2025 여름 세미나

Image Inpainting & Egocentric Hand Generation

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

Presented By
MinSuh Song

Outline

- Wensong Song, Hong Jiang et al. **"Insert Anything: Image Insertion via In-Context Editing in DiT."** arxiv 2025

- Junho Park, Andrew Sangwoo Ye et al. **"EgoWorld: Translating Exocentric View to Egocentric View using Rich Exocentric Observations."** arxiv, 2025

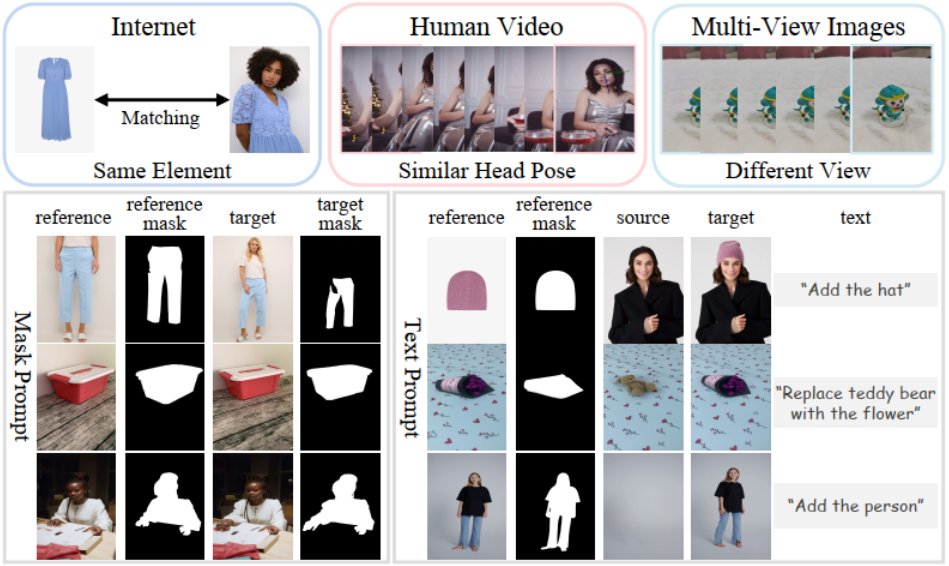
Insert Anything: Image Insertion via In-Context Editing in DiT

Insert Anything

- Introduction
  - 기존 Diffusion 기반 모델은 이미지 편집에 큰 성과를 보이지만, 대부분 단일 task에만 집중되어 있다는 문제점 존재
    - Task Specific Focus: 범용성이 떨어짐
    - Fixed Control Mode: Mask 또는 text 중 하나에 대해서만 지원
    - Inconsistent Visual-Reference Harmony: 삽입된 요소의 시각적 이질감, feature 손실 발생
  - 본 논문은 아래와 같은 contribution을 지님
    - 다양한 inpainting task를 학습하기 위해 AnyInsertion이라는 대규모 데이터셋 구축
    - Mask prompt 혹은 text prompt를 guidance로 활용하여 Diffusion Transformer (DiT)의 multimodal attention 수행
    - In-context editing mechanism을 도입하여 inpainting 대상과 배경 사이의 자연스러운 조화

Insert Anything

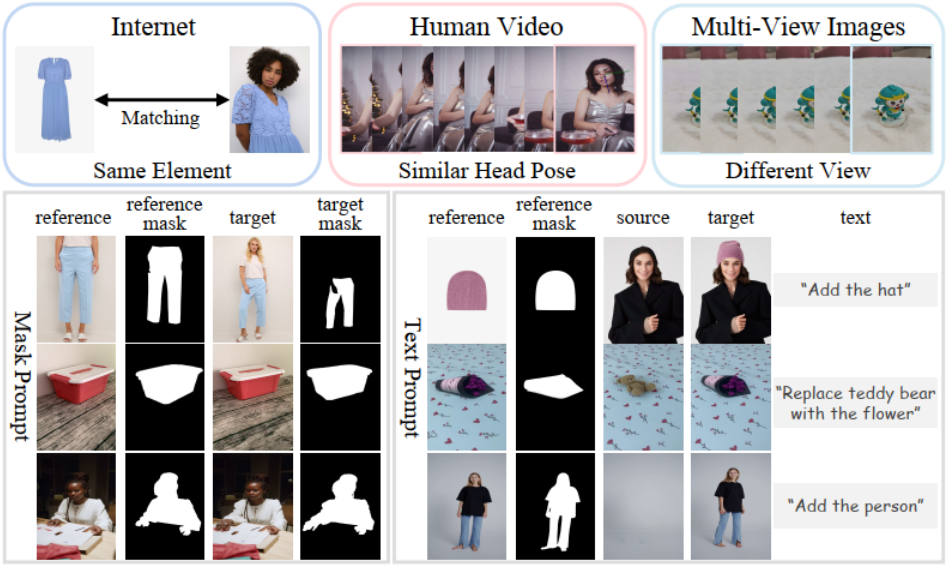
- AnyInsertion Dataset
  - 본 논문의 학습 및 성능 평가를 위해 새롭게 제시한 데이터셋
    - Insert Anything의 학습에 사용되는 데이터
      - Mask prompt
        - ✓Insert할 물체가 존재하는 reference image와 해당 물체의 segmentation mask
        - ✓해당 물체를 insert할 target 이미지와 해당 물체의 segmentation mask



< AnyInsertion Dataset 구성 >

Insert Anything

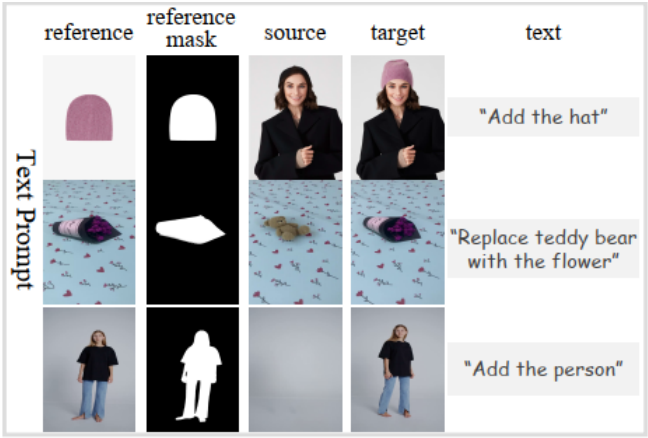
- AnyInsertion Dataset
  - 본 논문의 학습 및 성능 평가를 위해 새롭게 제시한 데이터셋
    - Insert Anything의 학습에 사용되는 데이터
      - Text prompt
        - ✓Insert할 물체가 존재하는 reference image와 해당 물체의 segmentation mask
        - ✓Source image와 text prompt의 설명 대로 만들어진 GT target image



< AnyInsertion Dataset 구성 >

Insert Anything

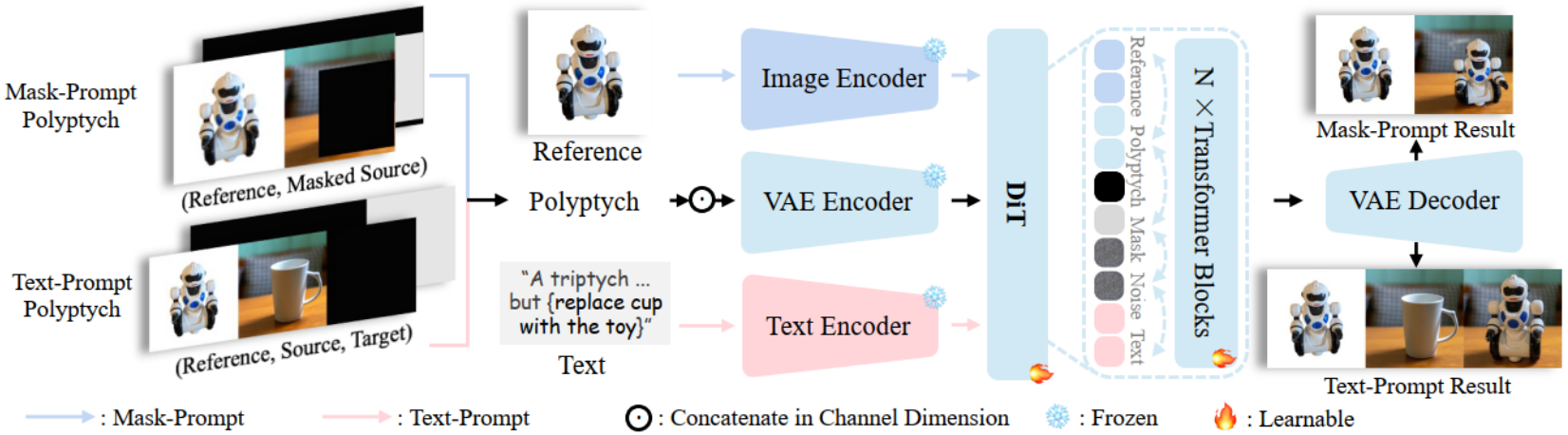
- AnyInsertion Dataset
  - 본 논문의 학습 및 성능 평가를 위해 새롭게 제시한 데이터셋
    - Insert Anything의 학습에 사용되는 데이터
      - Text prompt
        - ✓Reference image로부터 Segment Anything 모델을 활용하여 segmentation mask 추출
        - ✓Target image로부터 DesignEdit 모델을 활용하여 목표 물체를 삭제해서 source image 생성
        - ✓Text prompt는 “Replace/Add [source] with [reference]” 라는 template을 써서 마련함



< Text prompt tuple 구성 >

Insert Anything

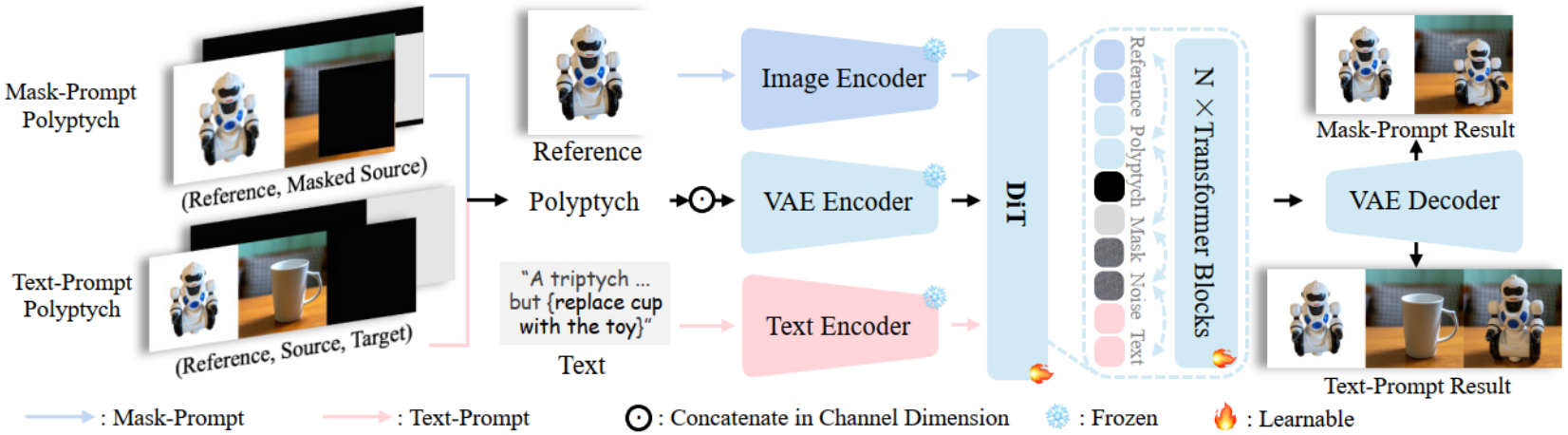
- Insert Anything Flow
  - Image와 mask 및 text prompt를 이용하여 DiT의 multimodal attention 수행
    - Insert Anything은 총 3개의 key input들이 필요함
      - Insert할 물체가 존재하는 reference image
      - Background context를 제공하는 물체가 삽입될 source image
      - Control prompt (mask or text)



< Insert Anything 모델 구조도 >

Insert Anything

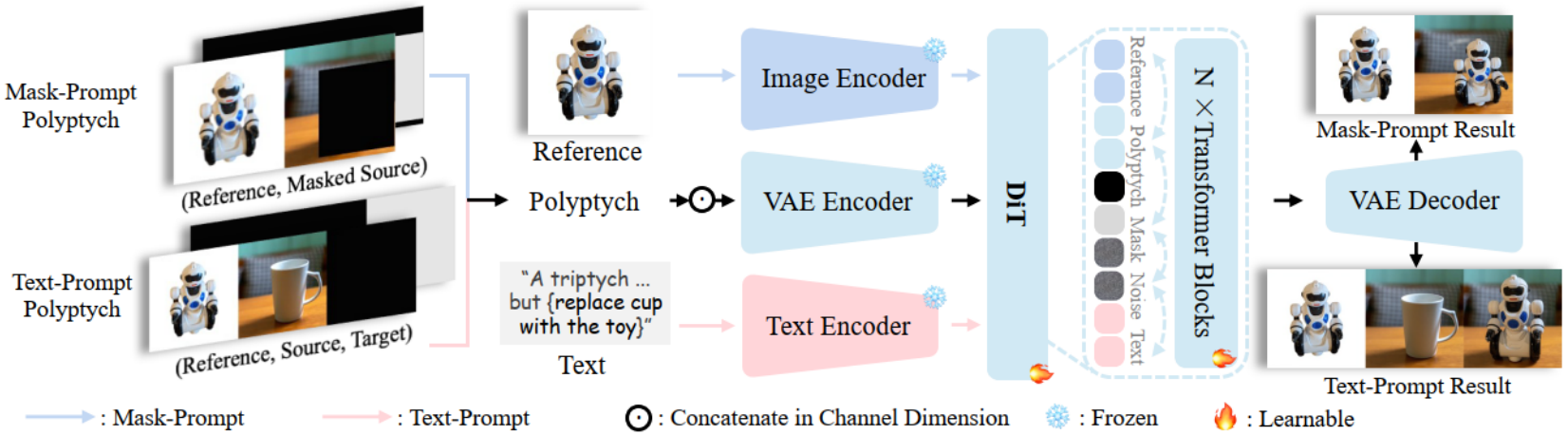
- Insert Anything Flow
  - Image와 mask 및 text prompt를 이용하여 DiT의 multimodal attention 수행
    - In-context editing
      - 목표 물체를 inpainting 하는 과정에서, reference image의 object와 target image의 background가 contextual relationship을 유지하는 방향으로 학습
      - 먼저 reference image에서 background removal 작업 진행 -> 물체만 남김
        - ✓Grounding-DINO, Segment Anything
      - 이후 mask-prompt diptych와 text-prompt triptych로 나뉨



< Insert Anything 모델 구조도 >

Insert Anything

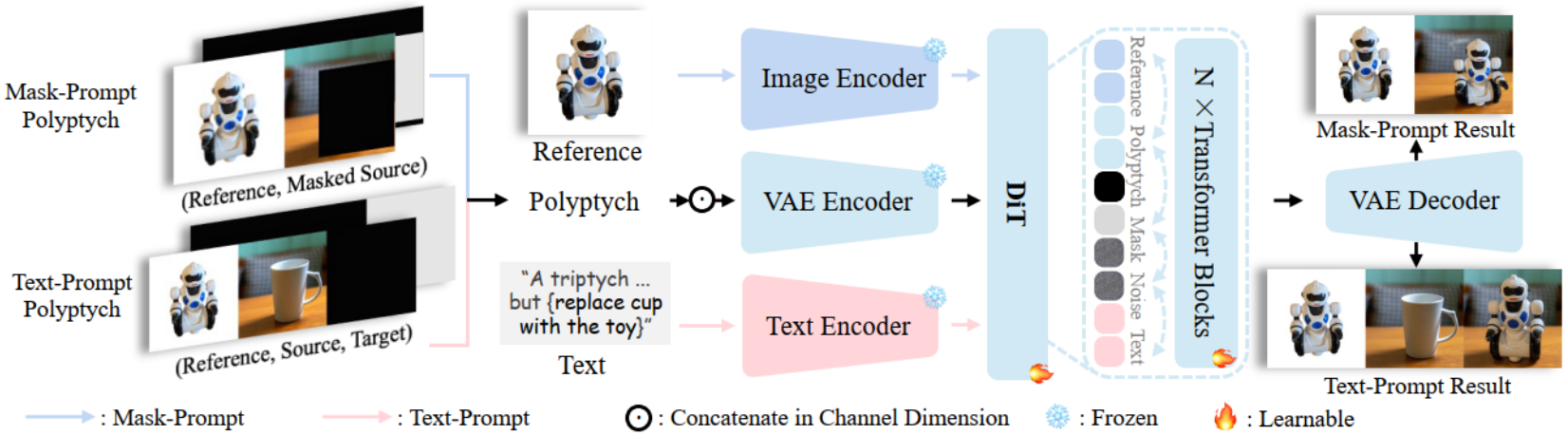
- Insert Anything Flow
  - Image와 mask 및 text prompt를 이용하여 DiT의 multimodal attention 수행
    - Mask-prompt diptych
      - Diptych: segment된 reference image와 부분 masking된 source image가 concatenate된 two panel structure
      - $I_{diptych} = [R_{seg}(I_{ref}); I_{masked_{src}}]$
      - Reference image의 물체 부분은 0으로, masked source image의 삽입될 부분은 1로 처리
      - 그 결과 target region에 inpainting 될 때 spatial guidance로 활용



< Insert Anything 모델 구조도 >

Insert Anything

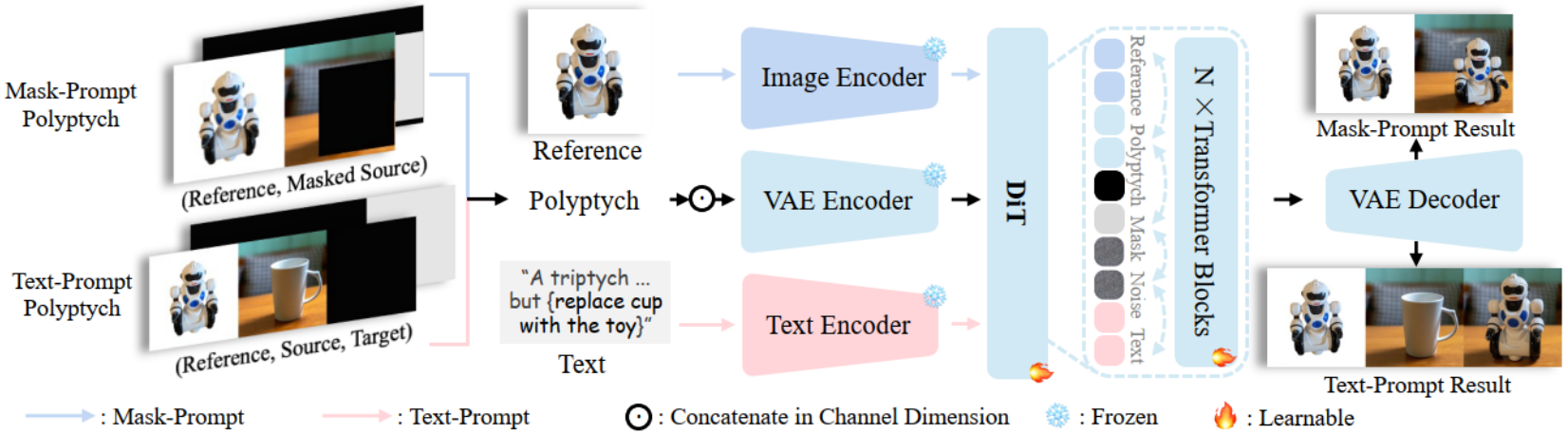
- Insert Anything Flow
  - Image와 mask 및 text prompt를 이용하여 DiT의 multimodal attention 수행
    - Text-prompt triptych
      - Triptych: segment된 reference image와 source image와 fully-masked된 three panel structure
      - $I_{triptych} = [R_{seg}(I_{ref}); I_{src}; \emptyset]$
      - Reference image의 물체 부분은 0으로, source image 또한 0, generate되야 하는 masked region은 1로 처리



< Insert Anything 모델 구조도 >

Insert Anything

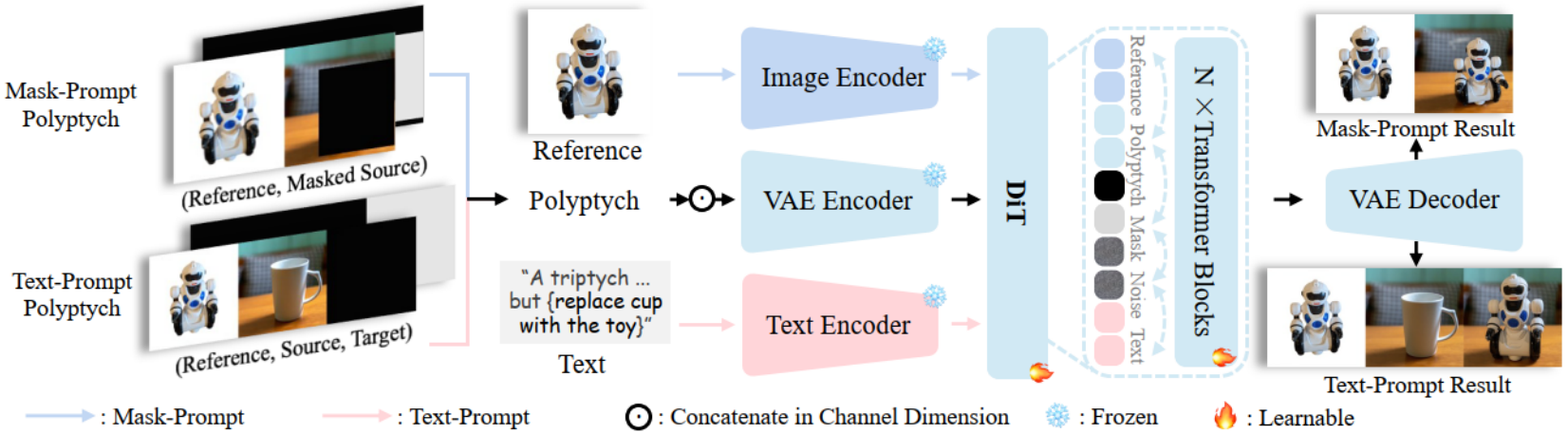
- Insert Anything Flow
  - Image와 mask 및 text prompt를 이용하여 DiT의 multimodal attention 수행
    - 각 encoder를 통해 DiT의 input으로 작용
    - Mask-Prompt
      - Mask-prompt diptych는 VAE encoder를 통해 DiT model의 image branch로 전달
      - Reference image는 CLIP image encoder를 통해 DiT model의 text branch로 전달
        - ✓Inpainting 과정에서 contextual guidance로 작용



< Insert Anything 모델 구조도 >

Insert Anything

- Insert Anything Flow
  - Image와 mask 및 text prompt를 이용하여 DiT의 multimodal attention 수행
    - Text-Prompt
      - Mask-prompt triptych는 VAE encoder를 통해 DiT model의 image branch로 전달
      - Text prompt는 CLIP text encoder를 통해 DiT model의 text branch로 전달
      - Text prompt template
        - ✓"A triptych with three side-by-side images. On the left, is a photo of [label]; on the right, the scene is exactly the same as in the middle but [instruction] on the left."



< Insert Anything 모델 구조도 >

Insert Anything

- Experiments and Results
  - Implementation details
    - DiT model: FLUX.1 Fill
    - LoRA rank: 256
    - Batch size: 8
    - Output image resolution: 768*768 pixels
    - Prodigy optimizer
    - 4 NVIDIA A800 GPU (80GB)
  - Evaluation metrics
    - Peak Signal-to-Noise Ratio (PSNR)
    - Structural Similarity Index (SSIM)
    - Learned Perceptual Image Path Similarity (LPIPS)
    - Frechet Inception Distance (FID)

Insert Anything

- Experiments and Results
  - Quantitative comparison on object insertion
    - Datasets
      - AnyInsertion, DreamBooth, VTON-HD

Methods	AnyInsertion (Object)				DreamBooth			
	PSNR ↑	SSIM ↑	LPIPS ↓	FID ↓	PSNR ↑	SSIM ↑	LPIPS ↓	FID ↓
AnyDoor [3]	21.39	0.7648	0.1831	67.99	16.68	0.5898	0.3029	95.14
MimicBrush [2]	20.80	0.7371	0.2178	67.19	18.20	0.6039	0.2849	88.59
ACE++ [25]	18.96	0.6922	0.1485	40.11	18.06	0.5695	0.1823	64.39
Ours	**26.40**	**0.8791**	**0.0820**	**28.31**	**21.95**	**0.7820**	**0.1350**	**47.09**

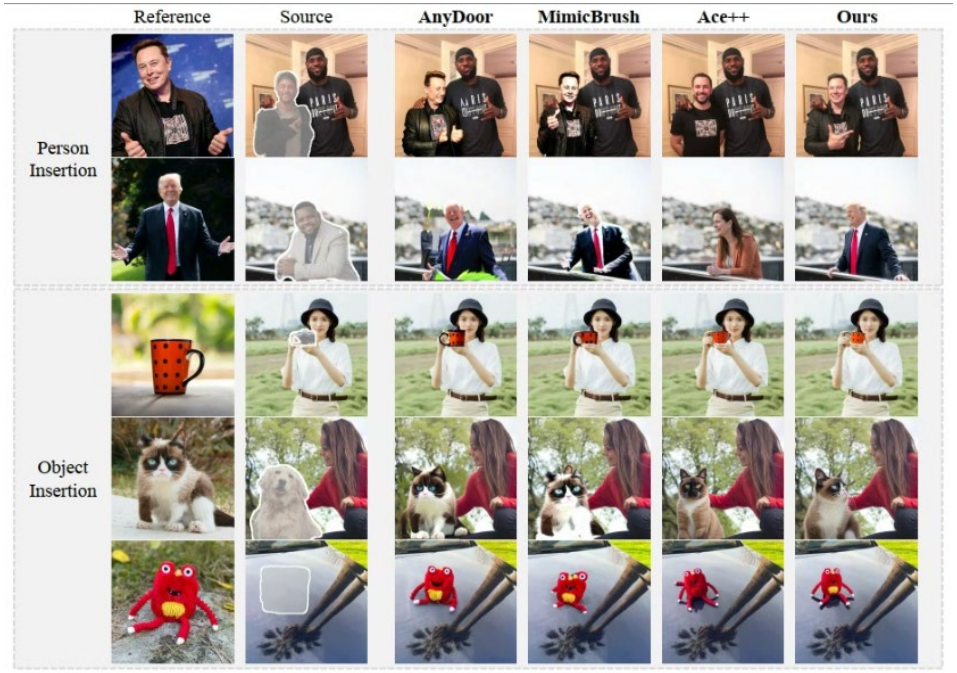
Methods	AnyInsertion (Garment)				VTON-HD			
	PSNR ↑	SSIM ↑	LPIPS ↓	FID ↓	PSNR ↑	SSIM ↑	LPIPS ↓	FID ↓
ACE++ [25]	18.11	0.7507	0.1086	35.62	17.48	0.7634	0.1107	28.96
Ootdiffusion [45]	18.07	0.8151	0.0970	87.38	21.63	0.8643	0.0605	28.36
CatVTON [6]	23.50	0.8477	0.0607	36.62	25.64	0.8903	0.0513	24.80
Ours	**23.78**	**0.8665**	**0.0522**	**28.54**	**26.10**	**0.9161**	**0.0484**	**19.51**

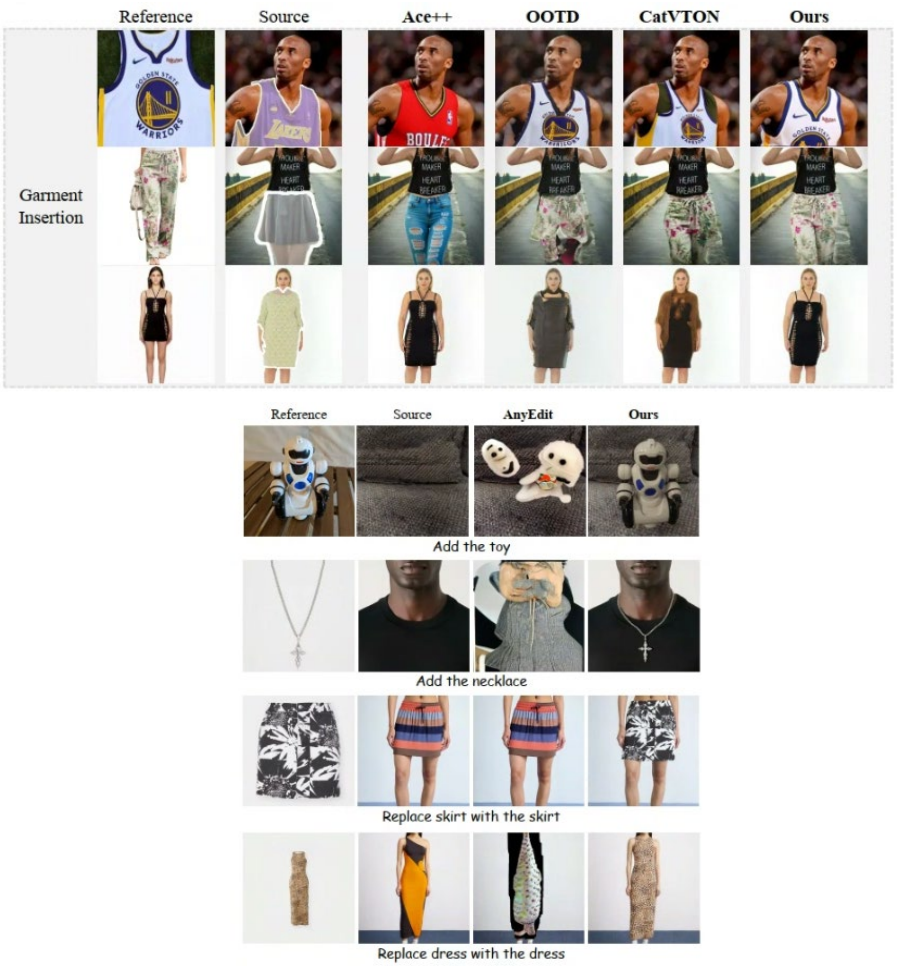
Methods	AnyInsertion (Person)			
	PSNR ↑	SSIM ↑	LPIPS ↓	FID ↓
AnyDoor [3]	14.71	0.6807	0.3613	217.17
MimicBrush [2]	20.58	0.7654	0.2125	108.26
ACE++ [25]	19.21	0.7513	0.1529	66.84
Ours	**23.85**	**0.8457**	**0.1269**	**52.77**

< Insert Anything 정량적 평가 비교 >

Insert Anything

- Experiments and Results
  - Qualitative comparison on object insertion
    - Datasets
      - AnyInsertion, DreamBooth, VTON-HD



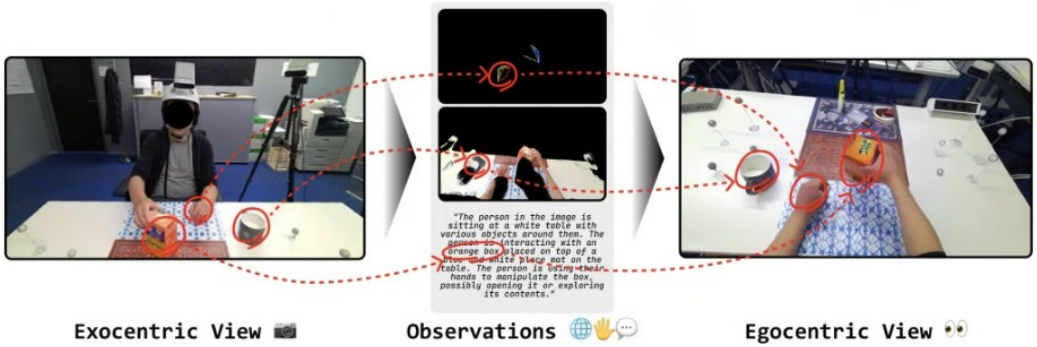


< Insert Anything 정성적 평가 비교 >

EgoWorld: Translating Exocentric View to Egocentric View using Rich Exocentric Observations

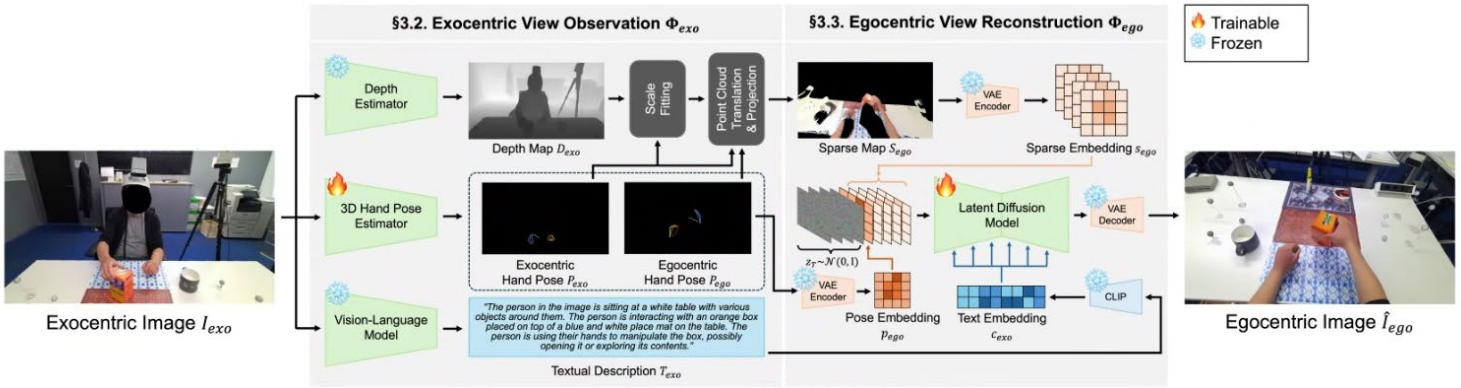
EgoWorld

- Egocentric Frame Generation
  - 1인칭 시점의 hand-object interaction image는 현재 3인칭 시점의 이미지에 밀려, 데이터 수집에 어려움이 있음
  - 기존 exo-to-ego 논문들은 현실적 사용이 어렵고, 일반화 성능도 낮음
    - 손의 occlusion을 처리 불가
    - 한 장면에 대해 여러 시점의 이미지 필요
    - 초기 egocentric frame 및 카메라 간 상대 위치 필요
  - 본 논문은 기존 3인칭 시점의 (exocentric) 이미지와 multimodal guidance를 활용해 1인칭 시점의 (egocentric) 이미지로 변환
    - Projected point cloud, 3D hand pose, textual description



EgoWorld

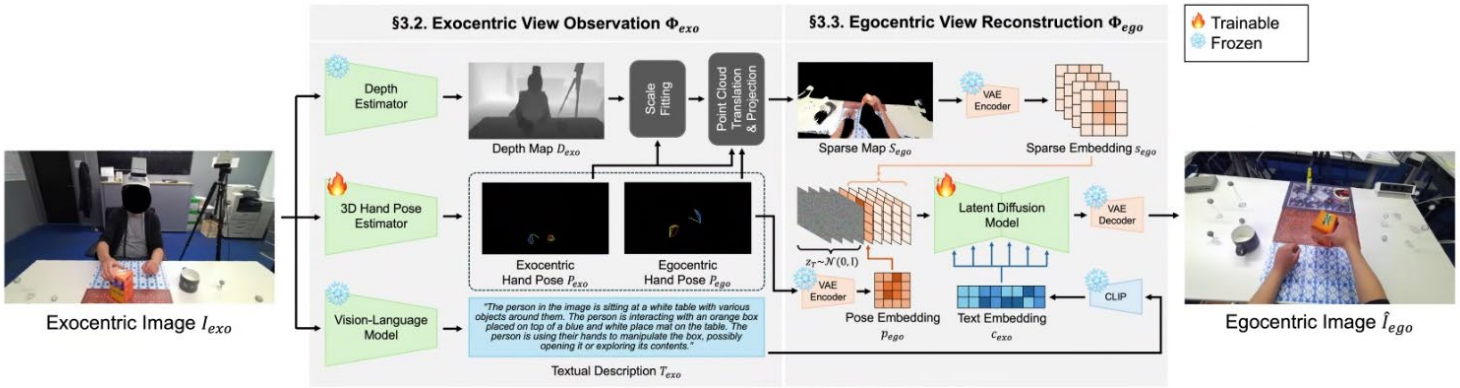
- Egocentric Frame Generation
  - First stage
    - 3인칭 시점의 이미지 (I_{exo})를 exocentric view observation (ϕ_{exo})을 통해 sparse egocentric RGB map (S_{ego}), 3D egocentric hand pose (P_{ego}), textual description (T_{exo})를 추출
    - $S_{ego}, P_{ego}, T_{exo} = \phi_{exo}(I_{exo})$
  - Second stage
    - 주어진 정보들을 바탕으로 egocentric view reconstruction (ϕ_{ego})를 통해 egocentric image (I_{ego})를 생성
    - $I_{ego} = \phi_{ego}(S_{ego}, P_{ego}, T_{exo})$



< EgoWorld의 pipeline >

EgoWorld

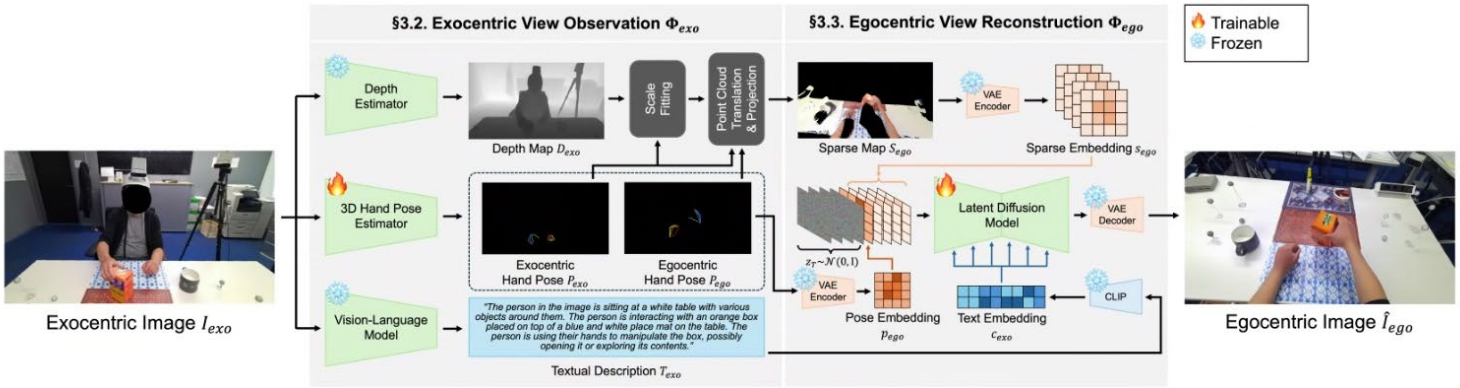
- Exocentric View Observation
  - 기존의 depth estimator (Vggt)를 활용하여 exocentric depth map (D_{exo})를 추출
    - 하지만, D_{exo}는 상대적 거리만 나타냄으로써 scale 정보가 없기에, 3D hand pose로 보정
  - 기존의 hand pose estimator를 활용하여 exocentric hand pose (P_{exo})를 추출
    - MANO mesh 기반의 P_{exo}를 활용해 hand depth map (D_{hand})를 추출
    - 이때 유효한 hand region은 Ω_{hand}로 정의
  - Scale 보정 계수 s* 계산
    - $s^* = median_{(u,v)\in\Omega_{hand}} \frac{D_{hand}(u,v)}{D_{exo}(u,v)+\delta\prime}$



< EgoWorld의 pipeline >

EgoWorld

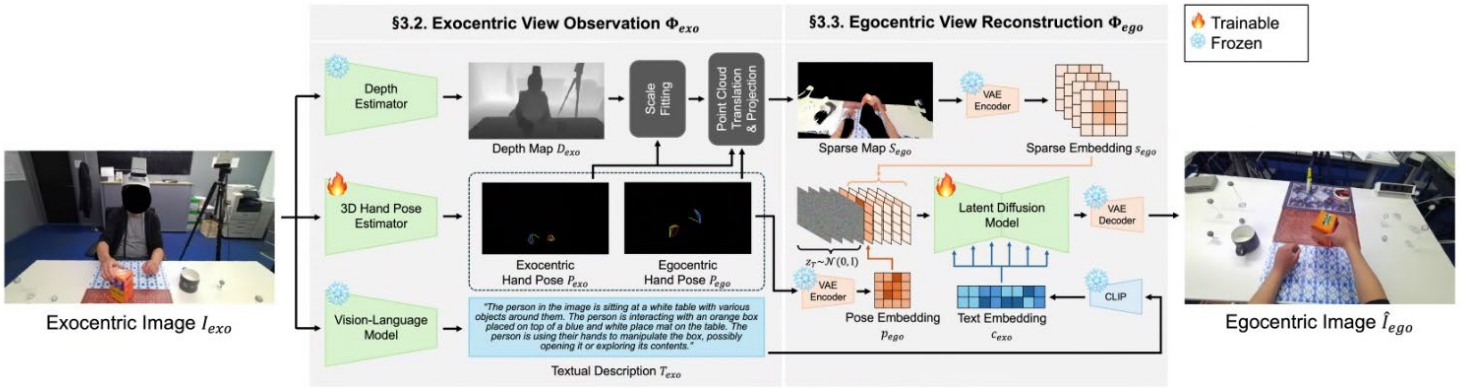
- Exocentric View Observation
  - Scale 보정 계수 s* 계산
    - $s^* = median_{(u,v)\in\Omega_{hand}} \frac{D_{hand}(u,v)}{D_{exo}(u,v)+\delta\prime}$
  - 최종 보정된 depth map (D'_{exo})
    - $D'_{exo} = s^* D_{exo}$
  - D'_{exo}, I_{exo}, intrinsic parameter (K_{exo})를 활용하여 point cloud (C_{exo}) 생성
  - 이후 P_{exo}와 P_{ego}의 대응점 사이의 관계를 파악해 exo-to-ego transformation matrix를 추정



< EgoWorld의 pipeline >

EgoWorld

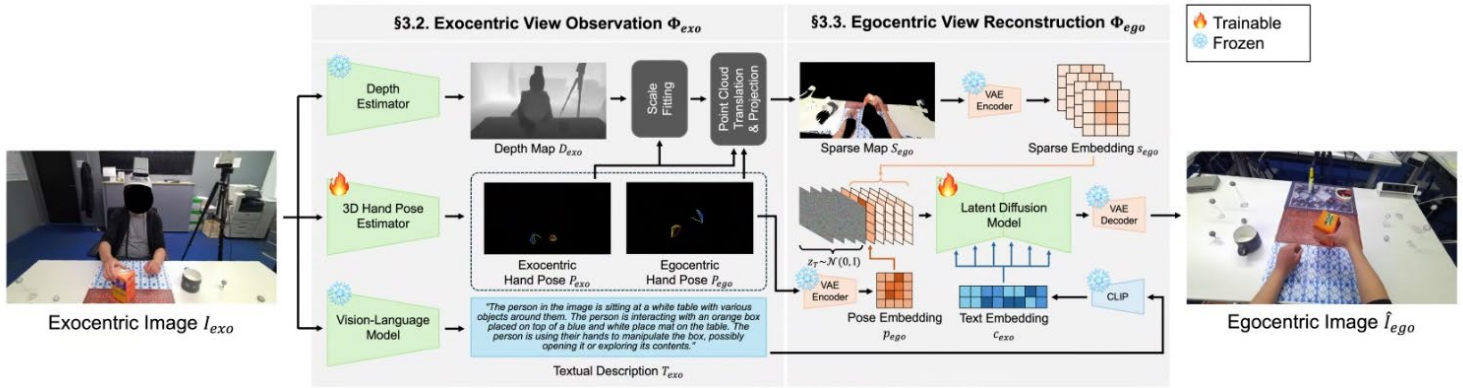
- Exocentric View Observation
  - 추정된 변환행렬을 통해서 C_{exo}를 point cloud (C_{ego})로 변환
    - 이후 intrinsic parameter (K_{ego})를 통해 egocentric 시점에서 투영한 sparse egocentric RGB map (S_{ego})를 산출
  - Text 설명 추출 (T_{exo})
    - VLM 모델 사용, I_{exo}와 질문 template을 사용하여 추출
      - "Describe in detail about the scene and the object that the person is interacting with using their hands."



< EgoWorld의 pipeline >

EgoWorld

- Egocentric View Reconstruction
  - 현재 주어진 정보 : $S_{ego}, P_{ego}, T_{exo}$
  - S_{ego}를 VAE encoder를 사용하여 latent embedding (4 channel)
  - P_{ego}는 intrinsic parameter로 2D project하여 2D egocentric hand pose map (P_{ego}^{2D}) 생성
    - P_{ego}^{2D}는 VAE encoder를 통해 4 channel에서 1 channel로 축소
  - 학습을 할 때는 GT egocentric image 또한 같이 latent (z_0)에 들어가고, noise가 추가됨
  - 결과적으로 LDM에 입력되는 최종 latent embedding z_t는 9 channel
    - $S_{ego}(4) + P_{ego}(1) + z_0(4)$



< EgoWorld의 pipeline >

EgoWorld

- Experiments and Results
  - Datasets
    - H2O datasets
    - TACO datasets
  - 모든 dataset은 4 가지로 나눔
    - Unseen objects
      - 6 종류의 object로 train, 새로운 2 종류의 object로 test
    - Unseen actions
      - 영상의 처음 80% frame들로 train, 마지막 20% frame으로 test
    - Unseen scenes
      - 4 종류의 scenes로 train, 새로운 2 종류의 scenes로 test
    - Unseen subjects
      - 한 사람의 이미지들로 train, 다른 사람의 이미지들로 test

EgoWorld

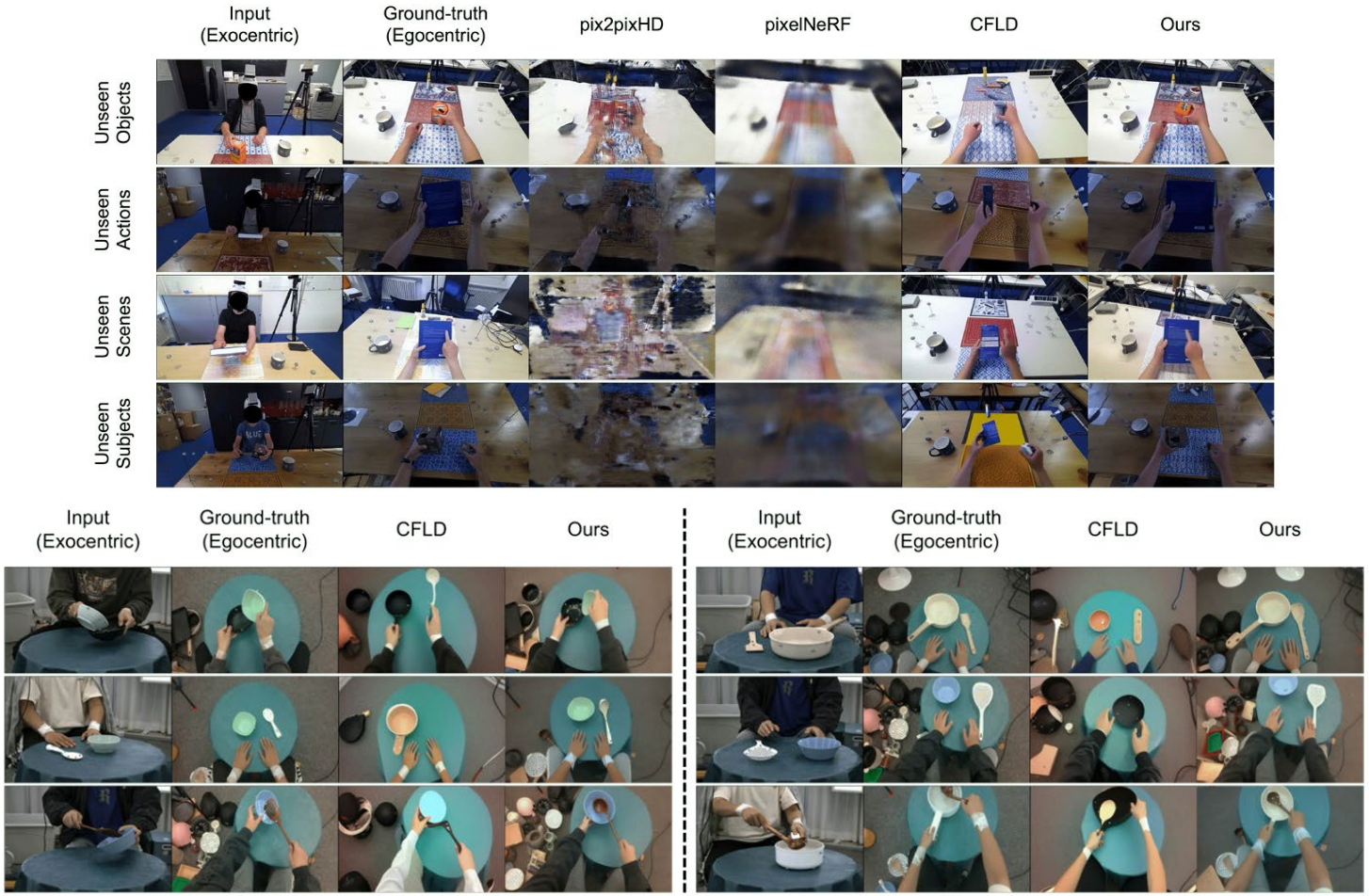
- Experiments and Results
  - Evaluation metrics
    - Peak Signal-to-Noise Ratio (PSNR)
    - Structural Similarity Index (SSIM)
    - Learned Perceptual Image Path Similarity (LPIPS)
    - Frechet Inception Distance (FID)

Scenarios	Unseen Objects				Unseen Actions			
Methods	FID↓	PSNR↑	SSIM↑	LPIPS↓	FID↓	PSNR↑	SSIM↑	LPIPS↓
pix2pixHD [44]	436.25	25.012	0.2993	0.6057	211.10	24.420	0.2854	0.6127
pixelNeRF [45]	498.23	26.557	0.3887	0.5372	251.76	27.061	0.3950	0.8159
CFLD [46]	59.615	25.922	0.4307	0.4539	50.953	28.529	0.4324	0.4593
EgoWorld (Ours)	**41.334**	**31.171**	**0.4814**	**0.3476**	**33.284**	**31.620**	**0.4566**	**0.3780**
Scenarios	Unseen Scenes				Unseen Subjects			
Methods	FID↓	PSNR↑	SSIM↑	LPIPS↓	FID↓	PSNR↑	SSIM↑	LPIPS↓
pix2pixHD [44]	490.32	18.567	0.2425	0.7290	452.13	18.172	0.3310	0.7234
pixelNeRF [45]	489.13	26.537	0.2574	0.7143	493.13	22.636	0.4135	0.6838
CFLD [46]	118.10	29.030	0.3696	0.6841	129.30	21.050	0.4001	0.6269
EgoWorld (Ours)	**90.893**	**31.004**	**0.4096**	**0.6519**	**96.429**	**24.851**	**0.4605**	**0.6188**

< EgoWorld와 다른 모델사이의 정량적 평가 비교 >

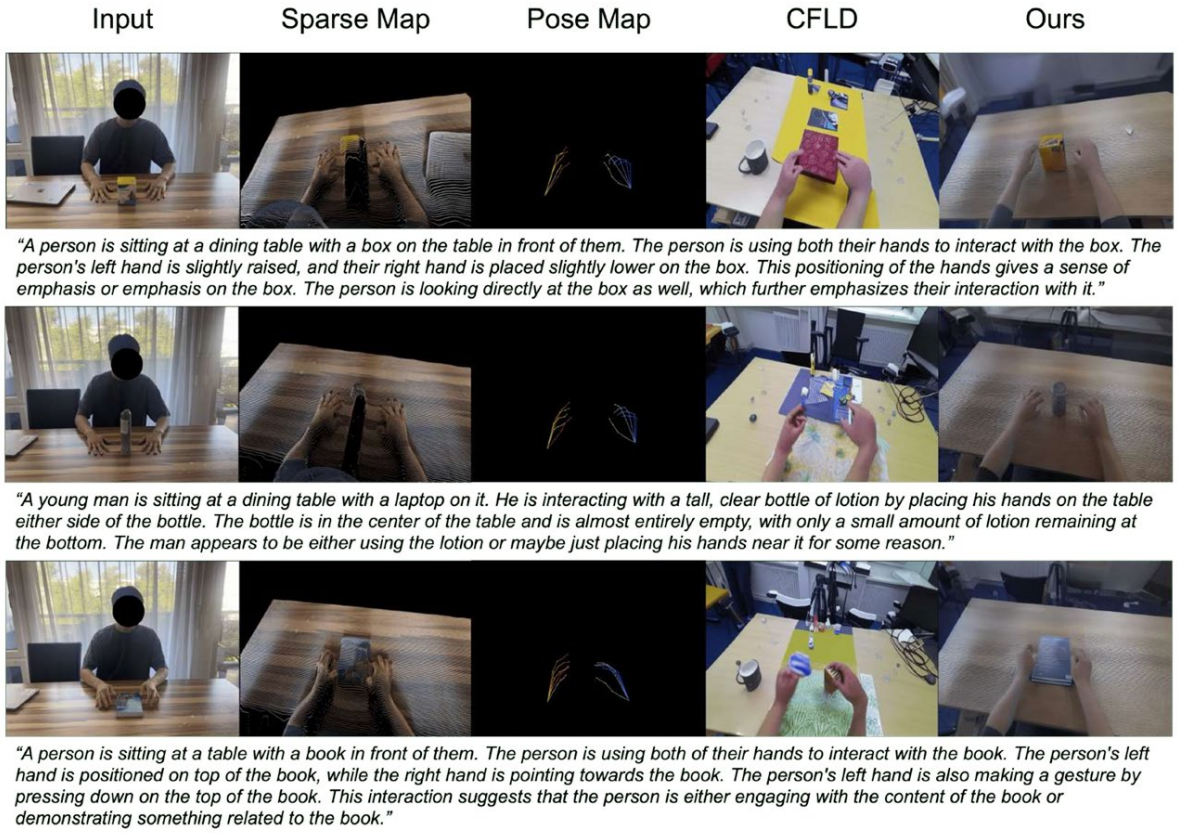
EgoWorld

- Experiments and Results
  - Qualitative comparison (위: H2O, 아래: TACO)



EgoWorld

- Experiments and Results
  - Qualitative comparison
    - Real world에서 기존 모델 (CFLD)과 비교



감사합니다