기술과제 결과 보고서

2023. 9. 1

HYUNDAI
MOTOR GROUP

연구 과제명	Instance segmentation에서 생성 모델을 통한 오류 시각화 기술 개발
연구 수행자	윤지석
기술과제 수행기간	2023.02.14 ~ 2023.09.01

2023. 9. 1

Code & Slides
https://jsyoon.kr/230901

연구 수행자 : 윤지석
지도 연구원 : 이승용 파트장

HYUNDAI
MOTOR GROUP

■ 연구 개요

연구목표

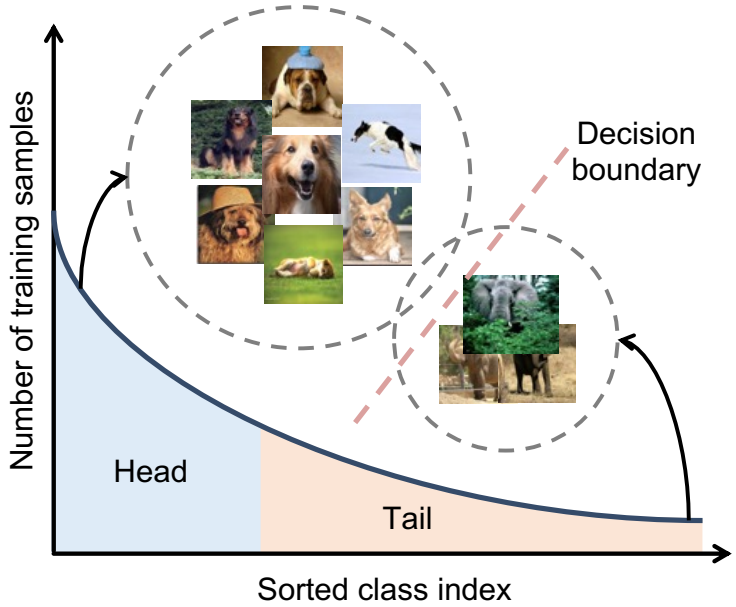
1. **Long-tailed** 환경에서의 **Explainable AI (XAI)** 기술 연구
2. Instance Segmentation (IS) **오류 시각화** 프레임워크 개발
3. XAI (Explainable AI) 기반 **data augmentation**으로 IS 성능 개선

- **Motivation 1 :** 연구실 환경에서 개발된 알고리즘 + Real-world 데이터 검증
  - XAI 연구는 주로 "clean"한 환경에서 진행 됨
    - Interpretability *vs.* generalization tradeoff[1]
  - 로보틱스랩에서 사용하는 데이터의 특성을 고려한 real-world application
    - Label imbalance, distributional shift (*e.g.,* covariate shift, concept shift)
- **Motivation 2 :** 각종 오류에 대한 이해와 원인 파악
  - Detection, segmentation, classification에서 발생하는 오류 시각화
  - 오류의 원인을 파악하고 개선 방안을 제시하여 모델 학습에 도움

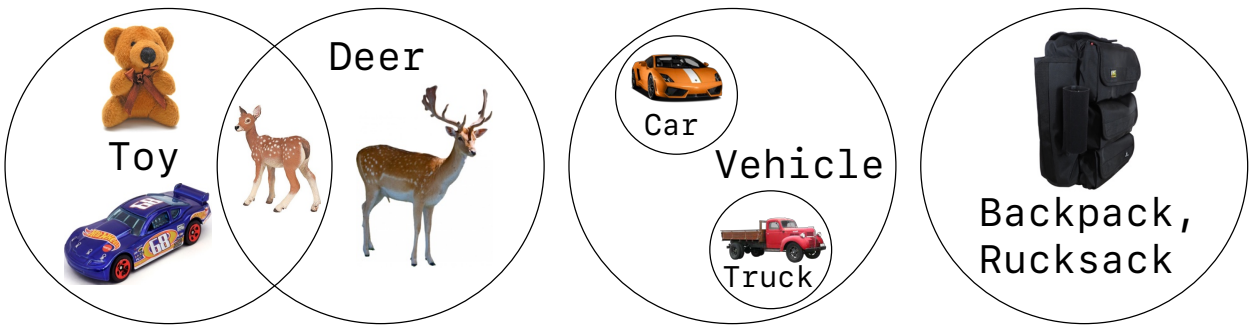
[1] "Trade-offs between Accuracy and Interpretability," *arXiv*, 2020

■ 연구 배경 - Long-tailed Instance Segmentation

Instance segmentation[2]



Long-tailed label distribution[3]



Category relationships[2]

[2] "LVIS: A Dataset for Large Vocabulary Instance Segmentation," *CVPR*, 2019

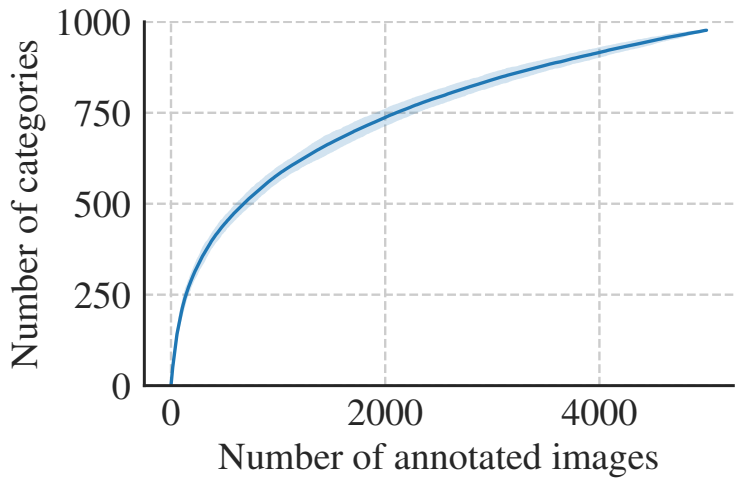
[3] "Deep Long-Tailed Learning: A Survey," IEEE-*TPAMI*, 2023

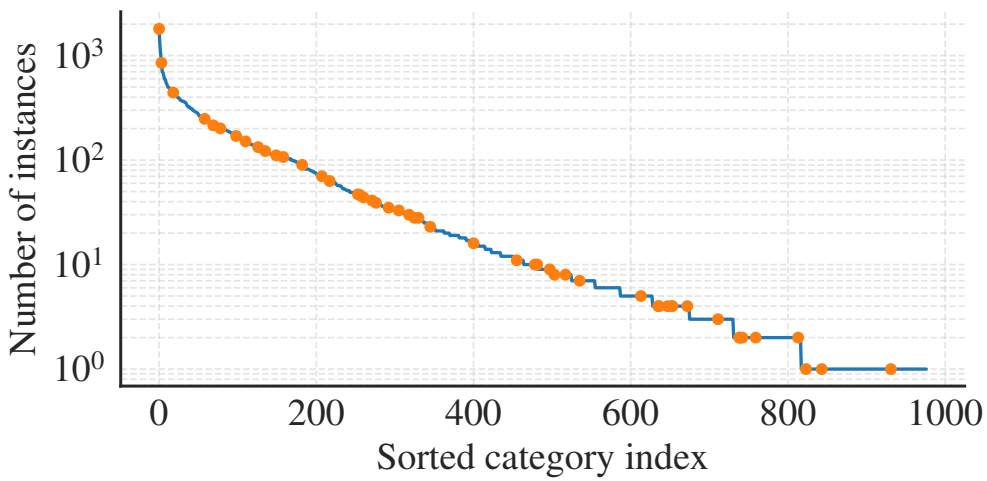
■ 데이터 소개: Large Vocabulary Instance Segmentation (LVIS)

[LVIS uses the COCO 2017 train, validation, and test image sets.]

LVIS[2] v1.0 Statistics (Rare: 1-10, Common: 11-100, frequent: 101-)

Set	Total	Divided by #image			Divided by #instance		
		rare	*common*	*frequent*	*rare*	*common*	*frequent*
Train	1,203	337	461	405	220	393	590
Validation	1,035	498	324	213	344	364	327





[본 기술과제의 모든 실험은 LVIS validation set을 사용했습니다.]

[2] "LVIS: A Dataset for Large Vocabulary Instance Segmentation," *CVPR*, 2019

■ Baseline Methods Review

- **Loss Re-weighting :** $L = -w_i \log(p_i)$
  - Sample 또는 category에 대한 가중치를 조정하여 bias을 완화하고, 분류 성능을 개선하기 위해 학습 중 빈도가 낮은 클래스에 더 많은 attention을 주는 방법.

- **Focal Loss :** $L = -(1-p_i)^{\gamma} \log(p_i)$
  - Loss re-weighting 방법 중 하나로, 예측을 기반으로 샘플에 adaptive weighting을 할당합니다. Head 및 tail class에 적절한 weighting을 가해주기 때문에 long-tailed 문제에 많이 사용 됩니다.

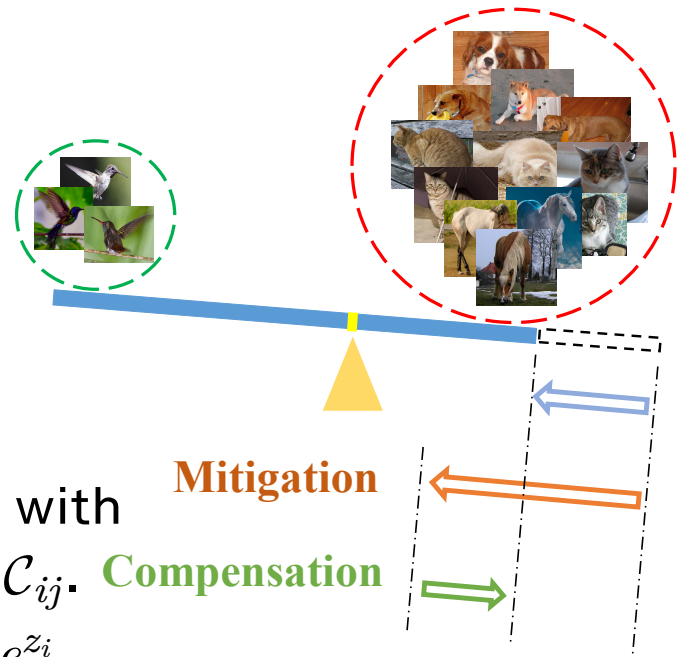
- **Class-aware Margin Loss :** $L = -\log e^{y_{iz}-\Delta_z} / (e^{y_{iz}-\Delta_z} + \sum_{c \neq z} e^{y_{ic}})$
  - Loss 계산에 클래스별 margin을 도입하여 적은 빈도의 클래스에 더 큰 margin을 할당하여 제한된 데이터로 더 나은 generalization를 촉진합니다 ($\Delta_j = C/N_j^{1/4}$).

■ Baseline Methods Review: Gradient-based Method[4]

Cross-entropy loss heavily penalize negative samples of tail classes

$$L_{ce}(\mathbf{z}) = -\sum_{i=1}^{C} y_i \log\left(\sigma_i\right), \text{ with } \sigma_i = \frac{e^{z_i}}{\sum_{j=1}^{C} e^{z_j}}$$

$$\frac{\partial L_{ce}(\mathbf{z})}{\partial z_i} = \sigma_i - 1 \qquad \frac{\partial L_{ce}(\mathbf{z})}{\partial z_j} = \sigma_j$$



Seesaw Loss[4] dynamically re-balance gradients with mitigation factor $\mathcal{M}_{ij}$ and compensation factor $\mathcal{C}_{ij}$.

$$L_{\text{seesaw}}\ (\mathbf{z}) = -\sum_{i=1}^{C} y_i \log\left(\widehat{\sigma}_i\right), \quad \text{with } \widehat{\sigma}_i = \frac{e^{z_i}}{\sum_{j\neq i}^{C} \mathcal{S}_{ij} e^{z_j} + e^{z_i}}, \quad \mathcal{S}_{ij} = \mathcal{M}_{ij} \cdot \mathcal{C}_{ij}$$

$$\mathcal{M}_{ij} = \begin{cases} 1, & \text{if } N_i \leq N_j \\ \left(\frac{N_j}{N_i}\right)^p, & \text{if } N_i > N_j \end{cases} \qquad \mathcal{C}_{ij} = \begin{cases} 1, & \text{if } \sigma_j \leq \sigma_i \\ \left(\frac{\sigma_j}{\sigma_i}\right)^q, & \text{if } \sigma_j > \sigma_i \end{cases}$$

[4] "Seesaw Loss for Long-Tailed Instance Segmentation," *CVPR*, 2021

■ Baseline Results

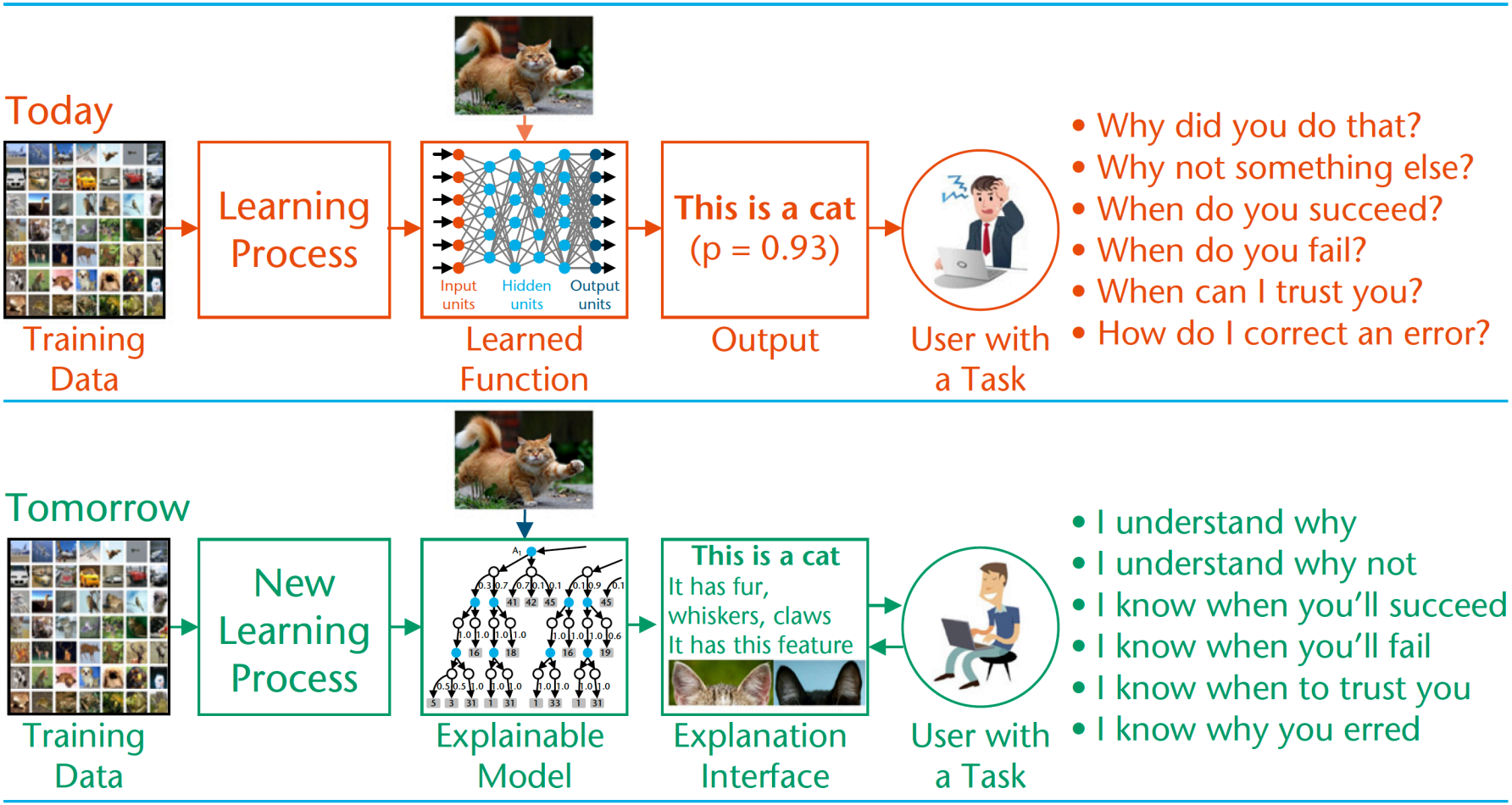
- Trained with Mask R-CNN with ResNet50-FPN backbone.
- **Seesaw loss** - Selected as the backbone model.

Table: AP_r, AP_c, AP_f are mask APs for rare, common, and frequent classes. AP^b denote bounding box AP.

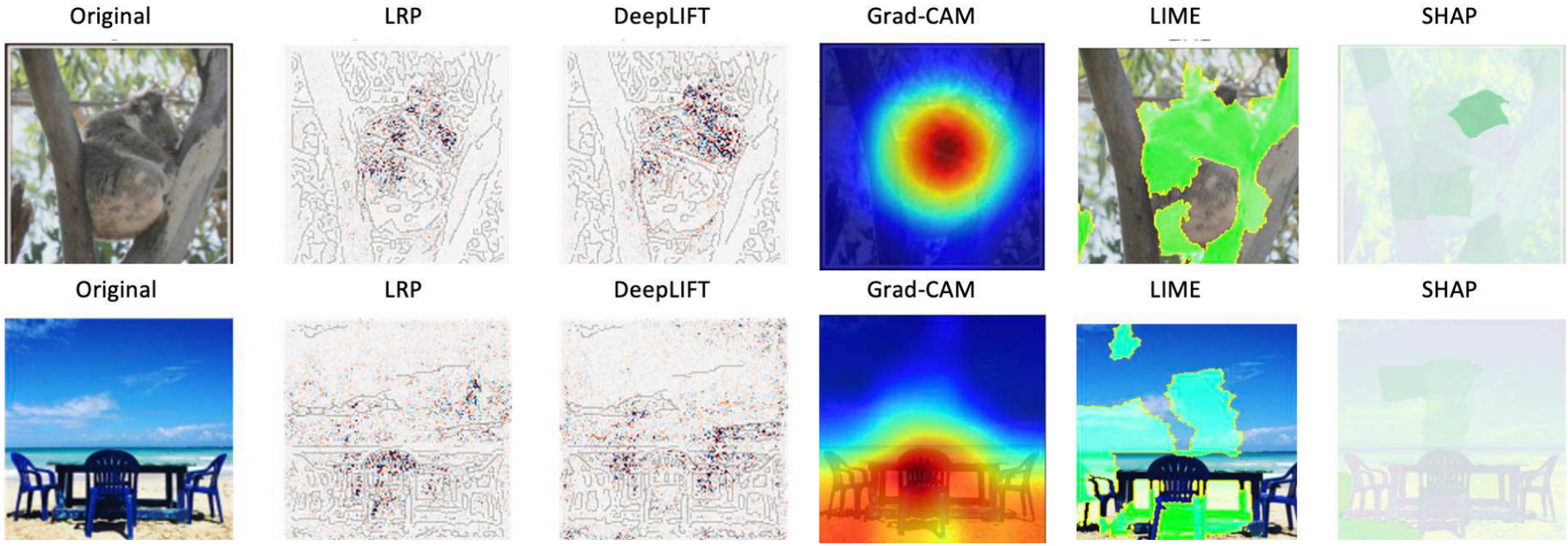
Model	AP_r	AP_c	AP_f	AP	AP^b_r	AP^b_c	AP^b_f	AP^b
r50	3.4	19.7	28.1	20.4	3.0	18.7	29.7	20.5
CM	8.1	20.9	25.2	20.7	6.5	20.2	26.1	20.4
LR	13.5	22.1	24.2	21.6	11.0	21.2	24.6	21.1
FL	12.1	20.6	25.6	21.2	10.2	20.7	26.7	21.4
SS	18.7	26.3	31.7	27.1	19.3	28.9	34.4	28.7

(r50: Baseline, CM: Class-aware Margin, LR: Loss Re-weighting, FL: Focal Loss, SS: Seesaw Loss)

■ XAI: Understanding, "Visualizing" and Interpreting Deep Models



[5] "DARPA's Explainable Artificial Intelligence Program," *AI Magazine*, 2019



- 기존의 오류 시각화 기술:
  - **Backpropagation-based:** 입력값에 대해 backpropagation을 통한 오차의 정도를 계산하여 각 픽셀의 중요도를 나타내는 방법 (*e.g.*, LRP, DeepLift)
  - **Activation-based:** Activation의 선형결합을 활용해 region of importance를 파악하는 방법 (*e.g.*, Saliency, CAM)
  - **Perturbation-based:** 입력값에 대한 작은 변화(purtubation)에 따라 예측값이 어떻게 변하는지를 통해서 중요도를 파악하는 방법 (*e.g.*, LIME, SHAP)

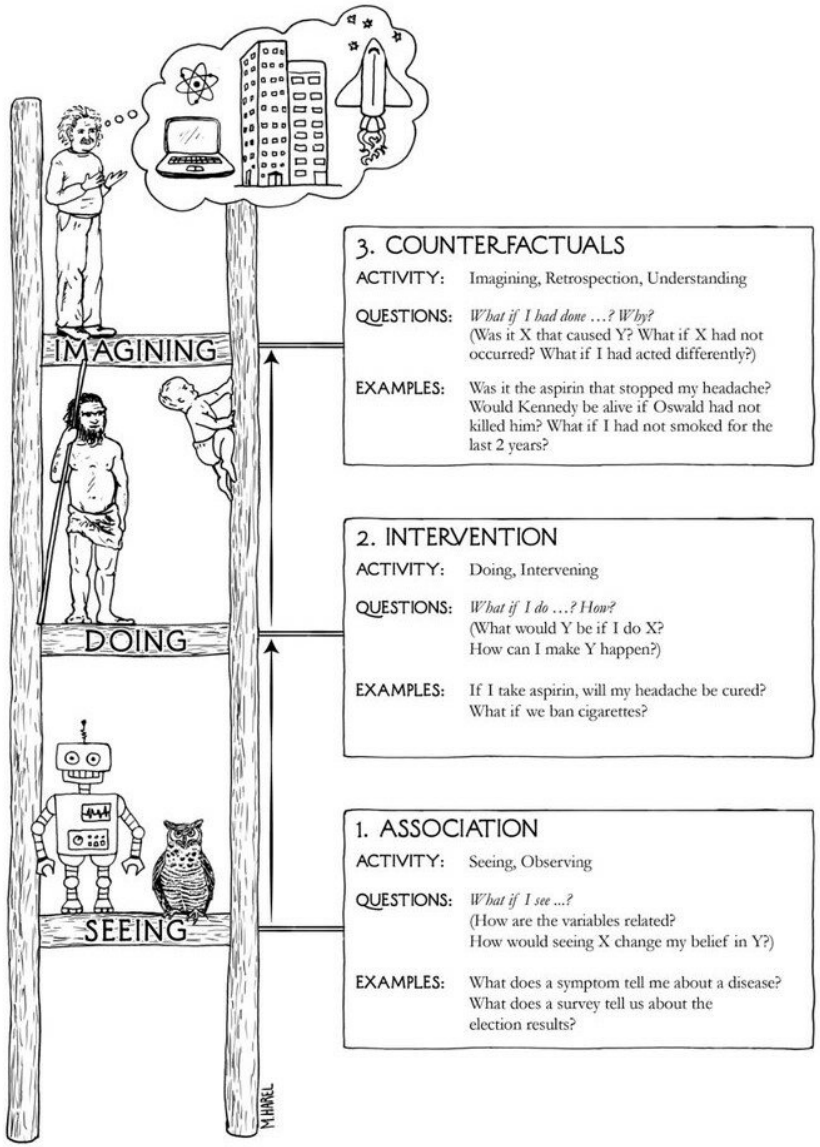
[6] "Opportunities and Challenges in Explainable Artificial Intelligence (XAI): A Survey," *arXiv*,

■ Pearl's Causal Hierarchy – Counterfactual Reasoning

- **Counterfactual:**
  What if I had done?, Why?
  가상의 상황을 생각해내는 인간의 고차원적인 사고 과정

- **Intervention:** *What if I do?*
  (예: Perturbation-based)

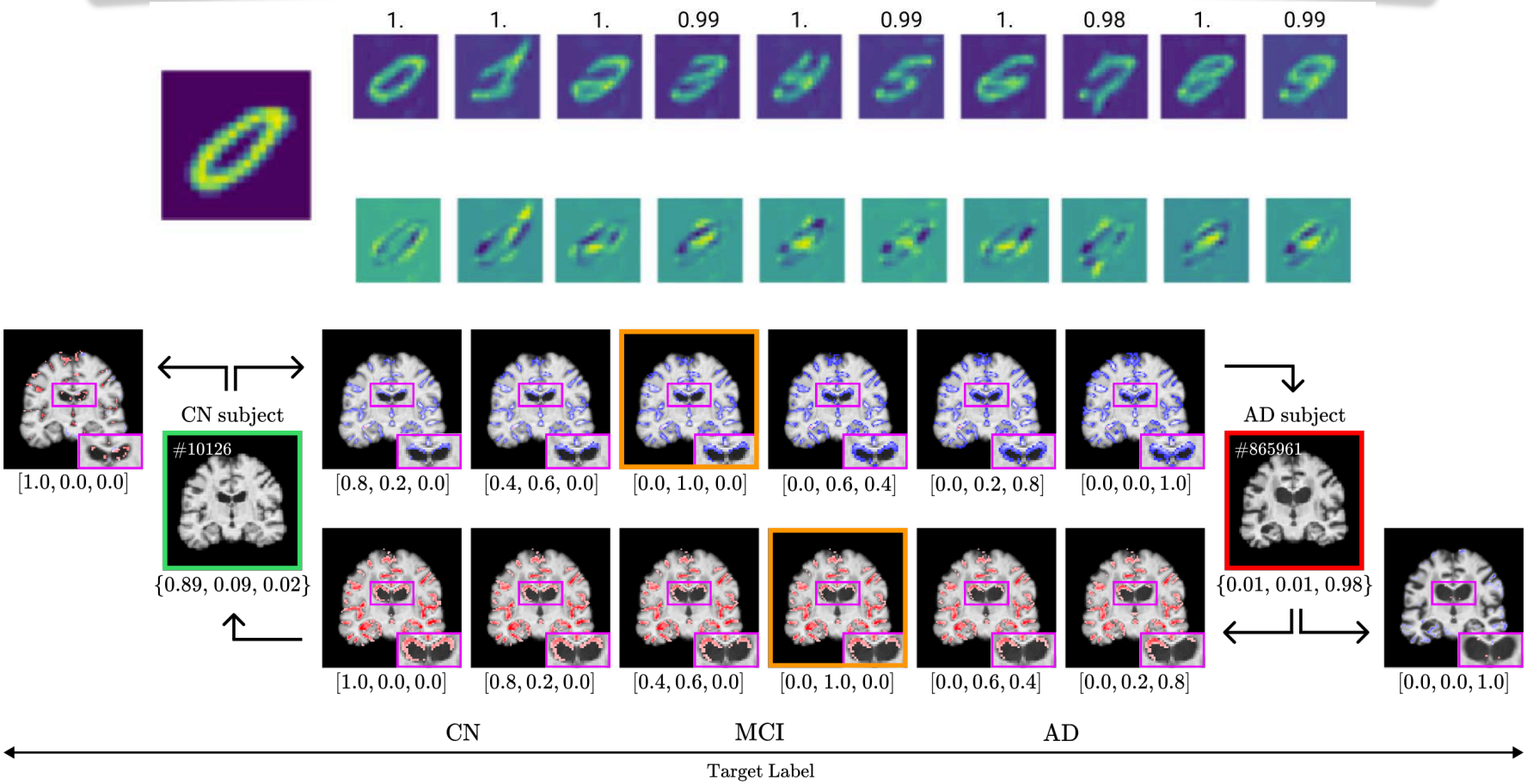
- **Association:** *What if I see?*
  (예: Activation-based)



[7] "The Book of Why: The new science of cause and effect," Pearl and Mackenzie, 2009

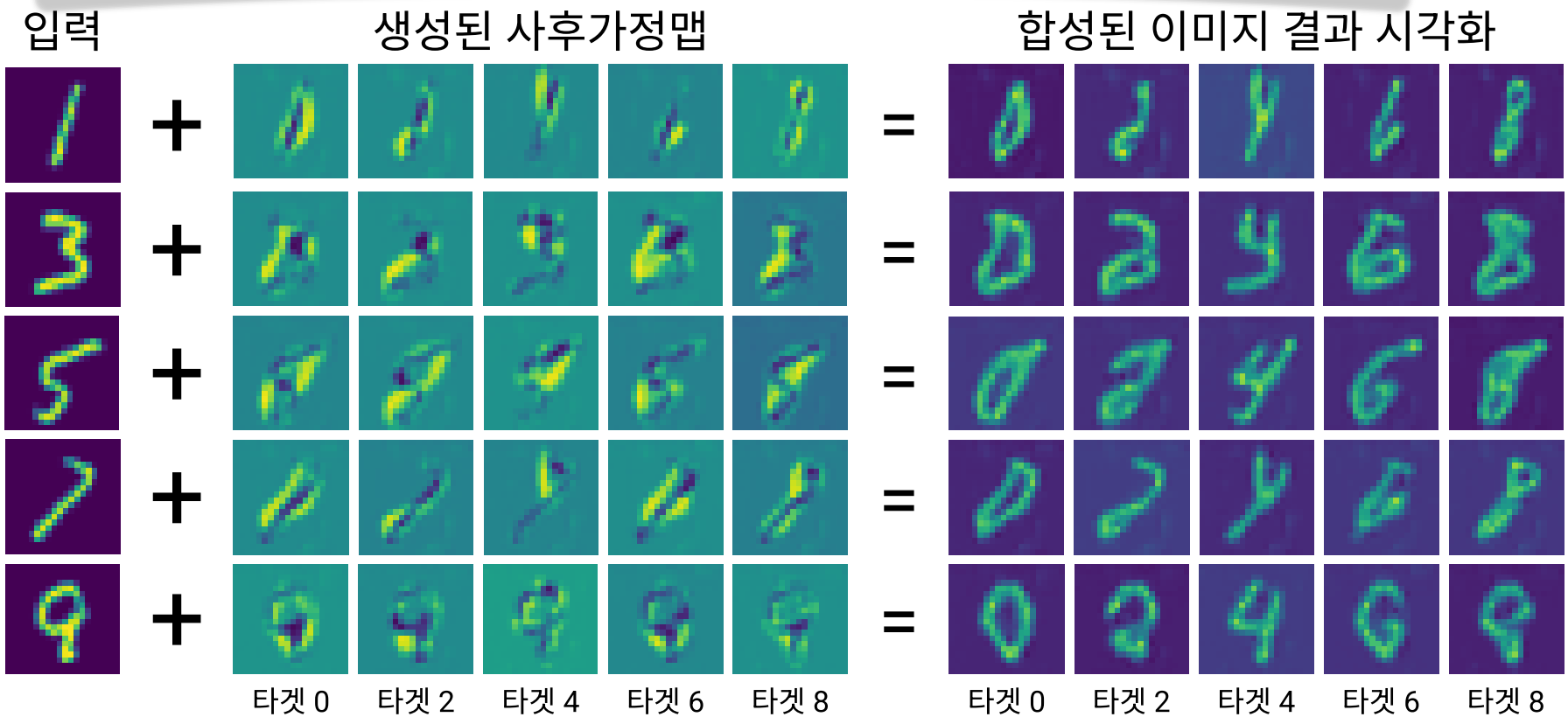
■ Explanation using Counterfactual Maps[5]

"For situation X, why was the outcome Y and not Z?"
"Had X been $\tilde{X}$, then the outcome would have been Z"

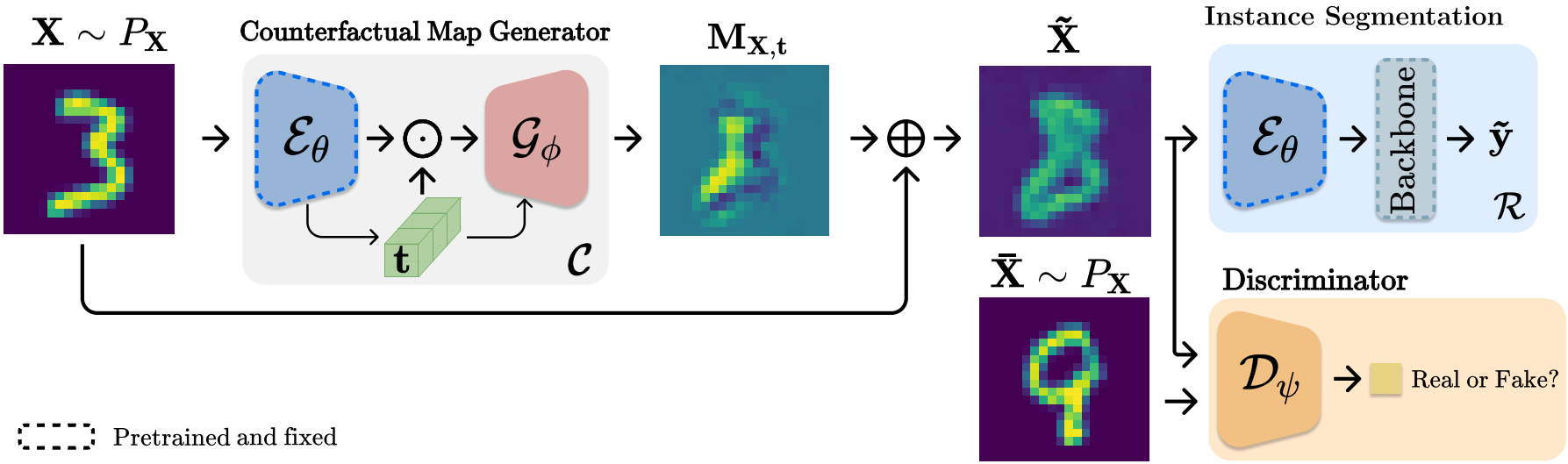


[5] "Learn-Explain-Reinforce: Counterfactual Reasoning and Its Guidance to Reinforce an Alzheimer's Disease Diagnosis Model," IEEE-*TPAMI*, 2023

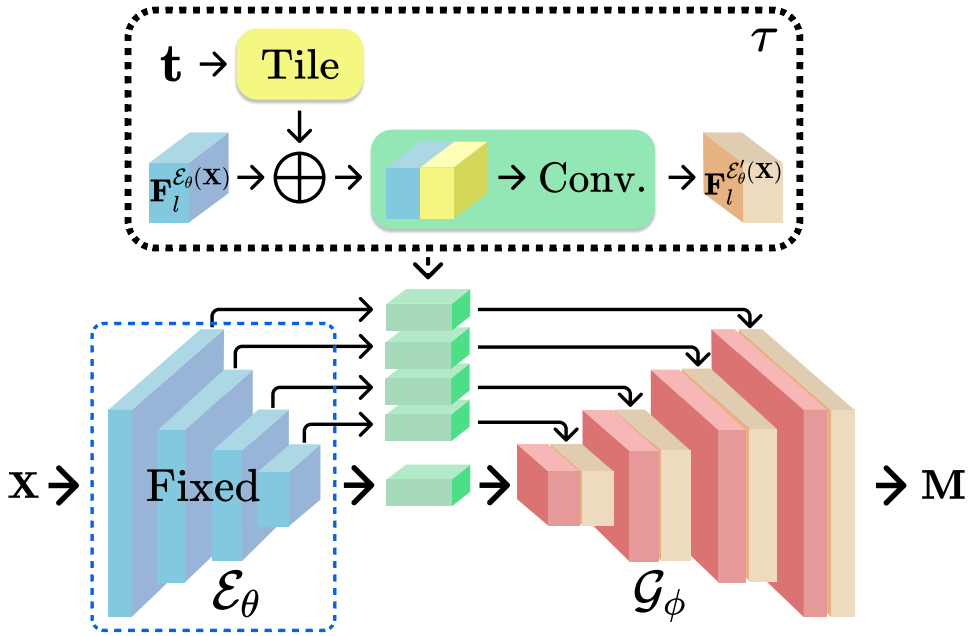
1. 타겟 factor에 대한 intervention →
다른 factor of variation은 동일하게 유지
2. Style transfer 과 유사하지만→
Style은 factor of variation을 유지하는 기능이 없음



■ Proposed Framework



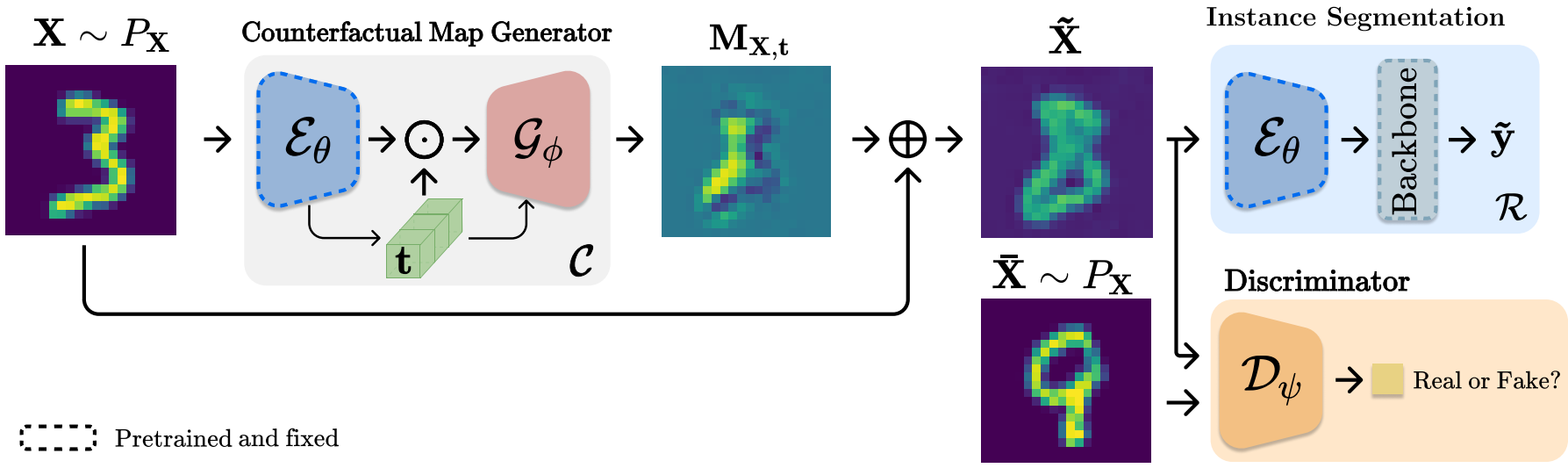
- 간단한 GAN 구조를 사용하여 instance segmentation target label에 해당하는 counterfactual map을 생성함.
  - 각 backbone network 및 target $\mathbf{t}$ 에대한 counterfactual map $\mathbf{M}_{\mathbf{X},\mathbf{t}}$ 생성하여 오류 시각화.
- Backbone network에 Mask R-CNN의 다양한 모듈을 추가하여 각 factor에 대한 counterfactual을 만들 수 있음
  - *E.g.*, classifier $\mathcal{L}_{cls}$, bounding box loss $\mathcal{L}_{box}$, mask loss $\mathcal{L}_{mask}$



Detailed view of the Counterfactual Map Generator

$$\mathcal{T}(\mathbf{X}, \mathbf{t}) = \left\{ \mathbf{F}_1^{\mathcal{E}'_\theta(\mathbf{X})}, \ldots, \mathbf{F}_L^{\mathcal{E}'_\theta(\mathbf{X})} \right\} \quad \mathbf{M}_{\mathbf{X},\mathbf{t}} = \mathcal{G}_\phi(\mathcal{T}(\mathbf{X}, \mathbf{t})) \quad \tilde{\mathbf{X}} = \mathbf{X} + \mathbf{M}_{\mathbf{X},\mathbf{t}}$$

- Fixed backbone network를 finetuning 시켜서 generator을 학습함.



$$\mathcal{L}_{\text{cyc}} = \mathbb{E}_{\mathbf{X}\sim P_{\mathbf{X}},\mathbf{t}\sim U(0,|\mathbf{y}|)}\left[\|\mathbf{X}' - \mathbf{X}\|_1\right], \mathcal{L}_{\text{map}} = \mathbb{E}_{\mathbf{X}\sim P_{\mathbf{X}}}\left[\lambda_1 \|\mathbf{M}_{\mathbf{X},\mathbf{t}}\|_1 + \lambda_2 \|\mathbf{M}_{\mathbf{X},\mathbf{t}}\|_2\right]$$

$$\mathcal{L} = \lambda_3 \mathcal{L}_{\text{adv}}^{\mathcal{G}_\phi} + \lambda_4 \mathcal{L}_{\text{adv}}^{\mathcal{D}_\psi} + \lambda_5 \mathcal{L}_{\text{cyc}} + \lambda_6 \mathcal{L}_{\text{bck}} + \mathcal{L}_{\text{map}}$$

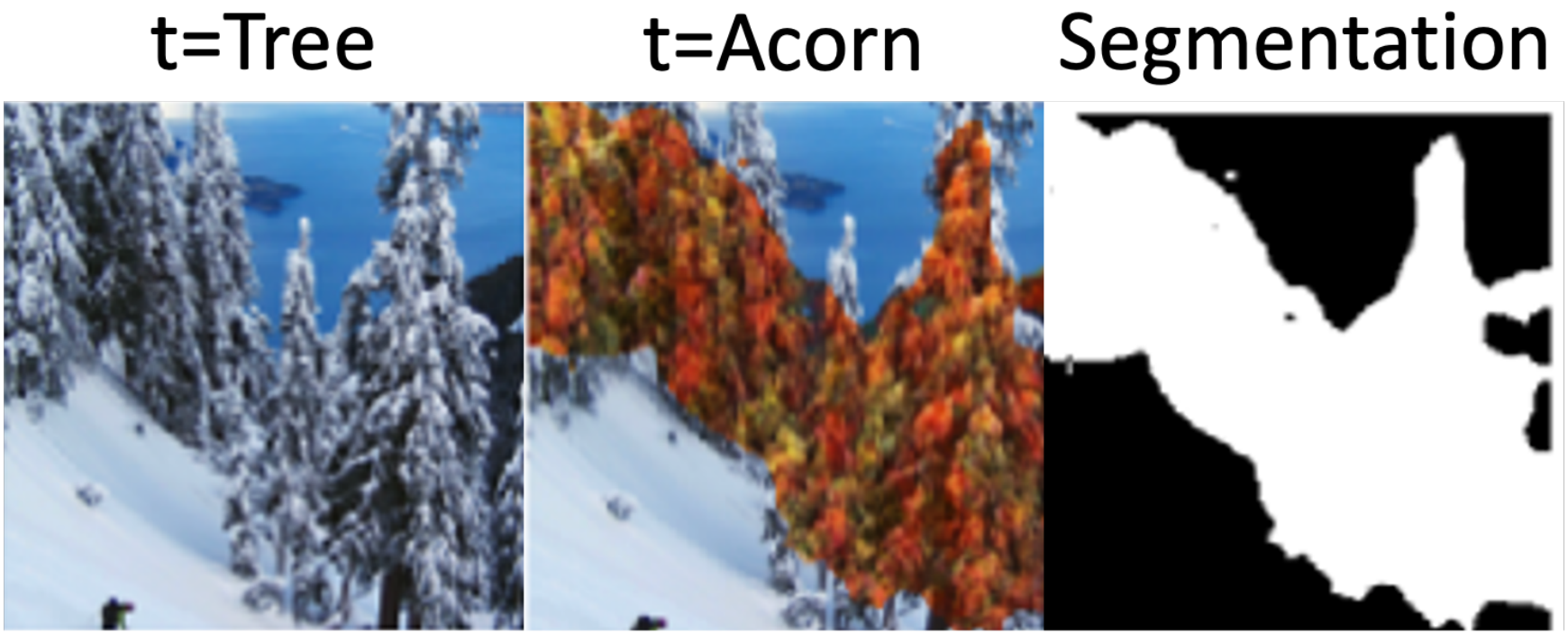
$\mathcal{L}_{\text{bck}}$: Backbone loss

- Target에 따라 $\mathcal{L}_{\text{bck}}$를 아래 Mask R-CNN loss 중 하나로 설정함:
  - Classifier $\mathcal{L}_{\text{cls}}$, bounding box loss $\mathcal{L}_{\text{box}}$, mask loss $\mathcal{L}_{\text{mask}}$
- $\mathcal{E}_\theta$는 pretrained된 ResNet50-FPN이며, backbone은 각 target loss에 해당하는 downstream network 사용.

■ 정성 평가 – Target을 class로 설정했을 때 counterfactual (i.e., $\mathcal{L}_{\text{bck}} := \mathcal{L}_{\text{cls}}$)

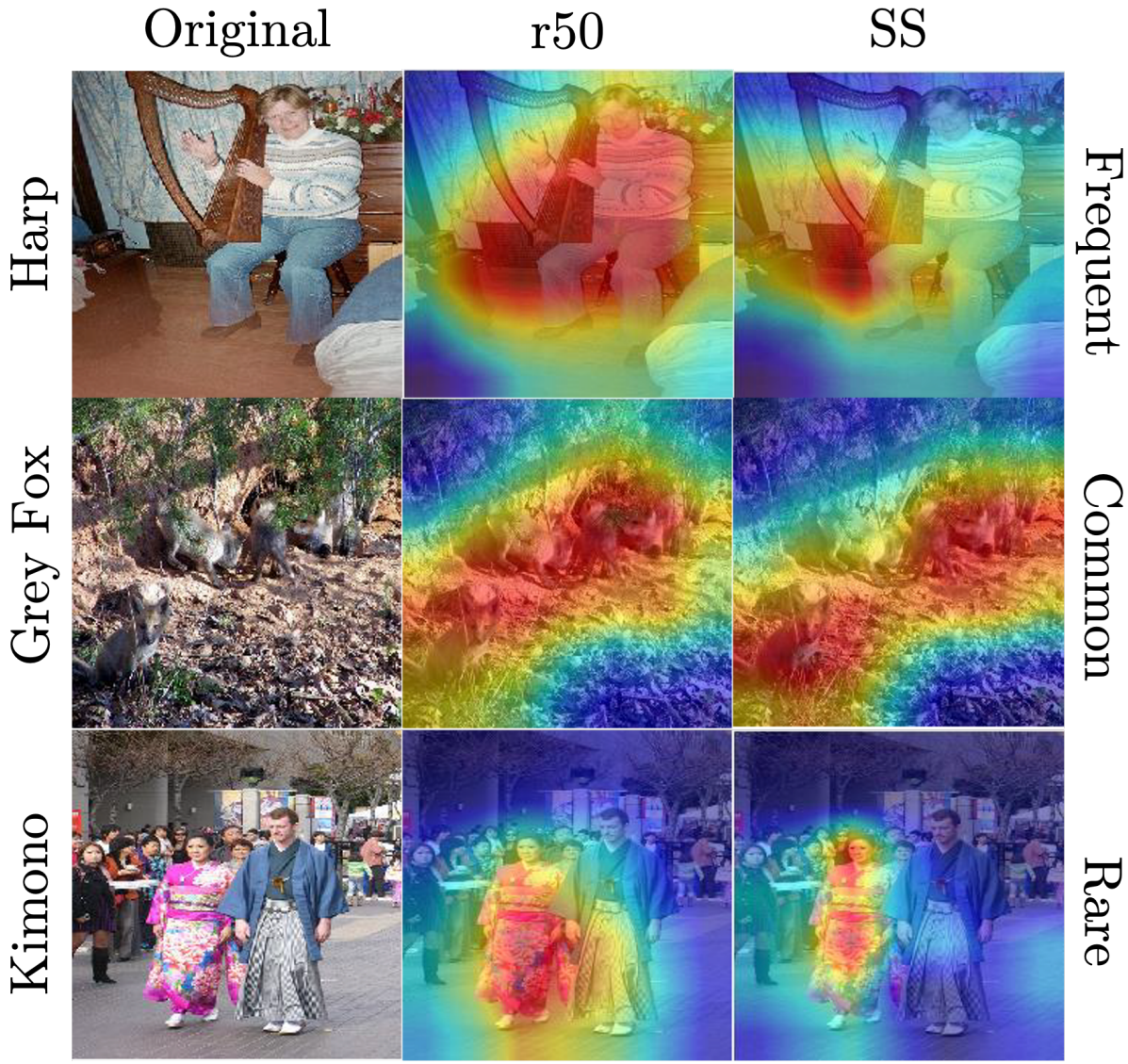
- $\mathrm{t} = \{\text{Grass, English springer}\} \rightarrow \mathrm{t} = \{\text{Snow, English springer}\}$ 으로 지정했을 때 결과

■ 정성 평가 – Segmentation에 끼치는 영향



- 클래스를 타겟으로 지정했을때, segmentation또한 어느정도 일치하는것으로 보임 (RoI에 대한 counterfactual 이기 때문).
- 즉, semantics에 대한 정보로 localization이 되는것을 보여줌.

■ 오류 시각화 – Baseline (r50) vs. Seesaw Loss



■ Counterfactual Image를 통한 Data Augmentation

LVIS-CF Augmentated Dataset Statistics

Set	Total	Image Count		
		rare	*common*	*frequent*
LVIS	360,952	1,496	17,998	341,458
LVIS-CF	402562	32,541	28,563	341,458

- 생성한 counterfatual image로 data augmentation
  - 모든 class에 최소 101개의 영상이 있도록 counterfactual image 생성
  - 랜덤 영상 선택 후 counterfactual image 생성함
- Evaluation Metrics
  - Inception Score (IS): Classifier의 label distribution entropy를 통해 계산
  - Frechet Inception Distance (FID): 실제 영상과 생성된 영상의 distribution 비교

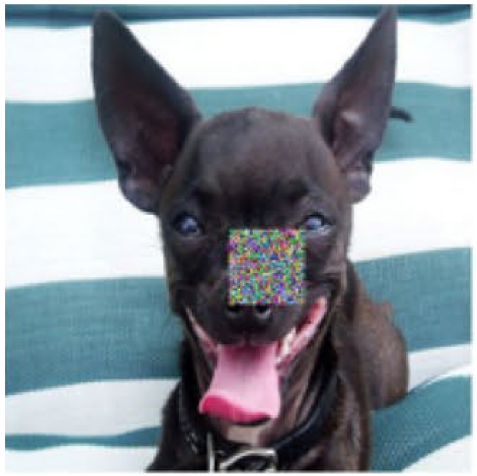
- Baseline Methods:
  - Baseline: Trained with original LVIS dataset with Seesaw loss
  - Image Transformation: Randomly applied 16 image factors by strength
    - 12 photometric factors (Brightness, Contrast, Color, Sharpness, Auto-Contrast, Invert, Equalize, Solarize, SolarizeAdd, Posterize, NoiseSalt, NoiseGaussian)
    - 4 geometric factors (Shear-X, Shear-Y, Rotate, Flip)

Model	IS	FID
Img. Trans.	6.59	402.7
Proposed	3.69	249.91

Model	AP_r	AP_c	AP_f	AP	AP^b_r	AP^b_c	AP^b_f	AP^b
Baseline	18.7	26.3	31.7	27.1	19.3	28.9	34.4	28.7
Img. Trans.	19.3	26.1	31.8	29.3	20.1	29.1	33.3	29.9
Proposed	22.0	29.1	31.9	31.1	25.1	32.3	35.0	30.0

■ Discussion & Conclusion

- Counterfactual image를 통해 오류 시각화 및 data augmentation 기술
  - Baseline 대비 mask AP 약 15% 성능 향상, bounding box AP 약 5% 성능 향상
- **[TODO]** 타겟을 bounding box나 segmentation에 적용
  - Implementation setting을 변경해야함
  - Target에 대한 데이터가 없음 (예: 강아지 mask $\rightarrow$ 어떤 mask 로 변경할지?)



Preliminary experiment with $\mathcal{L}_{\text{bck}} := \mathcal{L}_{\text{mask}}$.
Mask에 random class 를 넣음.

감사합니다

(Q & A)

wltjr1007 (AT) korea.ac.kr

https://milab.korea.ac.kr
https://jsyoon.kr

HYUNDAI
MOTOR GROUP