

YOLACT를 이용한 스티칭 속도 개선 방안

고성영 이성배 박성환 김규헌

경희대학교

kosy0413@khu.ac.kr rhee@khu.ac.kr gocheenee@khu.ac.kr kyuheonkim@khu.ac.kr

Stitching speed improvement method using YOLACT

Go, Sung_Young Rhee, Seong_Bae Park, Seong_Hwan Kim, Kyu_Heon

Kyunghee University

요약

최근 초고화질 영상, 가상현실 등 프리미엄 콘텐츠에 대한 요구가 커지면서 360° VR과 8K TV 등의 시장이 확대되고 있다. 360° VR 영상을 만드는 데에 스티칭 기술이 사용되고 있고, 8K 영상을 촬영할 수 있는 장비는 매우 제한적이기 때문에 스티칭 기술을 통해 콘텐츠를 확보하려는 노력이 이어지고 있다. 스티칭 기술은 여러 영상을 합성하여 기존 카메라의 좁은 시야각 문제를 해결하고 보다 넓은 시야각의 영상을 만드는 기술이다. 최근에는 해당 분야에 관한 연구가 진행됨에 따라 이미지를 넘어 동영상 스티칭에 대한 연구가 주로 진행되고 있다. 기존 동영상 스티칭 방식은 이미지 스티칭 방식을 프레임마다 반복하기 때문에 시간이 오래 걸린다는 단점이 있다. 컴퓨터 비전 분야에서는 딥러닝을 활용하여 객체가 존재할 것으로 예측되는 부분에 사각형 모양의 경계 상자(Bounding box)를 생성하는 객체 탐지(Object detection) 분야에 관한 많은 연구가 이루어져 왔고 이를 기반으로 객체의 경계선을 검출하여 해당 영역만을 구분하는 객체 분할(Instance segmentation)에 대한 연구 또한 진행 중이다. 본 논문에서는 앞서 말한 스티칭 속도 문제를 해결하기 위하여 빠른 속도로 객체 분할이 가능한 YOLACT를 이용하여 스티칭 속도를 개선하는 방안을 제안한다.

1. 서론

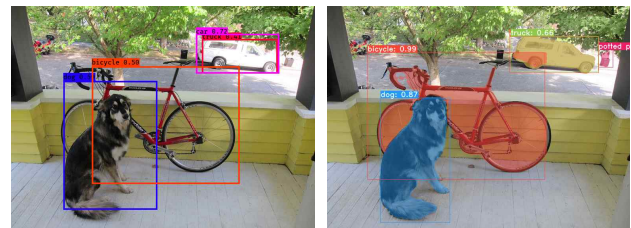
최근 초고화질 영상, 가상 현실 등 프리미엄 콘텐츠에 대한 요구가 커지면서 파노라마 및 360° VR(Virtual Reality) 영상이 주목받고 있으며, 가전 시장에서는 8K TV가 출시되면서 초고화질 시장이 확대되고 있다.^{[1][2]} 한 대의 카메라를 통해서 파노라마 및 360° VR 영상을 만드는 데에는 시야각(Field of View)이 제한되므로, 다수의 카메라로 촬영한 여러 영상을 하나의 영상으로 합성하는 영상 스티칭 기술이 사용되고 있다^[3].

기존 스티칭 기술은 다수의 정지 영상을 제대로 합성하기 위해 제안되었지만, 다양한 콘텐츠를 필요로 하는 사용자의 요구에 따라 동영상 스티칭에 대한 연구 또한 진행되고 있다. 그러나 기존 정지 영상을 기반으로 스티칭하는 기술은 동영상의 한 프레임을 합성하는 데 시간이 오래 걸리고, 이에 따라서 동영상 하나를 스티칭하는 데에 많은 시간이 소요되어 빠른 속도로 영상을 스티칭하는 기술이 필요하다.

한편, 동영상 기반의 스티칭 기술에서는 영상 내 움직이는 사물로 인해 촬영 카메라 간에 시차(Parallax)가 발생하고 이에 따라서 시차 왜곡이 발생할 수 있다. 시차 왜곡을 극복하기 위하여 동영상 스티칭 기술에서는 동영상 내에서 움직이는 사물을 검출하기 위해 객체 탐지 기술(Object detection)을 사용하고 있다.^[4]

객체 탐지 기술은 객체가 존재할 것으로 예측되는 부분에 대하여 사각형 모양의 경계 상자(Bounding box)를 생성하는 기술로써, 최근에는 딥러닝을 기반으로 한 YOLO(You Only Look Once)^[5], SSD(Single Shot Detector)^[6] 등의 객체 탐지 기술들이 대표된다. 또한, 앞선 객체 탐지 기술을 기반으로 한 객체 분할(Instance segmentation) 기술은 객체 모양대로 해당 영역만을 추출하는 기술로써, YOLACT(You Only Look At CoefficientTs)^[7], Mask R-CNN(Mask Regions with CNN) 등이 대표된다.

이 중 YOLACT는 실시간으로 객체를 탐지하고 객체를 분할하는 방법으로써, 빠른 속도로 영상 내 사물 영역만을 떼어내는 장점이 존재하므로, 기존 정지 영상 기반의 스티칭 기술을 통해서 동영상을 스티칭하



(a) YOLO v3

(b) YOLACT

그림 1. YOLO와 YOLACT를 사용한 객체 탐지 및 객체 분할

는 것이 아닌, 추출한 객체 영상을 스티칭 영상에 블렌딩하는 방식을 통해 빠른 속도의 영상 스티칭 기술을 본 논문에서 제안하고자 한다.

이에 본 논문의 2장에서는 기존 스티칭 방법에 대하여 간단히 분석하고, 3장에서는 YOLACT를 기반으로 하는 스티칭 속도 개선 방법을 제안한다. 또한, 4장에서는 테스트 영상을 통한 제안 방법과 기존 스티칭의 비교 실험 결과를 통해 제안 기술을 검증하며, 5장에서는 결론을 맺는다.

2. 기존 스티칭 방법 분석

2.1 일반적인 스티칭 알고리즘

하나의 객체를 둘 이상의 위치에서 촬영하면, 해당 객체는 촬영한 위치에 따라 서로 다른 좌표계에서 얻어지게 된다. 이처럼 서로 다른 좌표계를 하나의 좌표계로 통합하는 것을 영상 정합 (Image registration)이라고 한다. 영상 정합은 이미지 간의 대응 관계를 찾기 위해 주위 배경과 구분되면서 식별이 용이한 지점인 특징점(Feature)을 추출하고, 추출된 특징점들의 상관관계를 찾아냄으로써 매칭한 후, 매칭된 특징점 사이의 변환 관계를 찾아 중복 영역을 찾는 단계를 거친다. 이후 이를 자연스럽게 합성하기 위해 가중치를 할당하여 보정하는 블렌딩을 진행한다.[8][9]

2.2 최소 오류 경계를 활용한 스티칭 알고리즘^[10]

최소 오류 경계를 활용한 스티칭은 특징점 매칭을 통해 찾은 중복 영역에서 합쳐진 경계(seam)가 부자연스럽게 정합되는 문제를 해결하기 위해 경계선에서의 오류를 최소화하여 연결하는 방법이다. 중복 영역 픽셀의 회색조 값의 차를 제공하여 Error 행렬을 만들고 영상의 가장 윗단에서부터 아래 방향의 좌측, 원위치, 우측 중 가장 작은 값을 최하단까지 더하여 이 중에서 가장 낮은 값을 선택한다. 이를 아래에서부터 역으로 따라 올라가면 최소 오류 경계를 찾을 수 있다. 이 과정을 HSV를 기반으로 하여 3번 진행하고, 블렌딩 과정에서도 다양한 모듈을 통해 영상이 자연스럽게 정합되도록 한다.

기존 스티칭 방식을 동영상에 적용하게 되면 프레임마다 위 과정을 반복해야 하므로 처리 시간이 대폭 늘어나게 된다. 이에 본 논문에서는 실시간으로 객체를 분할할 수 있는 YOLACT를 이용한 스티칭 속도 개선 방안을 3장에서 제안한다.

3. YOLACT를 이용한 스티칭 방법

본 논문에서 제안하는 YOLACT를 이용한 스티칭 속도 개선 방안은 최소 오류 경계 기반 스티칭 방법으로써, 중복 영역에 대하여 seam을 생성하고 이를 통하여 그림 2와 같은 배경 프레임의 생성한다. 이때 동영상의 첫 번째 프레임에 대하여 생성된 seam은 끝날 때까지 고정적으로 사용한다.



그림 2. 최소 오류 경계를 활용해 정합한 배경 프레임

이후 움직이는 객체에 대해서는 각각 입력 영상의 중복 영역에 대하여 YOLACT를 통해 그림 3과 같이 객체를 분할한다.

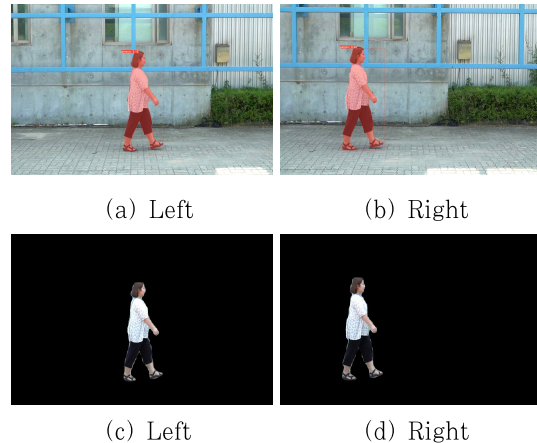


그림 3. YOLACT를 사용해 객체 분할한 영상

그림 4의 (a), (b)는 좌우 영상에서 각각 YOLACT를 사용해 객체를 이진화하여 분할한 영상이다. 두 영상을 단순히 겹치면 시차(parallax)로 인해 그림 4의 (c)처럼 객체가 서로 다른 위치에 나타나기 때문에 좌우 영상 중 하나를 선택하여 사용한다.

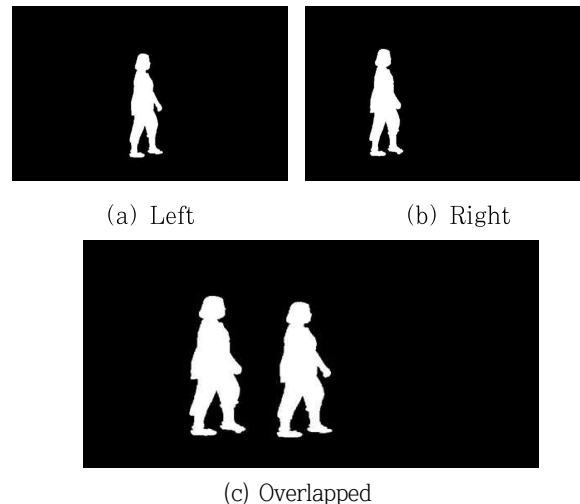
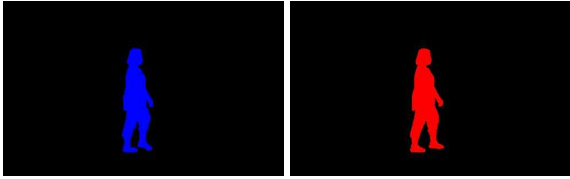


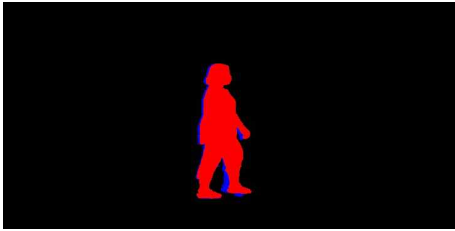
그림 4. YOLACT를 사용해 이진화하여 객체 분할한 영상

그림 5의 (a), (b)는 선택된 영상에서 이전 프레임과 현재 프레임, (c)는 두 프레임을 겹친 것이다. 오버랩된 영상에서 파란색으로 나타난 부분은 이전 프레임, 빨간색으로 나타난 부분은 현재 프레임이다. 움직

이는 객체에 대해서, 이전 프레임에 해당하는 위치에는 배경을, 현재 프레임에서 객체가 위치하는 부분에는 분할한 객체를 가져오는 방식을 사용한다. 이후 블렌딩을 통해 배경 프레임에 옮긴 후 그림 6과 같이 스티칭 된 결과 영상을 만든다.



(a) Previous frame (b) Present frame



(c) Overlapped frame

그림 5. 선택된 영상에서 프레임 간 객체 이동



그림 6. 결과 영상

4. 실험 결과

3절의 내용을 바탕으로 GeForce GTX 1180를 사용하는 ubuntu 18.04 환경에서 코드를 제작하였다. anaconda3 4.8.3 버전, python 3.7.7 버전, opencv 3.4.2 버전을 이용하였고 backbone으로는 Resnet101-FPN을 사용하였다.

분류	배경 프레임 생성 시간 및 YOLACT 환경설정 시간	스티칭 영상 생성 시간 (40frames)	fps
최소 오류 경계 방법	x	43.8sec	0.91
제안 방안	1.73sec	5.19sec	7.71

표 1. 속도 비교 결과

YOLACT를 이용하면 550×550 크기의 영상에 대하여 30fps 이상으로 객체를 분할할 수 있다. 본 논문에서는 1920×1080 크기의 영상을 입력 영상으로 사용하여 중복 영역은 1667×1065로 계산되었다. 계산된 중복 영역에 대하여 YOLACT를 이용한 결과, 영상 크기의 차이로 인해 실시간에는 미치지 못하지만 기존 최소 오류 경계를 활용한 방법에 비하여 스티칭 속도를 크게 개선하였다.

5. 결론

본 논문에서는 프레임마다 특징점을 추출 및 매칭하고 seam을 생성하여 정합하는 기존 이미지 스티칭 방식을 사용할 때 처리 시간이 오래 걸리는 문제를 해결하기 위하여 실시간 객체 검출 기술인 YOLACT를 적용하는 방법을 제시하였다. 배경 프레임에서는 기존 최소 오류 경계를 활용한 스티칭 방식대로 seam을 생성하고, 이후 프레임에서는 생성된 seam을 고정적으로 사용해 중복 영역에서 YOLACT를 이용해 객체를 분할하여 가져오는 방법으로 속도 문제를 개선하였다. 해당 기술은 배경을 미리 촬영할 수 있는 공연이나 스포츠 경기 등에 적용할 경우 빠르게 스티칭 영상을 제공할 수 있을 것으로 예상된다.

한편 객체를 분할하지 못할 경우에는 객체를 마스크하여 사용할 수 없어 객체가 사라지는 문제가 존재하며 이는 객체 분할 정확도가 더 높아진다면 개선될 것이다. 또한, 카메라 간 서로 다른 광학 원점으로 인한 시차 문제까지는 해결하지 못해 향후 이에 대한 추가연구를 진행할 예정이다.

* 본 연구는 한국전력공사의 2018년 착수 에너지 거점대학 클러스터 사업에 의해 지원되었음 (과제번호:R18XA02)

참고문헌

- [1] Hyeon-Deok Han, Jong-Ki Han, "Modified Seam Finding Algorithm based on Saliency Map to Generate 360 VR Image", The Korean Institute of Broadcast and Media Engineers, Vol.24, No.6, Nov, 2019
- [2] 8K TV competition'ignition'... Samsung vs. LG, Sony, Hisense, TCL, the winner?, <http://www.ddaily.co.kr/news/article/?no=190434> (accessed Jan, 18, 2020).
- [3] Woo-Kyung Jung, Jong-Ki Han. "360 VR Image Stitching Algorithm using Image Clustering based on Camera Sensor Data", The Korean Institute of Broadcast and Media Engineers
- [4] Wei, L. Y. U., et al. "A survey on image and video stitching." Virtual Reality & Intelligent Hardware Vol. 1, No.1 pp.55-83, 2019.
- [5] Joseph Redmon, Santosh Divvala, Ross Girshick, Ali Farhadi, "You Only Look Once: Unified, Real-Time Object Detection", CVPR 2016, 9, May, 2016
- [6] Wei Liu, Dragomir Anguelov, Dumitru Erhan,

- Christian Szegedy, Scott Reed, Cheng-Yang Fu, Alexander C. Berg, "SSD: Single Shot MultiBox Detector", ECCV2016, Dec, 29, 2016
- [7] Daniel Bolya, Chong Zhou, Fanyi Xiao, Yong Jae Lee . "YOLACT: Real-time Instance Segmentation", ICCV, 24, Oct, 2019
- [8] Vladan Rankov, Rosalind J. Locke, Richard J. Edens, Paul R. Barber, Borivoj Vojnovic, "An algorithm for image stitching and blending", Proceedings of SPIE -- Volume 5701, March 2005
- [9] Y.D. Lowe, "Distinctive image features from scale-invariant key- points," International Journal of Computer Vision, vol. 60, no. 2, pp.91-110, Nov. 2004.
- [10] Jeonho Kang, Junsik Kim, SangIL Kim, and Kyuheon Kim, "Method of Video Stitching based on Minimal Error Seam", The Korean Institute of Broadcast and Media Engineers, Vol.24, No.1, pp.142-152, January, 2019.