Metaverse Retargeting 종합설계프로젝트 보고서

1.논문 리뷰 요약

1.1 SegmentVR

목적 : video retargeting
이미지 처리 과정 : segmentation → inpainting → resizing → relocating

segmentation
각각의 frame 마다 사람(차지 면적이 1% 보다 큰 경우만)을 segment 진행함.
segment 한 결과를 binary masked 하고 이를 나중에 inpainting에 사용함.
segment 가 되지 않은 픽셀을 줄이기 위해 여백을 둠.

inpainting
비디오 중심 생성 모델과 이미지 중심 생성 모델이 있었는데 이미지 생성 모델은 전 프레임의 정보를 사용하지 않아서 어려움이 있었다.

resizing
해상도에 맞게 크기를 조정하였다.

relocating

한계 : relocating 과정이 일어나기 때문에 모든 객체가 정확히 탐지되지 않으면 문제가 발생함.

가장자리에 object가 위치할 때 object의 전체 part가 없으므로 relocating이 잘 되지 않음. (이미 객체 정보가 손실되었기 때문)

1.2 GPNN

목적 : PNN 기반 GAN -> GPNN 새로운 GAN과 유사한 생성 기능 + 가장 가까운 이웃 패치 기반 방법(PNN) ⇒ 두 개의 장점을 모두 얻음. GPNN은 단 몇초 안에, 최적화하고 학습할 필요없이 하나의 통합된 framework로 새로운 생성과제를 수행.
GAN과 유사한 새로운 생성 기능 GPNN은 SinGAN의 multi-scale architecture과 매우 비슷한 architecture를 가짐. 각 scale은 source xn과 유사한 patch distribution을 사용하여 다양한 output yn을 생성하는 단일 image generator G(PNN)로 구성됨.

PNN consists of 6 main algorithmic steps

Extract patches -> Compute distances -> Compute normalized scores -> Find NNs -> Replace by NNs -> Combine patches

1.3 InGAN

“Internal GAN”(InGAN) 제안; an image specific GAN

fully unsupervised, 단 한 장의 input 이미지 만으로 학습, 이미지 자체 조각의 분포를 유지하면서 다양한 크기나 모양의 이미지를 생성할 수 있다

한계 : InGAN은 비지도 학습이다→ 객체나 배경에 대한 의미적인 이해도가 없다. 때때로 웃기거나 부자연스러운 결과를 생성한다.

1.4 Seam Carving

효과적인 image resizing을 위해서는 geometric constraints뿐만 아니라 이미지 내용도 고려해야 한다. → 콘텐츠를 인식하면서 image reisizing하는 seam carving을 제안

*seam: a connected path of low energy pixels in an image

Seam carving 기술은 이미지 사이즈를 확대,축소할 수 있게 해 준다. 또한 content enhancement와 객체 제거를 하는 데 사용될 수 있다.

한계 : 이미지의 content amount less important pixel이 없으면 resizing 잘 못한다. 이미지 content의 형태 중요한 부분을 건너뛸 수도 있다.

2. 실험 진행

2.1 실험 진행 과정

~ 9월 25일까지 point cloud dataset 찾고, SegmentVR, InGAN 논문 리뷰 진행 - point cloud dataset 확보는 실패.(객체와 배경이 모두 잘 나타낸 데이터가 부족하여 실험에 알맞는 데이터를 확보할 수 없었음.)
9월 25일 ~ 10월 1일 GPNN, Seam Carving 논문 리뷰 진행, 전달받은 MIV 데이터셋 분석 및 파일 이름 정리.
10월 1일 ~ 10월 10일 local에 실험 환경 세팅. 대학원생분들과 회의 후 공대 9호관 실습실에서 실험 진행하기로 결정
10월 11일 공대 9호관 실습실 컴퓨터에 환경 설정
10월 12일 ~ 14일 실험 진행
10월 15일 논문의 '관련 연구' 파트 작성.

2.2 실험 결과

- 데이터셋 정보 : MPEG Immersive video 중 사람 객체가 포함된 데이터 총 6개

이름	해상도(원본비율)	리타겟팅 후 비율
C01 (Hijack)	4096x2048(2:1) => 1920x1080(16:9)	2560x1080
C02 (Cyberpunk)	2048x2048(1:1) => 1920x1080(16:9)	2560x1080
E02 (Carpark)	1920x1088(16:9)	1451x1088
E03 (Street)	1920x1088(16:9)	3414x1088
W01 (Group)	1920x1080(16:9)	3414x1080
W02 (Dancing)	1920x1080(16:9)	2560x1080

- version of cuda, python,pytorch: 11.7, 3.8.15,1.12.1

- model information

  segmentation - E2FGVI

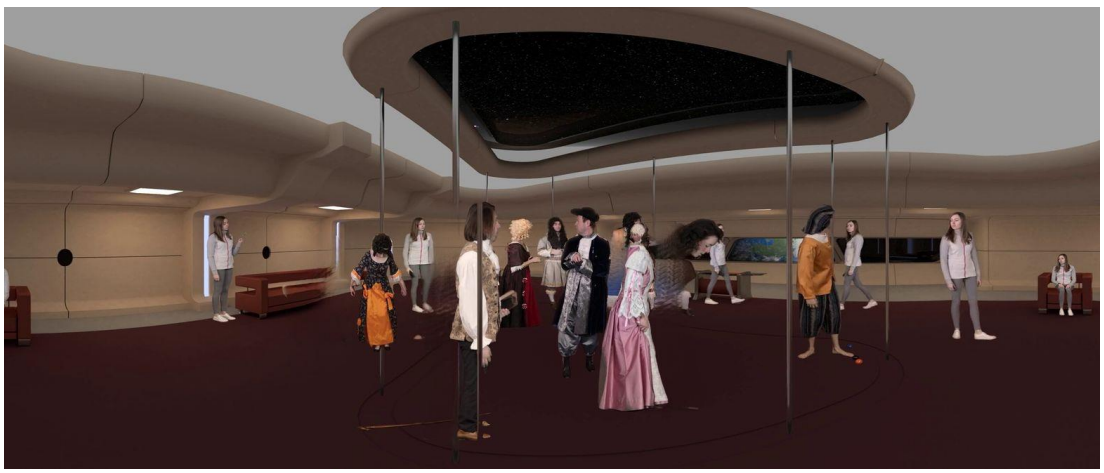
  inpainting - InGan
- code : https://github.com/coolho1129/Metaverse-Background-Research.git

- result video & image : https://drive.google.com/drive/folders/1D570_D3uIRF1B7EIM6_m8eo8aKS8Nfi5?usp=sharing

1. C01 (Hijack)
   객체(사람)가 많은 경우 Segmentation 과정에서 객체를 감지하지 못함.

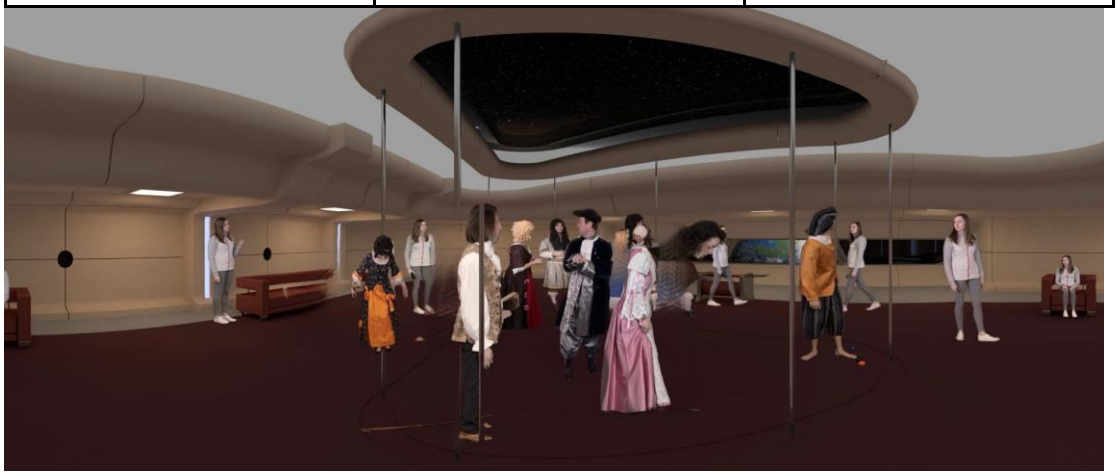
   Best Case - (000)

		
원본	masking	inpainting



   Worst Case - (225)

 원본	 masking	 inpainting



2. C02 (Cyberpunk)

Normal Case - (000)

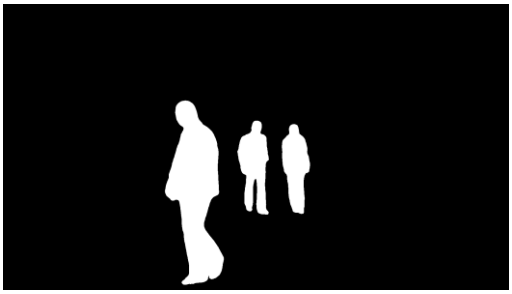
해상도가 너무 높아 해상도를 낮추는 과정에서 원본 이미지가 납작해짐. (추후 연구에서는 해상도를 낮출 때 원본 비율을 맞춰서 진행할 예정) 객체가 명확하고 다른 케이스에 비해 적어서 Segmentation이 잘 되었지만, inpainting이 완벽하지 않음.

 원본	 masking	 inpainting

3. E02 (Carpark)

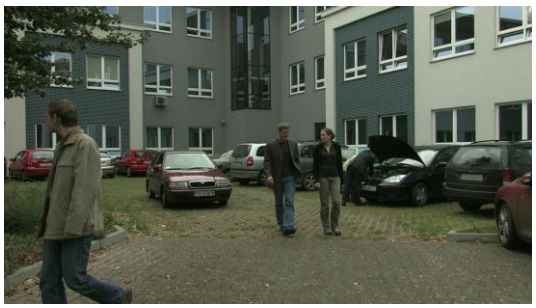
Best Case - (129)

객체 간 거리가 있을 때 Segmentation이 깔끔하게 되어 더 좋은 결과를 얻음.

 원본	 masking

Worst Case - (161)

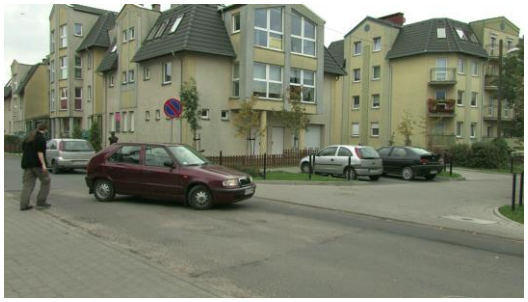
객체가 화면 밖으로 넘어가는 경우에 잘린 나머지가 반대편에 locating 됨.

 원본	 masking

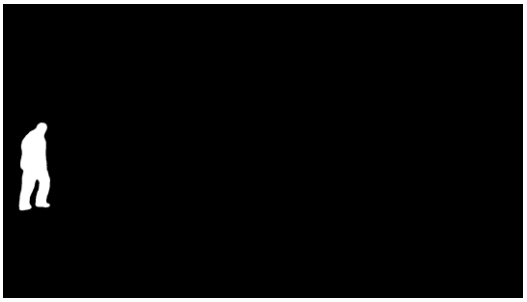
4. E03 (Street)

Best Case - (090)

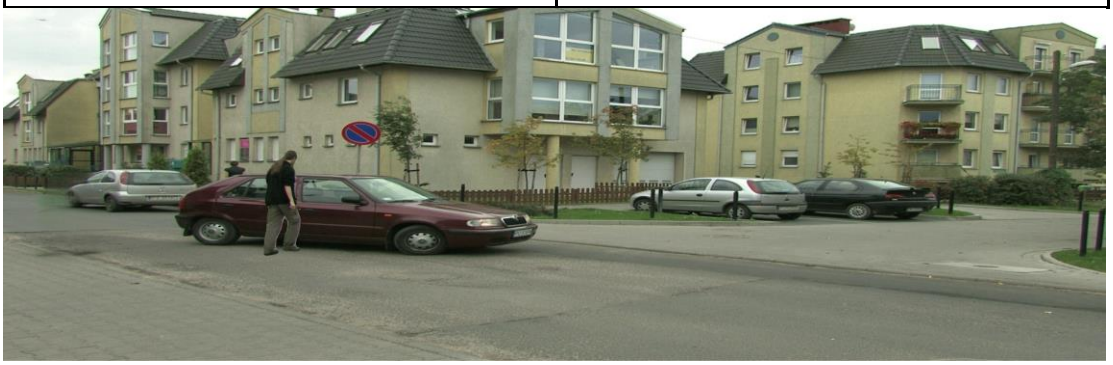
가로의 비율이 많이 늘어났음에도 사람의 비율이 늘어나지 않아 상대적으로 덜 부자연스러움.



원본

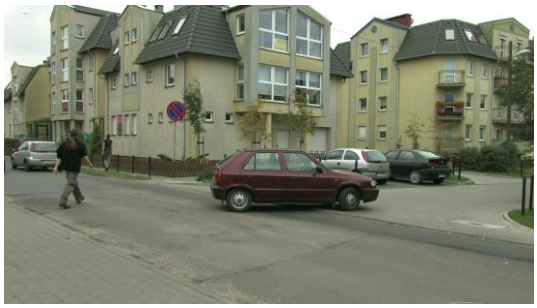


masking

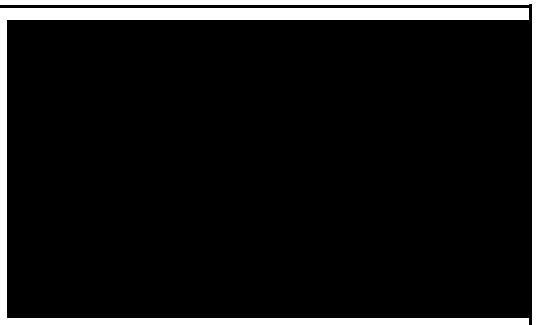


Worst Case - (110)

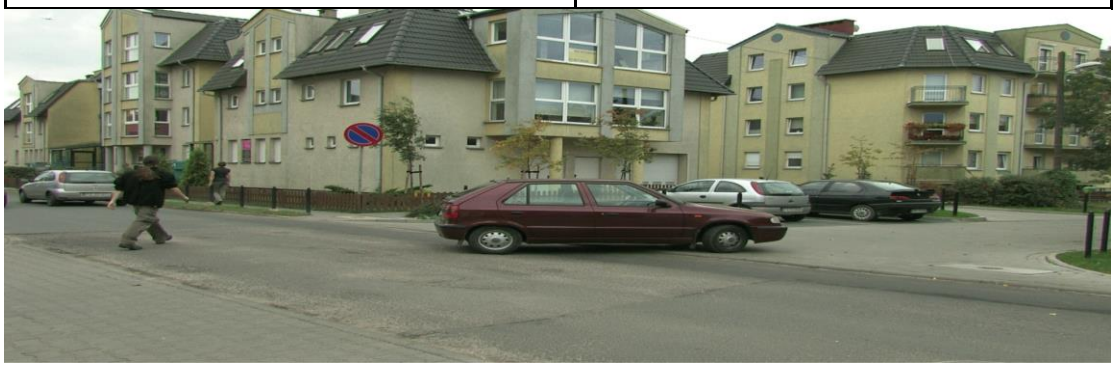
사람 인식을 하지 못해 Segmentation이 되지 않아 객체가 늘어남.



원본

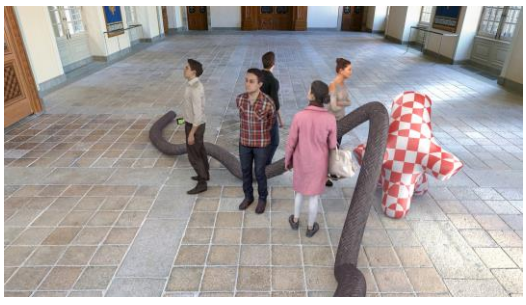


masking

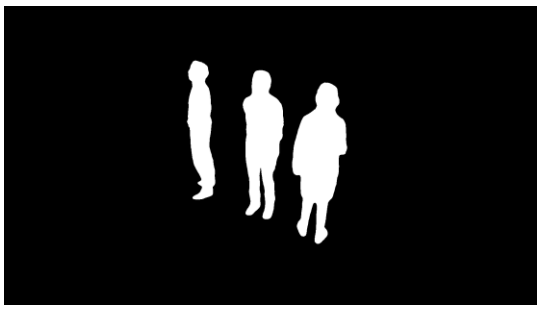


5. W01 (Group)

Normal Case - (034) : 모든 Frame이 좋지 않음.
객체가 많고, 일부는 구조물이나 다른 사람에게 가려져서 Segmentation이 잘 되지 않음. 특히 구조물 뒤쪽 2명의 사람은 모든 Frame에서 Segmentation 되지 않아서 매우 부자연스러운 결과를 얻음. 또한, 객체가 가지고 있는 object까지 추출을 하지 않아서 휴대전화가 공중에 떠 있음.



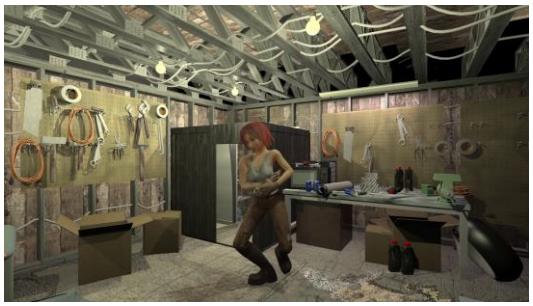
원본



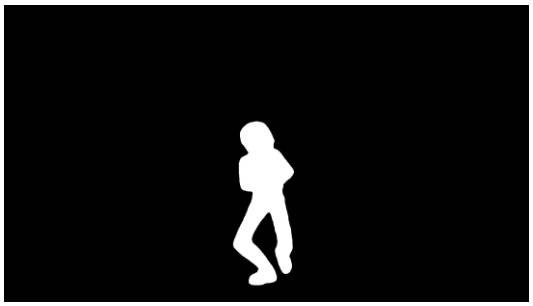
masking

6. W02 (Dancing)
 객체가 하나밖에 없고 정면을 바라보고 있는 동작을 취하고 있어 segmentation이 잘 진행되어 객체와 배경을 잘 분리해냄. 그 결과 inpating이 자연스럽게 수행되었고 만족스러운 결과를 도출해낼 수 있었음.

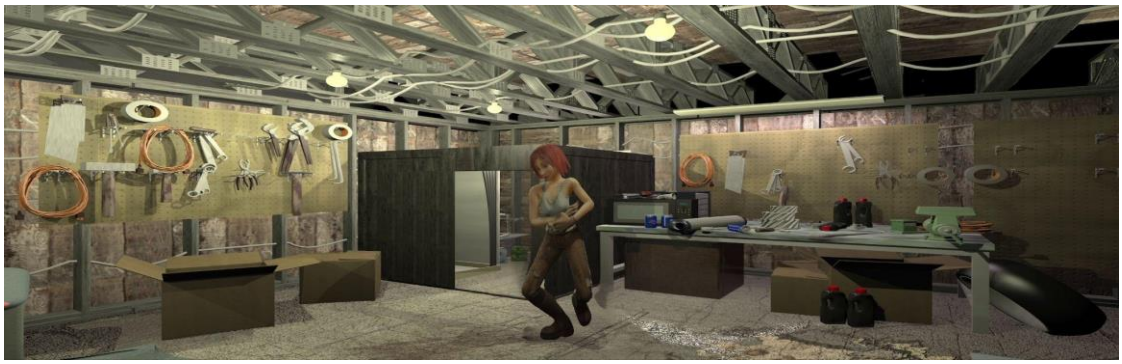
Best Case - (294)



원본



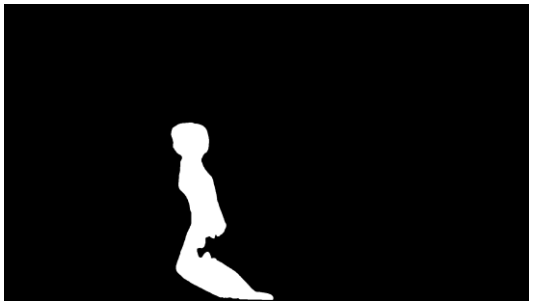
masking

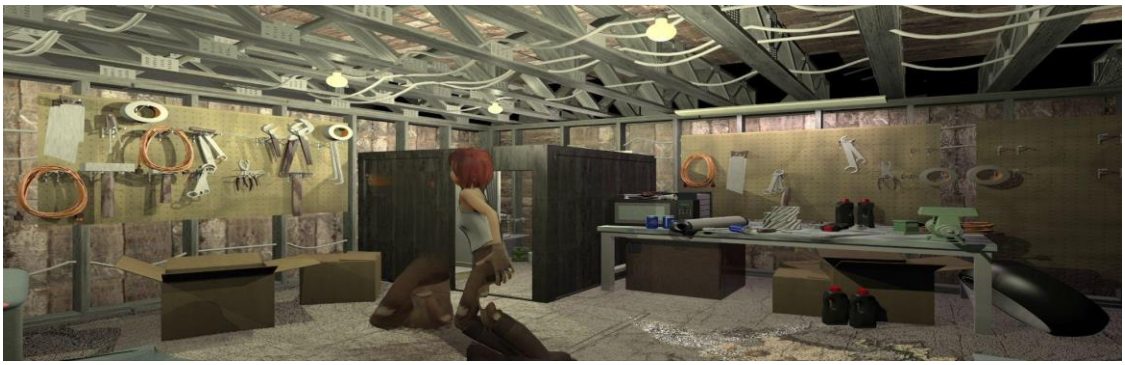


Worst Case - (239)

특정 동작을 취할때 segmentation이 잘 진행되지 않음. 이로 인해 배경과 객체를 잘 분리해내지 못하여 inpaiting이 부자연스럽게 진행됨.


원본


masking



2.3 실험 결과 분석

잘 된 결과 분석:
이미지 리타게팅은 특정 상황에서 우수한 성과를 나타내었다.
첫째로, 감지하려는 객체의 개수가 제한적인 상황에서 효과적으로 작동하였다. 이는 우리 모델이 한 번에 처리해야 하는 객체 수가 적을 때, 높은 정확도를 보였음을 알려준다.
둘째로, 객체들이 서로 겹치지 않는 상황에서 객체 인식 능력이 우수하였다. 이러한 조건에서, 우리 모델은 객체를 정확하게 식별하고, 객체 주변을 정확하게 마스킹하여 배경과 명확하게 분리하였다.
마스킹이 잘 된 사례에서는 객체가 없는 배경에서 인페인팅이 잘 되었고, 그 결과로 리타게팅된 결과도 준수했다.

잘 안된 결과 분석:
특정 상황에서 객체 감지 및 마스킹 시스템의 성능이 제한되는 상황이 관찰되었다.
먼저, 비디오에서 객체의 크기가 상대적으로 작거나 객체들이 많고 서로 겹칠 때, 객체 인식 성능이 저조하였다. 이러한 상황에서는 우리 시스템이 모든 객체를 정확하게 식별하기 어려워, 마스킹 작업이 완벽하게 이루어지지 않았다. 결과적으로, 이후의 인페인팅 단계에서 객체 주변에 흐릿한 효과가 발생하거나, 객체를 배경으로 인식하는 불완전한 결과가 도출되었다.
또한, 이슈 중 하나는 객체가 화면을 벗어나는 경우에 관한 것이다. 객체가 화면 밖으로 이동하는 상황에서, 객체가 화면의 반대쪽에서 잠시 등장하는 현상이 나타났다. 이는 마스킹

과정에서 객체 인식은 성공적으로 이루어졌음에도 불구하고, 객체의 위치 재조정 단계에서 부족한 점이 있었던 것으로 판단된다.

2.4 실험 중 이슈사항

1. local 환경 세팅 중 이슈

1. conda가 powershell에서 실행이 안됨

   ⇒ conda 재설치, conda 환경변수 세팅, powershell 대신 cmd 사용

2. colab 환경 세팅

   ⇒ 일단 소스코드 업로드 진행하였으나, 여전히 어려움 있음. 추가 방안 모색중

3. google drive 용량 부족

   ⇒ google drive 추가 결제를 진행하면 추가 저장공간 할당 가능. colab이 세팅이 된 뒤 용량 추가 구매하여 진행

4. 환경 세팅 중 mim 명령어 작동 안함

   ⇒ docker 환경으로 재 설정 시도 예정

5. masking.py 코드 실행 중 parameter src 설정의 어려움

   ⇒ 첨부된 shell 코드를 분석한 결과 mp4→ png로 바꾼 dir 임을 알아냄

6. mmcv-full 버전 오류가 있었으나 라이브러리 정리 후 해결됨. 하지만, GPU Ram 용량의 부족으로 코드 실행 불가

2. gpu가 없는 local에서 cpu로 masking.py를 실행했을 때, masking되지 않은 결과가 도출됨. 그 원인을 파악하기 위해 masking.py 코드를 분석해봄. detection을 진행하는 inference_detector 함수가 정의된 라이브러리에서 inference.py 파일을 살펴본 결과 주석에 CPU를 지원하지 않는다는 내용을 발견('CPU inference with RoIPool is not supported currently.') 그로인해 cpu에서는 detection을 지원하지 않는다는 것을 알게 되었음. ⇒ GPU가 공대9호관 실습실 이용으로 해결함

3. 1번 Hijack, 2번 Cyberpunk 영상을 inpainting하는 과정에서 gpu램 부족현상

 →4K영상 해상도를 FHD로 줄임으로써 해결

4. 1,2번 해상도 줄인 영상을 inpainting하는 과정에서 램 부족현상

→램자원을 소모하는 다른 응용프로그램을 종료하여 해결

3. 논문 작성(초안)

3.1 관련 연구

영상 미디어 산업의 급속한 발전과 디바이스의 보급으로 다양한 환경에서 영상 콘텐츠를 접할 수 있다. 텔레비전, 스마트폰, 빔프로젝터와 같은 다양한 디바이스는 각기 다른 화면 크기와 비율을 가지고 있으며, 동영상도 다양한 비율로 제작되고 있다. 특히 과거에는 4:3 표준화면 비율을 사용하였고 이는 현대의 16:9 비율 디바이스와의 호환성 문제가 발생한다. 이에 따라 영상 미디어의 화면 비율 조정이 점차 중요한 과제로 부각되고 있으며, 다양한 기술과 방법들이 발전해 오고 있다. 대표적으로 고전적인 이미지 처리기법인 seam carving[1]과 딥러닝을 활용한 InGAN[2]과 ORPVR[3] 이 있다.

seam carving은 이미지를 왜곡없이 리타게팅하기 위한 대표적인 알고리즘이다. 이 알고리즘은 콘텐츠를 인식하면서 이미지 리사이징을 가능하게 한다. seam은 한 이미지에서 그레이디언트 기반의 에너지 함수에 의해 정의되어, 위에서 아래 혹은 왼쪽에서 오른쪽으로 연결된 픽셀 경로이다. 낮은 에너지를 가진 seam을 제거하거나 삽입함으로써 이미지 전체의 에너지를 유지하거나 향상하게 하여 이미지의 종횡비를 변경할 수 있다. 이러한 seam carving 방식에는 두 가지의 한계가 있다. 이미지에서 덜 중요한 부분이 없으면 컨텐트를 인식한 이미지 리사이징이 잘 수행되지 않는다. 그리고 특정 구조를 가진 이미지에서 seam이 중요한 부분을 보존하지 못하는 경우도 있다.

InGAN은 한 장의 입력 이미지만으로 이미지의 내부 패치 분포를 학습하는 패치 기반 이미지 리타겟팅 모델이다. InGAN의 목표는 입력 이미지의 패치 분포를 생성 이미지와 매칭시키고, 이러한 이미지에 객체를 현지화하는 것이다. 이러한 목표를 달성하기 위해 멀티 스케일 분별망를 사용하여 패치 분포를 강화하고 인코더-인코더 아키텍처를 생성망에 도입하여 모델 붕괴를 막는다. InGAN는 여러 데이터 타입을 가진 이미지를 다양한 사이즈와 모양, 종횡비를 가진 이미지로 리타겟팅하여 일관성있고 완전한 결과를 도출한다. 그러나 InGAN은 입력 이미지 외의 추가적인 정보가 없기에 문맥적으로 의미가 맞지 않은 부자연스러운 결과를 도출한다는 한계가 존재한다.

기존의 Seam Carving 방식은 특정 형태의 객체를 가지거나 복잡한 이미지가 있으면 리타겟팅의 성능이 떨어진다는 문제점이 있다. 또한, InGan은 좋은 성능을 보였으나 객체에 대한 정보를 활용하지 않고 리타겟팅을 진행한다는 한계가 있다. [3]에서는 딥러닝기반의 분할과 인페인팅(Inpainting)을 활용하며 객체의 비율을 보존하며 리타겟팅을 진행할 수 있는 방식(이하 ORPVR)을 제안하였다. ORPVR은 다음과 같은 절차를 거쳐 비디오 리타겟팅을 수행한다. 먼저, 객체 검출을 통해 객체를 감지하고 객체 분할을 활용하여 주요 객체를 이미지에서 분리한다. 다음으로 객체가 분리된 부분의 공간을 인페인팅을 활용하여 채워 넣는다. 마지막으로 빈 곳이 채워진 이미지를 리사이징하고 분리해 놓은 객체를 이미지에 재배치한다. 이처럼 ORPVR은 객체의 정보를 활용하여 리타겟팅을 수행하여 InGan의 한계를 극복했을 뿐만 아니라 특정 형태의 객체가 있어도 좋은 성능으로 리타겟팅을 수행할 수 있어 Seam Carving 방식의 문제점도 해결하였다. 그러나 이 연구는

2D기반의 비디오 데이터셋에서만 수행되었으므로, 3D기반의 비디오 데이터셋에서 3D 정보를 활용하여 올바르게 리타겟팅을 수행할 수 있는지에 대한 추가 연구가 필요하다.

최근 메타버스 시장의 발전으로 3D 기반 데이터셋이 늘어나고 있다. 우리는 이러한 변화에 대응하기 위해 3D 기반 동영상인 MPEG Immersive video(MIV) 데이터셋으로 리타겟팅을 시도할 것이다. 하지만 앞서 소개된 모델들은 2D 기반 리타겟팅 기술이기에, 3D 데이터를 2D 데이터로 변환하는 사전 처리 과정이 필요하다. 본 논문에서는 사전 처리를 거친 MIV 데이터셋을 활용하여 3D 기반의 비디오 리타겟팅을 실험한 결과를 서술할 것이다.

참고문헌

[1] Avidan, S., & Shamir, A. (2023). Seam carving for content-aware image resizing. In *Seminal Graphics Papers: Pushing the Boundaries, Volume 2* (pp. 609-617).

[2]Shocher, A., Bagon, S., Isola, P., & Irani, M. (2019). Ingan: Capturing and retargeting the" dna" of a natural image. In *Proceedings of the IEEE/CVF International Conference on Computer Vision* (pp. 4492-4501).

[3]Jin, J. G., Bae, J., Baek, H. G., & Park, S. H. (2023). Object-Ratio-Preserving Video Retargeting Framework based on Segmentation and Inpainting. In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision* (pp. 497-503).