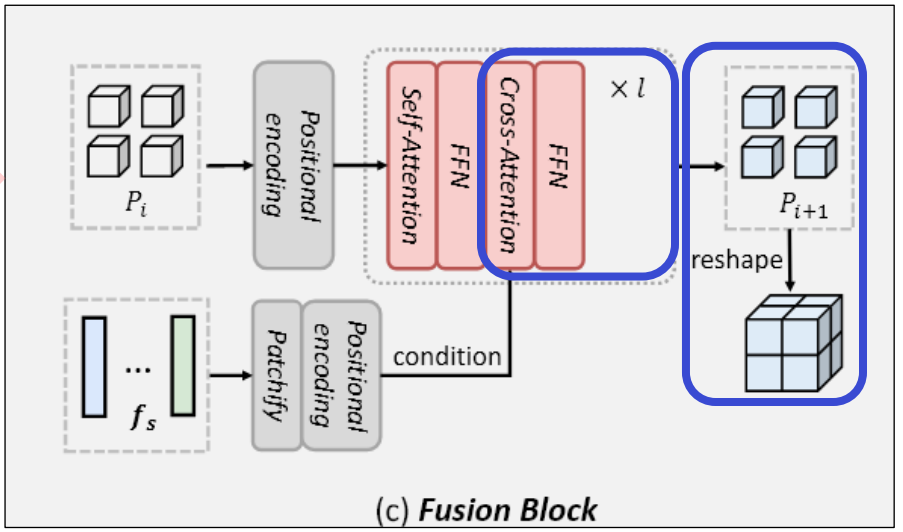




- IF(Inter-view Fusion)
  - f_s
    - patchify해서 1D token sequence로 만듦
    - Positional encoding을 통해 위치 정보를 더해 줌 (공간 구조 보존)
  - P_i
    - Positional Encoding (각 voxel 위치에 해당하는 좌표 정보를 embedding으로 더함)
    - Self-Attention (voxel 내부 위치들끼리 서로 관계 학습)
    - FFN (voxel 위치의 출력을 비선형으로 변환)

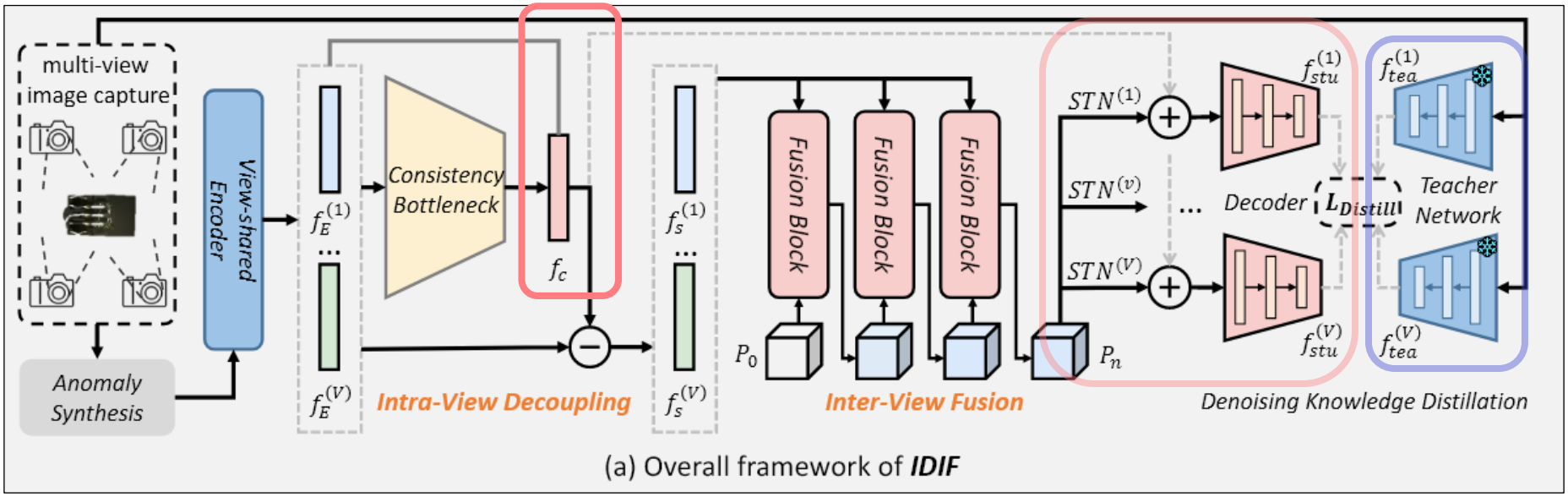
Overview





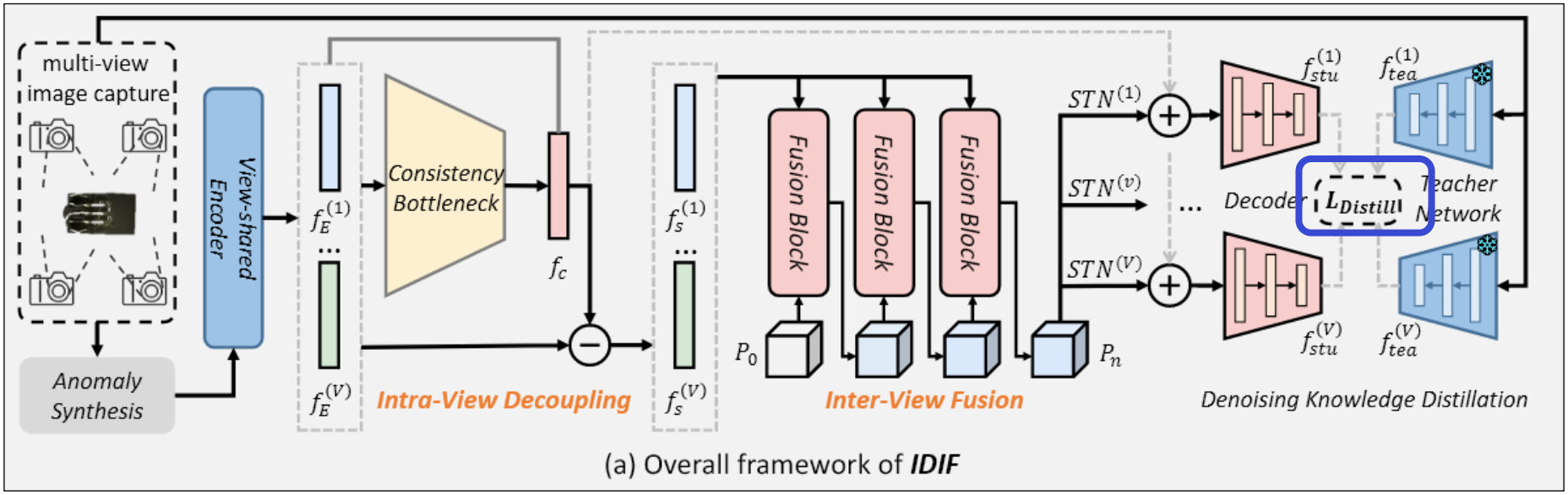
- IF(Inter-view Fusion)
  - Cross-Attention
    - Query: P_i (voxel) → 학습 중인 공간 구조
    - Key/ Value: f_s (view-specific feature) → 각 view에서 추출한 특화 정보
  - FFN
    - Cross-Attention 조정된 voxel 단위의 feature를 각 위치별로 강화 및 비선형 변환
  - Reshape
    - 최종적으로 얻은 1D token sequence를 3D voxel로 reshape

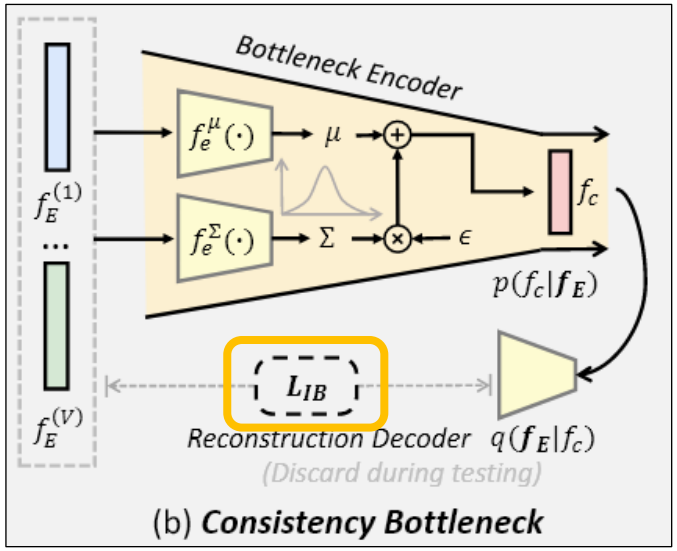
Overview



- STN (spatial Transformer Network)
  - 3D voxel P_n을 각 view 별로 다시 2D 평면에 투영
    - 각 view마다 STN 모듈 하나씩 존재
- STN을 각 view마다 f_c(common view)를 더함
- $L_{Distill}$
  - $f_p^{(v)} = STN^{(v)}(P_n, \alpha^{(v)}),\ v = 1, ..., V.$
  - $L_{\text{Distill}} = \sum_{v=1}^{V} \| f_{\text{stu}}^{(v)} - f_{\text{tea}}^{(v)} \|^2$

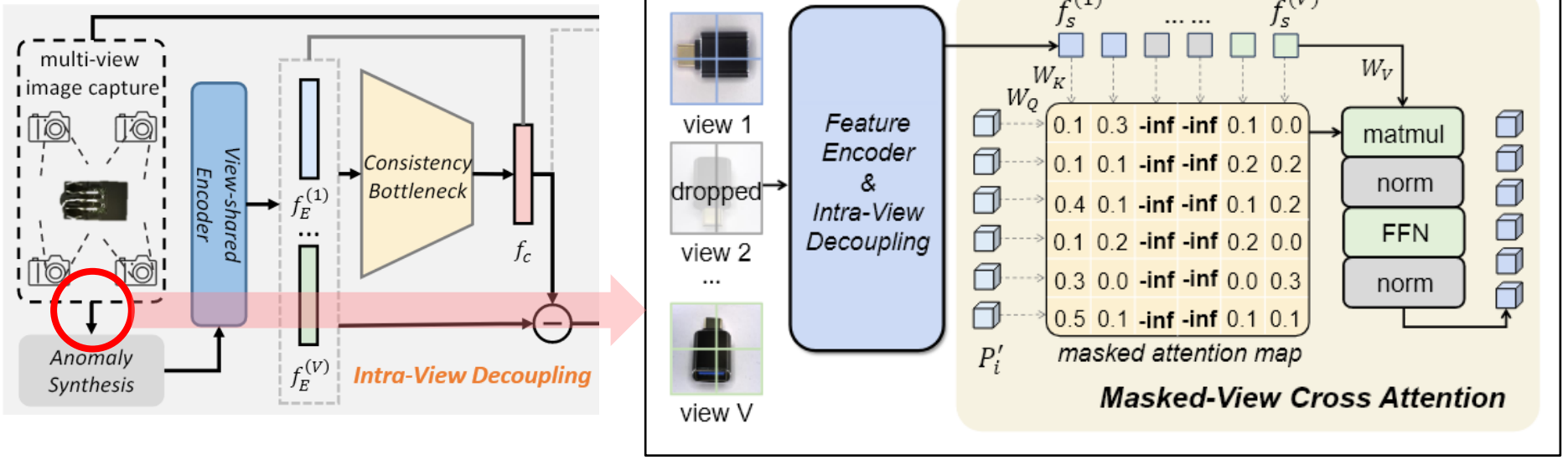
Overview





- L_{IB} (Intra-batch consistency loss)
  - Bottleneck Encoder로 $p(f_c|f_E)$, f_c 추출
    - VAE(Variational Autoencoder) 구조
  - Reconstruction Decoder로 $q(f_E|f_c)$ 복원
    - 기존 $f_E^{(1)} \sim f_E^{(V)}$와 L_{IB} 구함
    - View-common 정보의 품질을 평가

Overview

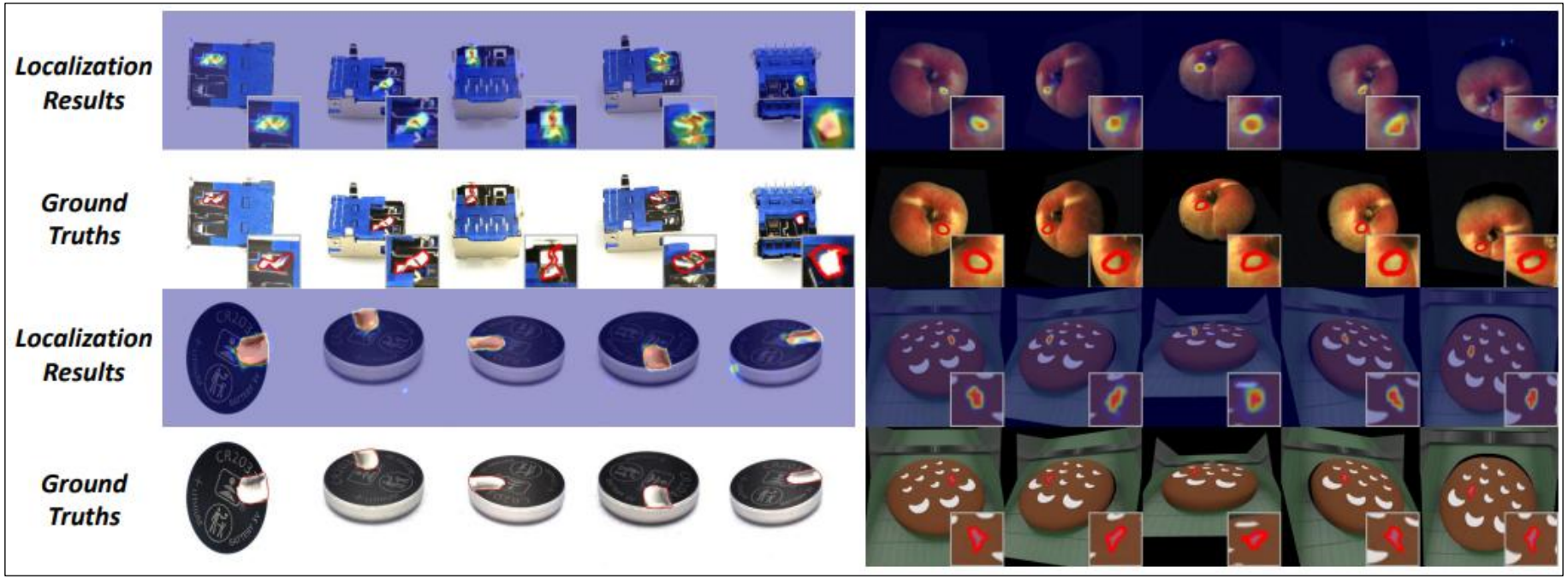


- View-wise Dropout
  - Train에서 일부 view의 image를 임의로 랜덤하게 dropout
    - 의도적으로 일부 view 정보를 제거해 누락 상황을 simulation
- Masked-View cross attention
  - Dropout으로 빠진 view에 대해서 attention을 못하게 강제로 masking

Experiment

- Datasets
  - Real-IAD
  - MVTec 3D-AD
  - Eyecandies
- Evaluation Metrics
  - S-AUROC (Sample-level AUROC)
    - 모든 view가 정상일 때만 해당 샘플을 정상으로 간주
  - P-AUROC (Pixel-level AUROC)
    - 이상 위치(localization) 평가
- Implementation Details
  - Backbone: ResNet18 (ImageNet pretrained)
    - 사용 블록: block1 (64x64), block2 (32x32), block3 (16x16)
    - Input Size: 256 × 256
    - Batch Size: 8

Experiment


Localization Results
Ground Truths
Localization Results
Ground Truths

Experiment

Category	PaDim	PatchCore	RD	UniAD	DeSTSeg	SimpleNet	**Ours**
Audiojack	92.2 / 94.2	89.3 / **98.4**	81.9 / 97.9	91.2 / 97.2	95.3 / 97.3	91.2 / 98.2	**96.1 / 98.4**
Bottle Cap	98.1 / 97.7	**99.4** / 99.3	93.7 / 99.0	97.3 / 99.2	92.4 / 99.4	**99.4** / 98.5	98.2 / **99.7**
Button Battery	88.7 / 90.4	90.6 / **98.8**	83.3 / 98.3	87.5 / 93.7	93.3 / 98.7	95.8 / 98.0	**97.7 / 98.8**
End Cap	76.1 / 95.2	91.9 / **98.0**	68.1 / 97.3	89.4 / 96.7	82.3 / 95.4	**94.2** / 94.2	94.1 / 97.7
Eraser	96.5 / 95.7	95.6 / 98.7	82.9 / 98.6	91.2 / 99.0	91.9 / 99.4	94.7 / 98.3	**96.6 / 99.5**
Fire Hood	**96.9** / 94.5	89.3 / 98.7	81.4 / 98.3	83.0 / 98.5	**96.9** / 98.6	95.6 / 97.5	92.6 / **99.2**
Mint	69.1 / 90.3	85.7 / **98.8**	67.7 / 97.7	73.0 / 94.3	77.7 / 93.9	86.8 / 94.1	**87.8** / 97.9
Mounts	98.4 / 97.5	**99.7** / 98.3	92.5 / 98.3	97.0 / 99.4	99.1 / 99.3	99.4 / 98.0	99.3 / **99.5**
PCB	88.4 / 91.0	93.0 / 99.1	79.3 / 98.8	83.2 / 96.6	83.6 / 98.5	90.7 / 98.4	**94.2 / 99.3**
Phone Battery	91.7 / 89.6	95.1 / 98.9	89.4 / 98.9	93.6 / 97.9	98.2 / 96.6	94.7 / 96.5	**98.5 / 99.3**
Plastic Nut	**98.2** / 95.1	97.8 / 98.8	72.8 / 98.9	87.1 / 98.6	94.4 / 98.2	95.7 / 98.1	96.9 / **99.5**
Plastic Plug	87.4 / 93.5	95.7 / 98.5	89.3 / 98.3	78.0 / 97.9	95.6 / 95.1	94.4 / 96.1	**97.6 / 99.1**
Porcelain Doll	93.8 / 91.7	96.1 / 98.0	89.6 / 98.2	92.8 / 97.3	94.6 / 97.5	96.2 / 96.6	**98.6 / 98.4**
Regulator	96.5 / 91.9	86.0 / 99.2	92.5 / **98.7**	55.5 / 93.7	93.0 / **98.7**	92.0 / 97.0	**99.8 / 98.7**
Rolled Strip Base	98.6 / 92.3	**99.7** / 99.1	80.3 / 99.0	99.3 / 98.9	98.9 / 99.4	99.6 / 98.8	**99.7 / 99.8**
SIM Card Set	94.2 / 85.4	99.3 / 99.0	89.9 / 97.7	94.0 / 96.7	98.3 / 97.6	99.2 / 97.3	**99.6 / 99.3**
Switch	82.1 / 97.3	94.6 / 98.5	87.3 / 98.6	95.3 / 99.4	96.6 / 99.5	**98.8** / 99.1	98.3 / **99.7**
Tape	99.8 / 97.9	99.9 / 99.1	89.5 / 99.0	99.1 / 99.5	99.1 / 99.6	**100.0** / 99.2	98.7 / **99.8**
Terminal Block	96.9 / 96.7	97.5 / 99.2	89.8 / 99.0	93.8 / 98.9	96.1 / 99.7	97.7 / 99.3	**97.9 / 99.8**
Toothbrush	91.7 / 87.2	94.7 / 96.2	86.7 / 96.3	95.0 / 96.8	97.9 / 92.1	95.3 / 94.3	**99.0 / 97.3**
Toy	91.4 / 83.3	92.8 / **98.3**	75.0 / 95.2	77.2 / 96.4	96.5 / 91.4	92.9 / 91.9	**97.8** / 96.6
Toy Brick	84.3 / 94.1	82.6 / 97.5	72.5 / 96.3	78.3 / 97.9	87.0 / 96.2	85.7 / 94.3	**92.8 / 98.7**
Transistor1	90.3 / 95.4	**99.8** / 98.9	94.7 / 98.8	99.3 / 98.8	99.0 / 98.5	99.7 / 99.1	**99.8 / 99.4**
U Block	98.3 / 96.3	**98.8** / 98.9	86.9 / 98.4	96.3 / 99.0	98.5 / 99.5	98.5 / 98.6	98.5 / **99.6**
USB	77.0 / 93.6	93.9 / 99.1	89.4 / 98.9	83.1 / 98.5	93.3 / 97.3	93.9 / 98.9	**98.4 / 99.6**
USB Adaptor	93.2 / 93.0	90.6 / 98.2	65.3 / 96.5	85.1 / 97.0	93.6 / 96.8	93.0 / 95.7	**95.1 / 98.6**
Vcpill	94.7 / 93.4	96.5 / 98.3	87.2 / 97.7	89.4 / **99.1**	96.4 / 98.1	97.5 / 98.6	**99.6** / 98.9
Wooden Beads	91.1 / 90.5	91.4 / 98.1	85.0 / 97.4	82.5 / 97.5	91.9 / **98.6**	92.9 / 96.7	**94.5 / 98.6**
Woodstick	81.8 / 93.7	74.5 / 97.3	71.9 / 98.0	76.0 / 96.6	90.2 / 98.1	81.5 / 93.5	**91.1 / 98.6**
Zipper	99.3 / 90.5	**100.0** / 98.3	96.1 / 98.6	98.8 / 97.5	99.7 / 91.2	99.7 / 98.6	**100.0 / 98.7**
Average All	91.2 / 93.0	93.7 / 98.5	83.7 / 98.1	88.1 / 97.6	94.0 / 97.3	94.9 / 96.8	**97.0 / 98.9**

Experiment

Method	MvTec 3D-AD Dataset										
	Bagel	Cable Gland	Carrot	Cookie	Dowel	Foam	Peach	Potato	Rope	Tire	Mean
PatchCore	91.8	74.8	96.7	88.3	93.2	58.2	89.6	91.2	92.1	88.6	86.5
AST	98.3	87.3	97.6	97.1	93.2	88.5	97.4	98.1	**100.0**	79.7	93.7
M3DM	99.4	90.9	97.2	97.6	96.0	**94.2**	97.3	89.9	97.2	85.0	94.5
MMRD	**99.9**	94.3	96.4	94.3	**99.2**	91.2	94.9	90.1	99.4	90.1	95.0
MCFM	99.4	88.8	**98.4**	99.3	98.0	88.8	94.1	94.3	98.0	**95.3**	95.4
Ours	98.2	**96.1**	89.0	**99.9**	97.1	90.8	**99.5**	**98.8**	99.7	86.5	**95.6**

Method	Eyecandies Dataset										
	Candy Cane	Chocolate Cookie	Chocolate Praline	Confetto	Gummy Bear	Hazelnut Truffle	Licorice Sandwich	Lollipop	Marsh	Pepperming Candy	Mean
PatchCore	44.8	95.0	77.9	92.8	88.8	41.6	91.2	83.1	**100.0**	96.3	81.1
AST	58.7	84.6	80.7	83.3	83.3	54.3	74.4	87.0	94.6	83.5	78.4
M3DM	62.4	95.8	**95.8**	**100.0**	88.6	**78.5**	94.6	83.6	**100.0**	**100.0**	89.7
MCFM	68.0	93.1	95.2	88.0	86.5	78.2	91.7	84.0	99.8	96.2	88.1
MMRD	85.4	**100.0**	94.6	99.8	90.8	74.7	**96.6**	**98.4**	**100.0**	**100.0**	94.0
Ours	**88.6**	99.4	94.1	99.2	**94.2**	77.8	91.2	98.0	**100.0**	99.2	**94.2**

Thanks