JDCS
디지털콘텐츠학회논문지
Journal of Digital Contents Society
Vol. 25, No. 11, pp. 3301-3308, Nov. 2024

Check for updates

FairMOT와 인페인팅을 이용한 자동 마스킹과 다중 객체 제거

이 효 진[1] · 변 혜 원[2*]
[1]성신여자대학교 대학원 미래융합기술공학과 석사
[2]성신여자대학교 AI융합학부 교수

Automatic Masking and Multi-Object Removal using FairMOT and Inpainting

Hyo-Jin Lee[1] · Hae-Won Byun[2*]

[1]Master, Department of Convergence Technology Engineering, Sungshin Women's University, Seoul 02844, Korea
[2]Professor, School of AI Convergence, Sungshin Women's University, Seoul 02844, Korea

[요 약]

영상 배경에 존재하는 콘텐츠와 무관한 객체는 시청자가 중심 객체에 집중하지 못하게 만들고 개인정보 유출 등의 문제를 일으킬 수 있어 제거될 필요성이 있다. 이 논문에서는 영상에서 다중 객체를 자동으로 제거하는 시스템을 제안한다. 기존 연구들은 주로 단일 객체에 대한 제거 기법에 초점을 맞추었으나, 본 연구에서는 다중 객체를 동시에 제거하는 방법론을 제안한다. 제안하는 시스템은 Yolov7과 FairMOT 모델을을 활용하여 영상 내 여러 객체를 자동으로 마스킹하고, 인페인팅 기술을 통해 제거된 다중 객체의 영역을 복원한다. 본 연구의 핵심은 여러 객체를 동시에 처리하는데 중점을 두었으며, 사용자가 별도의 세그멘테이션 작업을 할 필요없이 객체를 자동으로 식별하고 제거할 수 있다는 점, 그리고 인페인팅 손실 함수를 적절하게 설계하여 성능을 향상시킨 점이다. 이 연구는 신속한 처리 및 개인 정보 보호에 큰 기여를 할 수 있을 것으로 기대된다.

[Abstract]

Irrelevant objects in the background of a video can distract viewers from the main subject, expose sensitive information, or capture individuals without consent, raising privacy concerns and necessitating their removal. This paper presents a system for the automatic removal of multiple objects from an image. While previous studies have focused mainly on single-object removal, this study introduces a method for simultaneously removing multiple objects. The proposed system integrates the Yolov7 and FairMOT models to automatically detect and mask multiple objects in an image, using inpainting techniques to restore the regions where objects have been removed. The key contribution of this study is its ability to process multiple objects simultaneously, enabling automatic identification and removal without requiring user-defined segmentation. Additionally, performance is enhanced through the design of an inpainting loss function. This approach advances both processing speed and privacy protection.

색인어 : 객체 추적, 이미지 인페인팅, 다중 객체 탐지, 객체 제거, 생성형 AI
Keyword : Object Tracking, Image Inpainting, Multi Object Detection, Object Removal, Generative AI

http://dx.doi.org/10.9728/dcs.2024.25.11.3301

Received 14 October 2024; **Revised** 11 November 2024
Accepted 15 November 2024

***Corresponding Author; Hae-Won Byun**

Tel: +82-2-920-7615
E-mail: hyewon@sungshin.ac.kr

Ⅰ. 서 론

모바일 기기와 개인화된 기기의 확산으로 인해 개인의 데이터 생성 및 공유가 일상화되었으며, 특히 영상 데이터는 소셜 미디어를 통해 널리 공유되고 있다. 하지만 영상 배경에 존재하는 불필요한 객체는 개인정보 유출을 초래하거나 콘텐츠의 집중도를 떨어뜨릴 수 있어 제거의 필요성이 커지고 있다. 기존의 객체 제거 기법들은 주로 단일 객체에 초점을 맞추고 있으며, 여러 객체를 동시에 처리하기 위한 방법론은 부족하다.

기존 연구에서는 객체를 마스킹하여 제거한 후 인페인팅을 통해 복원하는 방식이 일반적이었다[1]-[3]. 그러나 이러한 방식은 대개 단일 객체에 초점을 맞추어 다중 객체를 동시에 처리하는 데 한계를 보였다[4],[5]. 예를 들어, Zhang 등은 인페인팅 기술을 활용하여 배경과의 일관성을 중시하였으나, 다중 객체를 동시에 처리하는 접근법은 제시하지 않았다[6]. 최근에 Park 등은 다중 객체 처리에 대한 연구를 시도하였으나 여전히 사용자 개입이 필요하여 실시간 처리의 가능성은 제한적이었다[7].

이러한 기존 연구들의 한계를 극복하기 위해 본 논문에서는 다중 객체 제거(Multi-Object Removal) 기법을 제안한다. 본 시스템은 객체 추적과 인페인팅 기술을 결합하여 이미지 내 다수의 객체를 자동으로 식별하고 제거할 수 있도록 설계되었다. 특히, 기존 방식과 달리 여러 객체를 동시에 처리하는 데 중점을 두었으며, 사용자가 별도의 세그멘테이션 마스크를 생성할 필요 없이 객체를 자동으로 마스킹할 수 있는 점에서 차별화된다. 본 연구는 기존의 객체 제거 기술이 가진 제약을 해소하고, 영상 공유 시 개인의 프라이버시 보호를 한층 더 강화하는 데 기여할 것으로 기대된다.

본 논문의 구성은 다음과 같다. 2장에서 객체 추적 및 객체 제거 기술과 객체 제거를 목표로 한 이미지 인페인팅, 모델을 경량화 연구를 서술하고 3장에서 제안하는 시스템을 설명한다. 4장에서는 제안한 시스템에 대한 실험을 평가 및 분석하며 5장에서는 제안한 시스템의 한계를 설명하며 마무리한다.

Ⅱ. 관련연구

객체 제거 기법에서 마스킹을 자동으로 진행하기 위해선 영상 내 객체를 탐지하는 과정이 필수적이다. 따라서 객체 단위 자동 마스킹에는 객체 탐지(Object Detection) 및 객체 추적(Object Tracking) 기술과 이미지 인페인팅 기술이 모두 이용된다.

2-1 객체 추적

객체 추적 모델은 객체 탐지 모델의 결과를 다른 모델로 전달하여 객체를 추적하는 방식과 객체 탐지 모델을 추적 모델로 이용하는 방식으로 나뉜다.

SORT(Simple Online and Real-time Tracking)[2] 및 DeepSORT[8]는 기존의 객체 탐지 모델을 활용하여 영상 내 객체를 분석하고 이를 기반으로 객체 추적을 진행한다. SORT는 칼만 필터(Kalman Filter)와 헝가리안 알고리즘(Hungarian Algorithm)을 활용해 이전 프레임과 현재 프레임의 객체 동일성을 예측 및 파악한다. 이를 통해 객체 식별 번호를 할당하고 객체를 추적하는 것이다. DeepSORT는 SORT의 객체 추적 성능을 향상하고자 딥러닝을 도입하여 구조적인 변화를 주었다. 이러한 구조 개선은 실시간 탐지를 가능하게 했으나, GPU와 CPU의 연산 부담으로 인해 구축 가능한 환경이 제한된다는 단점이 존재한다.

JDE(Joint Detection and Embedding)[7] 모델은 객체 탐지 모델의 중간 특성 맵(Feature Map)을 이용하여 프레임 간 객체 동일성을 향상시키고자 하였다. 그러나 객체 탐지 및 추적 단계를 종단 간 연결(End-to-end) 모델로 구성하여 객체 클래스 분류와 같이 객체 탐지 시에만 유용한 정보를 객체 추적 단계에 전달하였다. 또한, 파라미터를 갱신할 때 프레임 간 객체 매칭 실수로 인한 손실을 객체 탐지 단계에 전달하는 것이다. 이 때문에 객체 매칭 알고리즘을 개선한 다른 객체 추적 모델과 비교해 매칭 정확도가 낮다. 이에 FairMOT[6]는 JDE의 객체 검출, 임베딩 추출을 통합하되 객체 탐지 단계와 추적 단계를 분리한 모델을 제안하였다. 해당 모델은 기존의 객체 추적 모델과 비교해 FPS 대비 다중 객체 추적 성능이 높다.

2-2 객체 제거

Weder 등[4]은 다양한 각도의 이미지를 활용하여 객체를 제거하는 모델을 제안하였다. 해당 모델은 같은 시점의 다양한 배경 정보를 이용하여 결과 이미지를 자연스럽게 생성하지만, 다양한 각도의 이미지가 필요하고 중심이 되는 객체를 제거하여 이미지를 복원한다는 한계를 가진다. Alkobi 등[8]은 GAN을 다량으로 이용하여 이미지를 자연스럽게 복원하고자 하였으나 마스킹 영역 바깥에서부터 단계적으로 복원하여 페색된 객체나 복잡한 배경에 대해 자연스럽게 복원하지 못한다. Shetty 등[5]은 영상에서 객체를 확실히 제거하고자 복원한 이미지의 객체 탐지 결과를 손실 함수에 반영하였다. 이는 특정 클래스의 객체를 제거하는 데 효과적이지만, 영상에서 지우고자 하는 객체와 남기고자 하는 객체의 클래스가 같은 경우를 고려하지 못했다.

Darapaneni 등[9]은 YOLOv3를 이용해 객체 검출을 수행하고 Contextual Attention (CA) 구조를 탑재한 SRGAN을 이용해 이미지 인페인팅을 시도하였다. 해당 모델은 작은 객체에 대해서 높은 복원력을 보여주었지만, 이미지 복원 시 전체적인 색감에 변화를 준다는 단점이 있다.

Deepfillv2[10]는 Gated Convolution 연산을 활용한 인페인팅 모델로 다양한 마스크 입력에 대해 사실적으로 복원

하고자 하였으나, 복잡한 배경을 제대로 복원하지 못했다. 이와 유사하게 LaMa[11]는 Fourier Convolution 연산을 활용한 모델로 크기가 큰 마스크에 대한 복원 능력을 향상시키고자 하였다. LaMa는 배경 객체를 유지하며 고해상도의 이미지를 생성하지만, 이미지를 사실적으로 복원하기 위해 제거해야 하는 객체를 남기기도 한다.

IA(Inpainting Anything)[12]는 객체를 배경과 분리하여 효과적으로 제거하고자 인페인팅 모델에 세그멘테이션 모델을 결합한 모델이다. 세그멘테이션 모델로 인해 객체 단위 마스크를 생성할 수 있으며 이미지를 자연스럽게 복원한다. 그러나 해당 모델은 단일 객체를 대상으로 한다는 점과 긴 추론 시간이 한계이다.

Ⅲ. 영상 내 객체 제거 시스템

본 논문은 영상 내 객체의 남용 가능성을 고려하여 불필요한 객체를 제거하는 것을 목표로 한다. 이를 위해 영상에서 객체를 추적하고 중심 객체를 선정하는 과정이 필수적이다.

그림 1에서 나타낸 바와 같이, 먼저 입력된 영상에서 객체를 탐지하고, 식별된 객체를 마스킹하여 제거한다. 이후에는 인페인팅 기법을 사용하여 마스킹된 영역을 복원한다. 따라서 전체 시스템은 다음 세 가지 단계로 구성된다. 첫째, 입력 영상 내 객체를 분석하는 단계, 둘째, 중심 객체를 선정하고 나머지 객체를 영역별로 처리하는 단계, 셋째, 마스킹된 영역을 복원하는 단계이다.

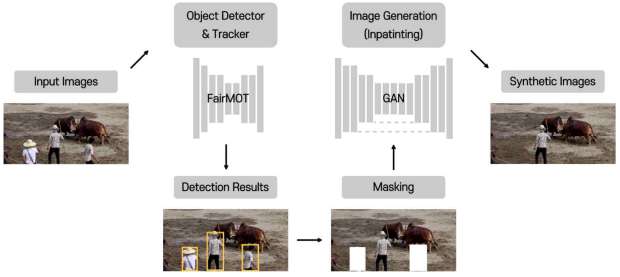

그림 1. 제안하는 객체 제거 시스템
Fig. 1. Proposed object removal system

3-1 영상 객체 분석

입력 영상 내 객체를 분석하는 단계에서는 제거할 객체를 탐지하기 위해 객체 추적 모델을 이용한다. 그림 2는 객체 추적을 위해 이용한 FairMOT[13]의 객체 추적 과정이다. 먼저 백본 네트워크를 통해 입력 이미지에서 특성(Feature)을 추출한다. 이때 백본 네트워크의 성능은 객체 추적 성능에 직접적인 영향을 끼치므로 사전 학습된 모델을 이용하는 것이 일반적이다.

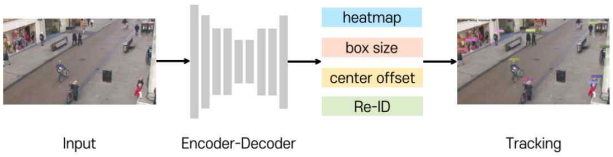

그림 2. 객체 추적 모델
Fig. 2. Object tracking model

이에 객체 검출 성능과 실시간성을 고려하여 YOLOv7[14]을 이용하였다. YOLOv7은 YOLOv5와 비교해 모델의 추론 시간은 유지하되 탐지 성능을 높인 모델이다. 이는 데이터 세트의 환경 다양성을 위한 증강 기법과 활성화 함수 개선 등을 통해 모델의 객체 검출 성능을 향상하면서 일정 이하의 추론 시간을 유지한 YOLOv4와 구조적 유사성을 갖는다. 그러나 YOLOv7은 기존의 YOLO와 확연한 차이점이 존재한다. 기존의 YOLO는 그림 3의 왼쪽처럼 추론 및 학습 시 그래디언트(Gradient) 갱신 흐름을 유지하고자 모델의 깊이를 늘리는 방식으로 Convolution 연산을 추가해왔다. 이와 달리 YOLOv7은 Convolution 연산 흐름을 Inception Module과 유사하게 구성하여 학습 효율을 높였다. 이 때문에 YOLOv7은 다른 YOLO 모델보다 FPS 대비 객체 검출 성능이 더 높게 관측된다.

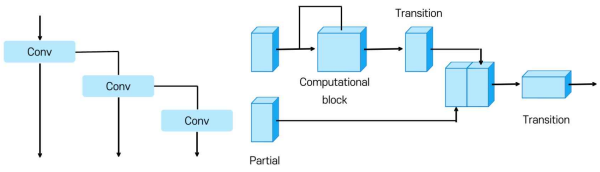

그림 3. (왼) 연결 기반 구조 (오른) 깊이와 너비를 고려한 복합 스케일링 구조[14]
Fig. 3. (left) concatenation-based structure (right) compound scaling up depth and width structure[14]

백본 네트워크를 통한 특징 추출이 완료되면 추출된 특징 인코더-디코더를 통해 객체 간 유사성 계산을 위한 임베딩 벡터로 변환된다. 이후 이를 활용하여 객체의 위치와 클래스를 예측한다. 나아가 수식 (1)의 코사인 유사도(Cosine similarity)를 통해 객체의 움직임을 연속적으로 추적하고, 각 객체의 고유 ID를 할당하여 객체를 추적하게 된다.

$$Cosine\ Similarity = \frac{X \cdot Y}{\|X\|_2 \cdot \|Y\|_2} \quad (1)$$

3-2 중심 객체 분류

영상 내 중심 객체를 선정하고 나머지 객체를 영역별로 처리하는 단계는 영상에 존재하는 객체 중, 불필요한 객체를 지우는 단계이다. 해당 단계에는 영상 내 중심 객체를 선정하는 과정과 제거되어야 할 객체를 판별하는 과정이 수반되어야 한다. 중심 객체는 되도록 영상의 초기 상태를 기준으로 중앙

에 존재하는 객체로 선정하였다. 또한, 선정된 중심 객체의 ID를 기록하여 다음 프레임에서 중심 객체를 마스킹 대상에서 제외하였다. 이후 객체를 영역별로 처리하는 것은 중심 객체로 선정되지 않은 나머지 객체를 대상으로 하며 탐지된 객체 정보를 이용하여 Bounding Box를 전체 마스킹하였다.

3-3 인페인팅

마스킹 영역을 복원하는 단계는 객체를 단순 제거한 이미지를 실제에 가까운 자연스러운 형태로 복원하는 단계이다. 이전 단계의 출력이 객체가 존재한 위치의 모든 정보를 제거한 이미지이기 때문에 영상의 시각적인 품질과 실제성을 높이고자 마스킹 처리된 영역을 복원하는 과정이 필요하다. 이 과정은 그림 4에서 볼 수 있듯이, 생성자(Generator)를 통해 객체가 제거된 가상의 이미지를 생성한다. 또한, 학습 과정에서 판별자(Discriminator)를 통해 생성된 이미지의 품질과 진위를 확인하는 과정을 거친다. 이러한 구조는 기존의 이미지 인페인팅 모델과 유사하게 CA[3] 구조를 기반으로 하였다. 해당 모델은 Coarse Network를 통해 전반적인 복원을 완료하는 1차 복원, Refinement Network를 통해 이미지 품질을 높이는 2차 복원으로 진행된다. 이렇게 생성된 이미지는 전체 이미지와 복원된 영역만을 고려하는 두 개의 판별자에 의해 평가되며 이를 통해 전체 모델에 피드백된다. 이러한 구조는 복원 영역의 해상도를 높여 보다 자연스러운 이미지를 생성한다.

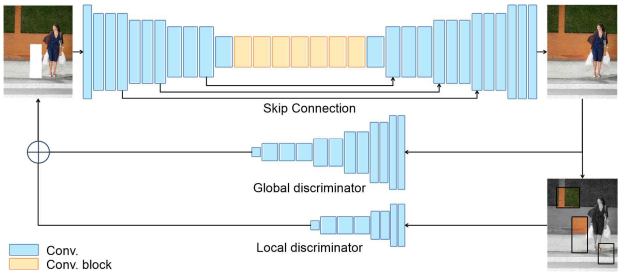

그림 4. 이미지 인페인팅 모델
Fig. 4. Image inpainting model

생성자의 이미지 생성 과정을 그림 4에서 확인할 수 있다. 스트라이드(Stride)를 조절한 Convolution 연산을 통해 특성맵의 크기를 줄인다. 중심에 존재하는 Convolution 블록에서는 마스킹 바깥의 패턴을 참고해 마스킹 영역의 값을 변형한다. 이후, Skip-Connection과 Up-sampling을 통해 이미지의 전반적인 특징을 유지하며 보다 자연스러운 이미지를 생성한다.

판별자는 이미지의 전반적인 복원 품질을 확인하고 마스킹 영역의 복원 품질을 확인하고자 2개의 판별자로 구성하였다. 또한, 2종의 판별자는 학습 과정에서 그래디언트 피드백 흐름을 통일시키고자 각각 4x4 크기의 패치를 결과로 도출하며 이 두 결과물을 결합하여 입력으로 주어진 이미지의 진위를 판별하고 각 판별자의 진위 판별을 진행한다.

3-4 손실 함수

손실 함수는 전체 시스템에서 학습을 주도하는 인페인팅 모델을 기준으로 구성하였다. 영상 내 객체를 분석하는 단계, 중심 객체를 선정하고 나머지 객체를 처리하는 단계는 각각 사전 학습된 모델과 학습을 진행하지 않는 알고리즘을 이용하여 손실 함수에 반영하지 않았다.

$$L_G = E_{\tilde{x} \sim P_G}[1 - D(G(\tilde{x}))] \tag{2}$$

수식 (2)는 생성자의 손실 함수로서 생성자가 생성한 이미지를 판별자가 가짜로 판별했을 때 큰 값을 반환한다. 이때 판별자를 통한 결과를 하나로 구성하여 모델의 학습 안정성을 높이고자 하였다. 학습 안정성은 모델 학습 시에 파라미터 수렴 가능성을 의미한다. 학습 안정성을 높이면 모델 학습은 파라미터나 손실 함수의 급감 및 급증과 같은 변동 없이 단조롭게 수렴하게 된다. 반면, 복잡한 손실 함수는 학습 중에 기울기 소실(vanishing Gradient)이나 모드 붕괴(Mode Collapse)와 같은 문제가 발생하기 쉽다. 또한, 학습 데이터 세트의 정보를 지나치게 피드백하여 과적합(Over-fitting)이 일어날 수 있으며, 이는 모델의 전반적인 성능 저하로 이어질 수 있다. 이러한 문제를 방지하고자 생성자의 손실 함수를 간단하게 구성하였다.

제안한 인페인팅 모델에서 판별자는 전체적인 복원 품질을 확인하는 전체 판별자(global discriminator)와 복원된 영역의 품질을 집중적으로 확인하는 지역 판별자(local discriminator)로 나뉜다. 그러나 수식 (3)을 보면 알 수 있듯이 판별자의 전체 손실 함수에 전체 판별자와 지역 판별자를 통한 손실 함수뿐 아니라 전체 판별자의 손실 함수가 포함되어 있다. 일반적인 CA 구조는 판별자의 손실 함수 또한 전체와 지역으로 나누어 각각 피드백하는 것에 그친다. 그러나 제안한 시스템의 학습 과정에서 판별자의 그래디언트 피드백 흐름을 통일시키고자 하나의 진위 값을 도출하게 구성하였으므로 전체 판별자가 등장하게 되었다. 따라서 두 판별자의 결과물을 결합하여 진위를 확인하는 전체 판별자의 결과를 손실 함수에 추가하였다. 전체 판별자 뿐만 아니라 전체, 지역 판별자 또한 각각 피드백하는 것은 각 판별자의 파라미터를 섬세하게 조절하기 위해서이다. 또한, 이를 통해 생성자와 판별자간의 학습 균형을 맞추고자 하였다.

$$\begin{aligned} L_D = & -E_x[\log(D_{Global}(x))] - E_x[\log(D_{Local}(x))] \\ & - E_x[\log(D_{Total}(x))] - E_{\tilde{x} \sim P_G}[1 - D(G(\tilde{x}))] \end{aligned} \tag{3}$$

수식 (4)는 생성자와 판별자의 손실 함수를 하나의 손실 함수로 나타낸 것이다. 이 손실 함수를 통해 생성자는 두 번째 항의 계산 값을 줄이고자 하며, 판별자는 두 항의 값을 모두 피드백 받아 전체 수식의 값을 높이고자 한다. 이렇게 생성자

와 판별자의 상반된 손실 함수 구성을 통해 인페인팅 모델이 학습된다.

$$\min_{G}\max_{D}\left[E_{x\sim P_R}[D(x)]-E_{\tilde{x}\sim P_G}[D(\tilde{x})]\right] \quad (4)$$

Ⅳ. 실험 및 결과

4-1 데이터 세트와 실험 환경

본 논문에서 중심 객체와 지워야 할 객체의 클래스는 편의상 사람 클래스로 제한하였다. 이에 BGVP (Background Vulnerable Pedestrian Dataset)[15] 데이터와 WiderPerson[16] 데이터와 같이 다량의 사람이 존재하는 이미지 데이터를 학습 데이터로 선정하였다. 그림 5는 BGVP 데이터셋을 보여준다.

실험에 사용된 컴퓨터 사양은 표 1과 같다. 또한, 실험에 이용한 데이터는 학습 데이터(Train data), 검증 데이터(Validation data), 지표 계산을 위한 테스트 데이터(Test data)로 나누어지며 8:1:1의 비율로 나누어 사용하였다.

그림 5. BGVP 데이터 세트[15]
Fig. 5. BGVP dataset[15]

표 1. 실험 환경
Table 1. System configuration

Computing Environment	Workstation
Processor	Intel i7-7700, 4.2GHz CPU
Memory	64GB
Operating System	Ubuntu 16.04
Graphics Card	NVIDIA GTX 1080 Ti GPU 4대

4-2 평가 지표

모델 학습에 이용한 파라미터는 배치(Batch Size)가 16, 평균 에포크(Epoch)가 500이다. 또한, 학습은 SGD와 Adam 중, 더 작은 손실 값을 보인 Adam을 통해 최적화되었다. 나아가 다음의 3가지 평가 지표를 통해 인페인팅 모델의 학습을 평가하였다.

$$PSNR=10\times\log_{10}\left(\frac{H\times W\times MAX_I^2}{\sum_{i=0}^{H-1}\sum_{j=0}^{W-1}[R(i,j)-G(i,j)]}\right) \quad (5)$$

수식 (5)는 PSNR(Peak Signal-to-Noise Ratio)로 생성한 이미지의 Noise 비율을 계산하는 지표이다. 원본 이미지(R)와 생성한 이미지(G) 사이의 픽셀 간 차이를 평가하여 영상을 복원하는 과정에서 생긴 선명도 손실을 집중적으로 측정한다.

$$SSIM=\frac{(2\mu_R\mu_G+C_1)(2\sigma_{RG}+C_2)}{(\mu_R^2+\mu_G^2+C_1)(\sigma_R^2+\sigma_G^2+C_2)} \quad (6)$$

수식 (6)은 SSIM(Structural Similarity Index Measure)으로, 데이터의 구조적인 유사성을 평가하는 지표이다. PSNR과 마찬가지로 원본 이미지(R)와 생성한 이미지(G) 사이의 차이에 집중하는데 각 집단의 분포를 고려하고자 통계량을 이용한다.

$$LPIPS=\sum_l\frac{1}{H_lW_l}\sum_{h,w}\left|w^l\odot\left(\hat{y}^l_{R,hw}-\hat{y}^l_{G,hw}\right)\right|_2^2 \quad (7)$$

LPIPS(Learned Perceptual Image Patch Similarity)는 AlexNet, VGG, SqueezeNet 모델의 중간 특성 맵을 활용한다. 수식 7처럼 각 모델에서 원본 이미지(R)와 생성한 이미지(G)의 차이에 대한 가중 평균의 평균으로 계산된다. LPIPS는 좋은 성능을 내는 거대 분류기를 통해 이미지의 특성을 추출하고 이를 기반으로 이미지 간의 차이를 계산한다는 점에서 FID(Fréchet Inception Distance)와 유사하다. 그러나 FID는 Inception 모델, LPIPS는 AlexNet, VGG, SqueezeNet 모델을 이용한다는 차이가 있다. 이 차이로 인해 LPIPS는 인간이 구조적으로 이미지를 비교하는 정도를 수치로 나타내는 평가 지표로 활용된다.

4-3 객체 제거 성능 분석

본 논문에서 제안한 시스템은 입력 이미지에서 제거할 사람 객체를 탐지하고 마스킹하는 과정을 포함하고 있다. 그러나 기존의 인페인팅 모델은 이미 마스킹 된 이미지를 입력으로 받기 때문에 이를 이용하기 위해선 마스킹 된 이미지가 필요하다. 이에 제안한 시스템에서 마스킹한 이미지를 저장하여 다른 모델과 비교하는 데 활용하였다. 객체 탐지 부분에서 탐지한 사람 객체 중, 가장 큰 객체이거나 임의의 한 객체를 중심 객체로 선정하였으며, 이에 따라 그림 6에서 볼 수 있듯이 마스킹 처리된 이미지와 이에 대한 마스크를 생성하였다. 이로 인해 인페인팅 결과, 주변에 불필요한 사람 정보는 제거되고 이미지에는 중심객체인 사람 한 명만 존재하게 된다.

그림 6. 마스크 생성 결과
Fig. 6. Generated mask images

그림 7은 제안한 객체 제거 시스템의 결과물로서 홀수 열은 입력 이미지를, 짝수 열은 출력 이미지를 보여준다. 이 결과를 확인해 보면, 입력 이미지의 전체적인 색감과 구성을 유지하고 있으며 배경의 패턴을 영역으로 연장하여 이미지를 자연스럽게 생성함을 알 수 있다. 더불어 단일 객체뿐 아니라 다중 객체도 제거한 결과를 시각적으로 확인할 수 있다. 그러나 탐지된 정보에 따른 박스 마스크를 이용하면서 배경과 객체가 제대로 분리되지 않아 배경 객체의 정보를 일부 잃어버린 경우가 존재했으며 이로 인해 다소 우그러진 형태로 복원되기도 하였다.

표 2는 기존 모델과 제안한 시스템의 인페인팅 성능을 정리한 것이다. FPS를 제외한 모든 평가 지표에서 IA 모델의 성능이 가장 뛰어나나, 비교한 모든 모델 중 모델의 추론 시간이 가장 길다. Deepfillv2의 경우, FPS가 가장 높게 측정되었으나, 제안한 시스템보다 PSNR, SSIM은 작은 값을, LPIPS는 큰 값을 보였다. LaMa는 전반적으로 제안한 시스템과 유사한 성능을 보였다. PSNR의 경우, 모델별 인페인팅 결과 이미지에서 번짐 현상을 가장 많이 보인 Deepfillv2의 측정값이 가장 높게 관측되었다. 이는 PSNR이 픽셀 간 차이 값이 인간의 지각적 판단과 정확하게 비례하지 않기 때문이다[17]. 그러나 제안한 시스템의 PSNR이 Deepfillv2의 PSNR과 큰 차이를 보이지 않으므로, 결과적으로 제안한 시스템이 상당히 자연스럽게 이미지를 복원하고 있음을 알 수 있다. 나아가 종합적인 이미지 복원 성능을 FPS와 비교할 때 제안한 시스템은 FPS 대비 가장 높은 복원 성능을 보였다. 표 3은 비교 실험에 이용한 모델별로 다중 객체 제거 시스템에 관한 기능을 정리한 것이다.

표 2. 인페인팅 성능 비교
Table 2. Comparing inpainting performance

	PSNR ↑	SSIM ↑	LPIPS ↓	FPS ↑
Deepfillv2[10]	34.495	0.529	0.170	12.16
LaMa[11]	34.970	0.630	0.158	8.32
IA-Remove[12]	35.800	0.721	0.150	2.40
proposed	35.145	0.600	0.167	8.30

표 3. 인페인팅 성능 비교
Table 3. Comparing inpainting performance

	Deepfillv2	LaMa	IA-Remove	proposed
1) main object select	✗	✗	✗	✓
2) multi object	✓	✓	✗	✓
3) object masking	✗	✗	✓	✓
4) video input	✗	✗	✓	✓

그림 8은 모델에 따른 인페인팅 결과물이다. 1행은 마스킹 전 이미지, 2행부터 5행은 각각 Deepfillv2, LaMa, IA 모델과 제안한 시스템의 결과 이미지이다. 비교한 모든 모델이 작은 객체에 대해선 자연스러운 이미지를 생성하였다.

Deepfillv2의 경우, 제거하고자 하는 객체가 커질수록 배경 정보를 제대로 활용하지 못해 눈에 띄게 우그러지는 경향을 보였다. 반면 LaMa는 객체 크기에 영향을 받지 않았으며, 배경에 전반적인 패턴을 이용해 이미지를 자연스럽게 생성하였다. 그러나 2열처럼 지우고자 하는 객체를 일부 남기는 경우가 존재했다. IA 모델은 Segmentation을 일부 진행함에 따라 3, 4열처럼 배경에 존재하는 객체를 살리기도 하였다. LaMa와 IA 모델은 5, 6열처럼 복원하고자 하는 영역에 작은 객체가 존재하는 경우, 이미지를 제대로 복원하지 못했다. 특히 6

그림 7. 제안한 시스템의 인페인팅 결과
Fig. 7. Inpainting results

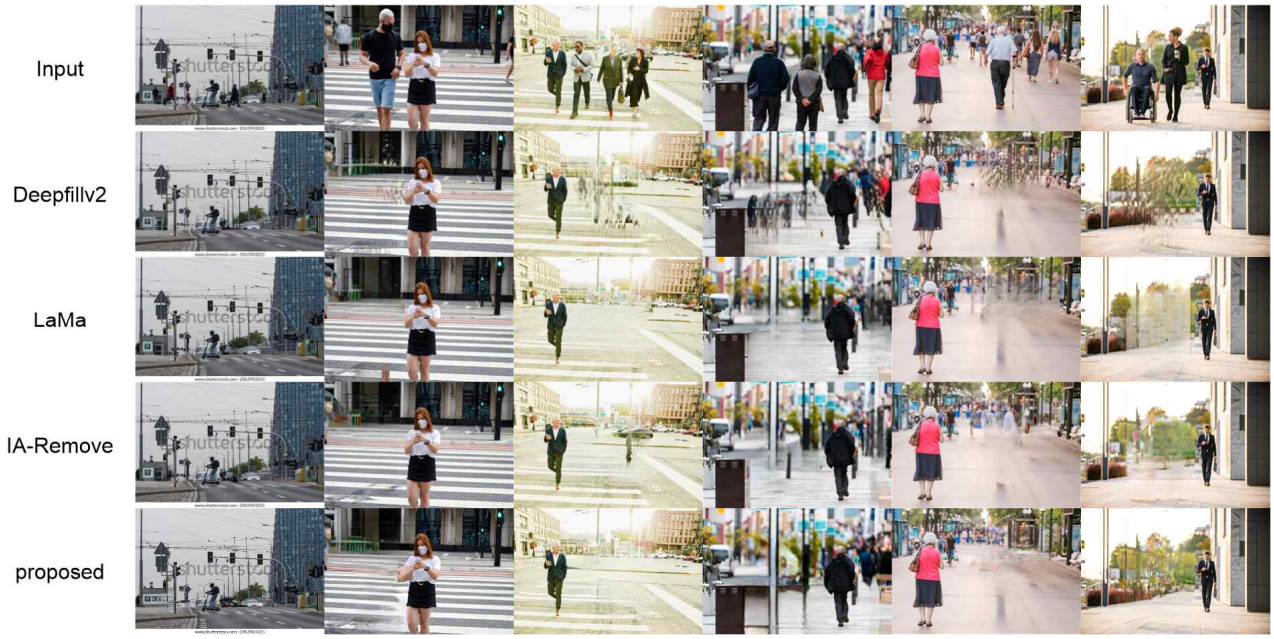

그림 8. 인페인팅 결과 이미지 비교
Fig. 8. Comparing inpainting results

열의 경우, 주변의 평균값이나 특이값으로 채운 듯한 이미지가 생성되었다. 이와 달리 제안한 시스템의 결과 이미지는 확실한 색 차이를 보여주는 경우, 과감히 색을 유지하고 끝맺음으로써 전반적으로 더 자연스러운 이미지를 생성하였다.

V. 결 론

본 논문은 영상의 불필요한 객체를 효율적으로 제거하기 위해 객체 추적 기술과 이미지 인페인팅 기술을 결합한 다중 객체 제거 시스템을 제안하였다. 이를 위해 YOLOv7과 객체 추적 모델 FairMOT을 통해 영상의 중심 객체를 추적하였고, 이를 제외한 나머지 객체를 마스킹하고 인페인팅 기술을 도입하여 객체를 영상에서 제거하였다. 적합한 손실함수를 설계하여 인페인팅 모델의 성능은 유지하되 FPS를 높일 수 있었다.

기존의 인페인팅 모델과 비교해 전반적으로 더 나은 성능을 보여주었지만, 영상 내 다중 객체를 빠르게 추적하고자 바운딩 박스를 그대로 마스킹 처리하면서 영상이 왜곡되는 경우가 발생하였다. 또한, 중심 객체는 영상의 중심 부분에 위치한다는 전제하에 중심 객체를 단순하게 선정하고 있어서 복잡한 장면에서는 중심 객체 탐지 오류가 발생할 수 있다. 향후 연구에서는 중심 객체를 선정하는 기준에 다양한 특징을 추가하여 중심 객체 선정 방법을 개선할 필요가 있다. 또한, 객체 탐지뿐만 아니라 인페인팅 시에도 이전 프레임을 이용하여 동영상에 대한 다중 객체 제거 시스템의 성능을 높이는 방법을 연구하고자 한다.

참고문헌

[1] C. Faklaris, F. Cafaro, A. Blevins, M. A. O'Haver, and N. Singhal, "A Snapshot of Bystander Attitudes about Mobile Live-Streaming Video in Public Settings," *Informatics*, Vol. 7, No. 2, 10, March 2020. https://doi.org/10.3390/informatics7020010

[2] A. Bewley, Z. Ge, L. Ott, F. Ramos, and B. Upcroft, "Simple Online and Realtime Tracking," in *Proceedings of 2016 IEEE International Conference on Image Processing (ICIP)*, Phoenix: AZ, pp. 3464-3468, September 2016. https://doi.org/10.1109/ICIP.2016.7533003

[3] J. Yu, Z. Lin, J. Yang, X. Shen, X. Lu, and T. S. Huang, "Generative Image Inpainting with Contextual Attention," in *Proceedings of 2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition,* Salt Lake City: UT, pp. 5505-5514, June 2018. https://doi.org/10.1109/CVPR.2018.00577

[4] S. Weder, G. Garcia-Hernando, Á. Monszpart, M. Pollefeys, G. Brostow, M. Firman, and S. Vicente, "Removing Objects from Neural Radiance Fields," in *Proceedings of 2023 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, Vancouver, Canada, pp. 16528-16538, June 2023. https://doi.org/10.1109/CVPR52729.2023.01586

[5] R. Shetty, M. Fritz, and B. Schiele, "Adversarial Scene Editing: Automatic Object Removal from Weak Supervision," arXiv:1806.01911, June 2018. https://doi.org/

10.48550/arXiv.1806.01911

[6] Y. Zhang, C. Wang, X. Wang, W. Zeng, and W. Liu, "FairMOT: On the Fairness of Detection and Re-Identification in Multiple Object Tracking," *International Journal of Computer Vision,* Vol. 129, No. 11, pp. 3069-3087, November 2021. https://doi.org/10.1007/s11263-021-01513-4

[7] Z. Wang, L. Zheng, Y. Liu, Y. Li, and S. Wang, "Towards Real-Time Multi-Object Tracking," in *Proceedings of the 16th European Conference on Computer Vision (ECCV 2020)*, Glasgow, UK, pp. 107-122, August 2020. https://doi.org/10.1007/978-3-030-58621-8_7

[8] N. Alkobi, T. R. Shaham, and T. Michaeli, "Internal Diverse Image Completion," in *Proceedings of 2023 IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops (CVPRW)*, Vancouver, Canada, pp. 648-658, June 2023. https://doi.org/10.1109/CVPRW59228.2023.00072

[9] N. Darapaneni, V. Kherde, K. Rao, D. Nikam, S. Katdare, A. Shukla, ... and A. R. Paduri, "Contextual Attention Mechanism, SRGAN Based Inpainting System for Eliminating Interruptions from Images," arXiv:2204.02591, April 2022. https://doi.org/10.48550/arXiv.2204.02591

[10] J. Yu, Z. Lin, J. Yang, X. Shen, X. Lu, and T. Huang, "Free-Form Image Inpainting with Gated Convolution," in *Proceedings of 2019 IEEE/CVF International Conference on Computer Vision (ICCV)*, Seoul, pp. 4470-4479, October-November 2019. https://doi.org/10.1109/ICCV.2019.00457

[11] R. Suvorov, E. Logacheva, A. Mashikhin, A. Remizova, A. Ashukha, A. Silvestrov, ... and V. Lempitsky, "Resolution-Robust Large Mask Inpainting with Fourier Convolutions," in *Proceedings of 2022 IEEE/CVF Winter Conference on Applications of Computer Vision (WACV)*, Waikoloa: HI, pp. 3172-3182, January 2022. https://doi.org/10.1109/WACV51458.2022.00323

[12] T. Yu, R. Feng, R. Feng, J. Liu, X. Jin, W. Zeng, and Z. Chen, "Inpaint Anything: Segment Anything Meets Image Inpainting," arXiv:2304.06790, April 2023. https://doi.org/10.48550/arXiv.2304.06790

[13] GitHub. ifzhang/FairMOT [Internet]. Available: https://github.com/ifzhang/FairMOT.

[14] C.-Y. Wang, A. Bochkovskiy, and H.-Y. M. Liao, "YOLOv7: Trainable Bag-of-Freebies Sets New State-of-the-Art for Real-Time Object Detectors," in *Proceedings of 2023 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, Vancouver, Canada, pp. 7464-7475, June 2023. https://doi.org/10.1109/CVPR52729.2023.00721

[15] D. Sharma, T. Hade, and Q. Tian, "Comparison of Deep Object Detectors on a New Vulnerable Pedestrian Dataset," arXiv:2212.06218v1, December 2022. https://doi.org/10.48550/arXiv.2212.06218

[16] S. Zhang, Y. Xie, J. Wan, H. Xia, S. Z. Li, and G. Guo, "Widerperson: A Diverse Dataset for Dense Pedestrian Detection in the Wild," *IEEE Transactions on Multimedia*, Vol. 22, No. 2, pp. 380-393, February 2020. https://doi.org/10.1109/TMM.2019.2929005

[17] K. Suresh and U. Sakthi, "Robust Multi-Thresholding in Noisy Grayscale Images Using Otsu's Function and Harmony Search Optimization Algorithm," in *Proceedings of the 1st Springer International Conference on Emerging Trends and Advances in Electrical Engineering and Renewable Energy (ETAEERE 2016)*, Majitar, India, pp. 491-499, December 2016. https://doi.org/10.1007/978-981-10-4765-7_52

이효진(Hyo-Jin Lee)

2023년 : 성신여자대학교 대학원 미래융합기술공학과(공학석사)
2021년 : 성신여자대학교 정보시스템공학과(공학사)

※관심분야 : 이미지처리, 딥러닝, 인페인팅

변혜원(Hae-Won Byun)

2004년 : KAIST 대학원 (공학박사-컴퓨터그래픽스)
1992년 : KAIST 대학원 (공학석사)
1990년 : 연세대학교 전산과학과 (공학사)

2006년～현 재: 성신여자대학교 AI융합학부 교수

※관심분야 : 컴퓨터 그래픽스, 멀티모달 딥러닝, 생성형 AI 등