

## 경계 오류를 최소화한 동영상 블렌딩

신서영, 윤서현, 정희민, 김준식, 김규현  
경희대학교

coast407@naver.com, dbstjgus123@naver.com, scr02070@naver.com, junsik@khu.ac.kr,  
kyuheon.kim@khu.ac.kr

### Video Blending based on Minimal Error Seam

Seoyoung Shin, Seohyeon Yoon, Heemin Jung, Junsik Kim, Kyuheon Kim  
Kyung Hee Univ.

#### 요 약

현재 기술의 블렌딩 된 동영상을 재생하는 과정에서 사용하는 기법은 각각의 이미지들을 스티칭 후 동영상을 재생하므로 이전 프레임의 경계를 고려하지 않는다. 이러한 과정에서 움직임이 동일한 경계 부분을 정확하게 파악하여 연결하지 못하는 오류가 발생하여 자연스러운 영상 재생이 불가능하다. 이에 대한 해결책으로 각 영상의 frame 을 추출하여 경계의 가중치를 비교한 후 차이가 클 경우 새로운 경계를 생성하여 적용하는 과정을 고안했다. 상기의 과정을 통해 블렌딩 된 동영상의 초기 오류가 크게 감소하였다. 본 논문에서는 해당 기술에 대한 구체적인 방안을 제시하여 다양한 동영상을 효과적으로 개선하여 오류를 최소화한 동영상을 얻을 수 있도록 하였다.

#### 1. 서론

최근 파노라마 서비스, VR 콘텐츠 뿐만 아니라 2018 년도 평창 올림픽에서 선보인 것과 같이 360 도 카메라를 통한 실시간 VR 중계의 선택적 시청 등 사용자의 시점에 맞춘 영상의 수요가 지속적으로 증가하고 있다. 하지만 촬영 도구의 화각이 제한되어 있기 때문에 현재 사용중인 도구를 바탕으로 여러 화면을 촬영 후 그 두 영상을 잇는 비디오 스티칭이 필수적이다. 이 중 파노라마 서비스나 VR 콘텐츠와 같은 경우 후 작업을 통한 오류 수정이 어느 정도 가능하지만, 실시간 VR 중계의 경우 그에 대한 해결책이 없는 실정이다. 따라서 사용자 입장에서 보기 자연스러운 비디오 블렌딩 기법이 더욱 필요한 상황이다.

기존의 방식인 이미지 스티칭을 통한 동영상 블렌딩 기법은 영상에서 각각의 frame 들을 추출하여 이 frame 들을 붙이는 데만 치중하였다. 이에 따라 앞 frame 의 seam line 은 고려하지 않는 것을 알 수 있었다. 그러나 이런 방식으로 진행할 경우 경계의 가중치에 대한 비교가 이루어지지 않아 정확한 영상을 재생해 내는데 문제가 발생하여 자연스러운 영상을 시청할 수 없게 된다.

Image Stitching 이란 두 개의 frame 을 붙이는 것을 의미하고 합쳐진 부분에서 생겨난 경계를 seam 이라 일컫는다. 이와 비슷한 Video Stitching 은 두 영상에서 frame 을 추출 후 Image Stitching 방식에 의해 두 frame 을 스티칭하는 것을 말한다. 이 이미지들을 이어 붙인 후 다시 비디오로 제작하여 재생할 수 있게 된다. Video Blending 은 Video Stitching 에 속하는 과정으로 스티칭한 부분에 있어서 부자연스러운 부분을

자연스럽게 만드는 과정 자체를 말한다.

본 논문에서는 이미지들을 스티칭 하여 n-1 번째 frame 과 n 번째 frame 사이의 seam 차이가 큰 경우 n-1 번째 seam 을 참고하여 n 번째 frame 을 스티칭하는 과정에 적용한다. 본 논문에서는 seam 의 오류를 최소화 할 수 있는 아이디어와 그에 해당하는 알고리즘을 제시한다.

본 논문의 구성은 다음과 같다. 2 절에서는 현재 비디오 스티칭 기술에서 생겨날 수 있는 오류를 확인하고 이에 기반하여 새로운 알고리즘을 제시한다. 3 절에서는 본 논문에서 제안하는 기법에 관한 기술을 적용한 결과를 선보인다. 마지막으로 4 절에서는 본 논문에 대한 결론을 맺는다.

#### 2. Video Stitching 오류

본 논문에서 언급하는 비디오 스티칭은 두 대의 카메라에서 얻은 비디오를 하나의 영상으로 만드는 작업을 의미한다. 스티칭 작업은 다음과 같다. 먼저 두 개의 입력 영상의 각 frame 에서 특징점을 추출한다. 다음으로 추출한 특징점 토대로 두 영상을 매칭하여 겹치는 부분을 찾아낸다. 그리고 그 겹쳐진 영역에서 최소 오류를 갖는 seam 을 구하고 그 seam 을 경계로 두 영상을 정합한다. 이 과정을 각 frame 마다 수행해주는 것이 video stitching 의 과정이라고 할 수 있다. 영상으로 정합하는 과정에서 각각의 영상에서 frame 을 추출하여 재생하였을 경우에는 영상이 자연스러우나 두 영상을 Stithcing 하여 영상으로

정합한 후 다시 영상을 재생하였을 때에는 아래 그림 1, 그림 2 와 같이 부자연스럽게 순간이동하는 현상이 발생하게 된다.

그림 1 은 92 번째 frame 의 이미지를 추출한 것이며 그림 2 는 바로 다음 frame 인 93 번째 frame 의 이미지를 추출한 것이다.



그림 1. 오류 영상 (92<sup>nd</sup> frame)

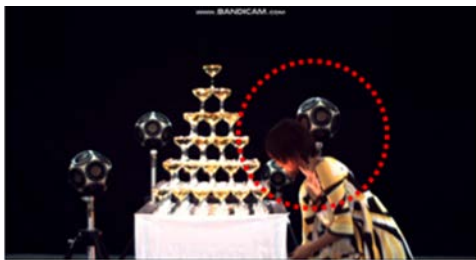


그림 2. 오류 영상 (93<sup>rd</sup> frame)

해당 영상에서의 오류가 발생한 원인을 찾고자 seam 정보를 추출해 보았다.



그림 3. seam 정보 (92<sup>nd</sup>, 93<sup>rd</sup> frame)

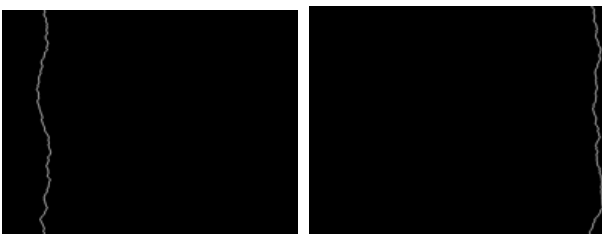


그림 4. seam 정보 (168<sup>th</sup>, 169<sup>th</sup> frame)

크게 변화가 존재하는 부분은 총 2 군데로, 해당 영상에서 오류가 발생한 부분은 각각 92 번째와 92 번째, 그리고 168 번째와 169 번째의 seam line 에서 급격한 변동이 발생한 것을 볼 수 있다. 이러한 seam line 의 변화로 두 개의 영상이 정합되는 경계가 달라져 소위 순간이동과 같은 부자연스러운 현상이 나타난 것을 알 수 있다. seam line 의 급격한 변동으로 두 개의 영상의 경계가 달라지기 때문에 소위 순간이동 현상과 같은 부자연스러운 현상이 나타나게 된다.

Video Stitching 과정에서 발견한 위와 같은 error 는

$n-1$  번째 frame 과  $n$  번째 frame 이 정합되는 과정에서 변화되는 부분에 대한 프로그램 상 인지의 오류로 인해 발생되었다. 그에 따라 각각의 seam line 에 대한 급격한 변동을 방지하는 방법에 관한 새로운 아이디어를 고안하게 되었다.

본래 Video Blending 이 두 개의 영상들을 합쳐서 그대로 Blending 을 진행한 방식이라면 본 논문에서 제안하는 방식은 좀 더 명확한 seam 을 얻고자 이후에 새로 검토하는 과정을 도입하였다. 아래는 본 논문에서 진행한 영상을 토대로 설명한 알고리즘 구동 방식이다.

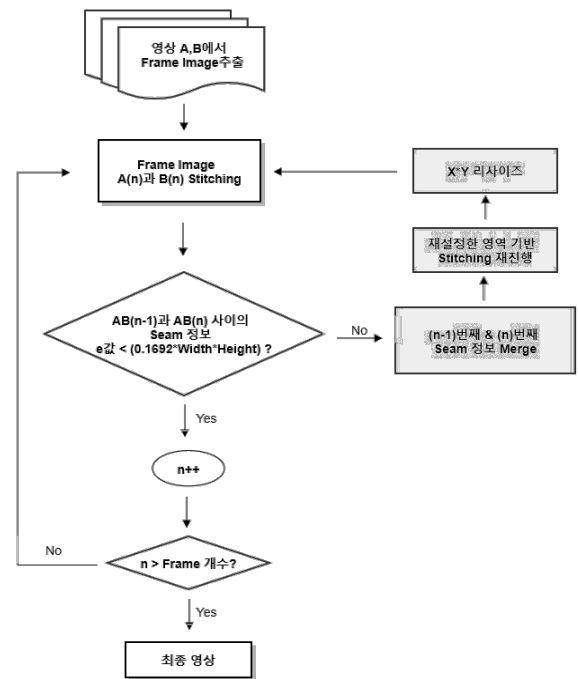


그림 5. 본 논문에서 제안하는 알고리즘

우선 영상에서 frame image 를 추출해 낸다. 이렇게 추출한 image 들을 Stitching 한다. 그리고 if 문을 넣어 만약 Stitching 된 이미지의 seam 정보가 일정 크기보다 작으면 다음 Frame 으로 넘어가서  $n$  과  $n+1$  frame 의 Stitching 을 진행한다. 그러나 만약 일정 크기보다 크면  $n-1$  번째와  $n$  번째 seam 정보를 merge 한다. 그렇게 merge 된 seam 정보는 한 frame 에 두 개의 seam line 이 생기게 된다. 해당 영상에서는  $n-1$  번째 frame 의 seam line 이 우측,  $n$  번째 frame 의 seam line 이 좌측에 위치한다. 그러므로 2 개의 seam line 중 우측에 위치한 seam line 의 가장 왼쪽 픽셀값을  $A(\min)$  으로 두고, 가장 오른쪽 픽셀값을  $A(\max)$  이라고 둔다. 또한 좌측에 위치한 seam line 의 가장 왼쪽 픽셀값을  $B(\min)$  으로 두고, 가장 오른쪽 픽셀값을  $B(\max)$  라고 둔다. 아래는 Merge 된 seam 정보의  $A(\min)$ ,  $A(\max)$ ,  $B(\min)$ ,  $B(\max)$  를 나타낸 그림이다.

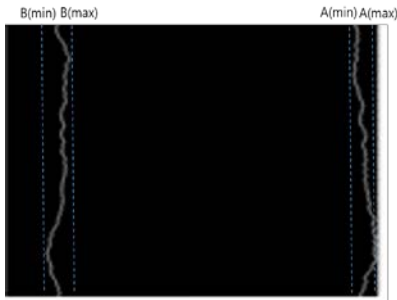


그림 6. Merge 된 seam 정보 (92nd, 93rd frame)

이렇게 Merge 된 Seam 정보 영역 전체의 width 를 X, height 를 Y 라고 둔다. 또한 전체 영역의 가장 좌측에서 A(max)까지의 width 를  $j_1$  이라고 두고, B(min)까지의 거리를  $j_2$  라고 둔다. 이를 이용하여 A(max)와 B(min)의 중간값을 지정하고 이를  $j'$ 으로 둔다. 이에 해당하는 식  $j' = (j_1 + j_2) / 2$  을 사용하여 정의한다. n-1 번째 frame 의 seam line 이 우측에 위치하므로, 영역을 재설정 할 때 기반이 되는 부분 또한 우측을 기준으로 한다. 즉 width 는  $X - j'$ , height 는 Y 인 영역이 생성된다. 이를 토대로 좌측의 비어있는  $j'$  영역 부분은 다시 width 가 X 가 될 수 있도록 빈 공간을 resize 시킨다. 이렇게 재설정 한 영역을 바탕으로 n 번째 seam 의 수정을 진행한다. 이렇게 수정된 seam 을 바탕으로 다시 n-1 과 n 번째 frame seam 정보를 계산해보고 일정 크기보다 작으면 해당 seam 을 바탕으로 n 번째와 n+1 번째 frame 의 Stitching 을 진행한다. 만약 새로 수정한 seam 정보가 일정 크기보다 클 경우 위 과정을 해당 범위 안에 속할 때까지 반복 진행한다.

해당 알고리즘에서 제시하는 일정 크기인 threshold 값은 e 를 바탕으로 표현하였으며 e 값은 seam 정보의 차이의 평균값을 픽셀 값을 모두 더해준 값과 같다. 정확한 값은 본 영상 자체에서 Stitching 을 적용한 frame 의 크기인 1326\*962 로 나누었다. 즉 추출한 e 값에 영상의 width 와 height 를 곱하고 이를 Stitching 적용 frame 값으로 나타낸 것이다. 정확한 식으로 추출한 결과  $e * (\text{width} * \text{height}) / (1326 * 962)$  으로 도출됨을 알 수 있다 즉 해당 알고리즘에서 쓰이는 e 값은  $0.1692 * \text{Width} * \text{Height}$  이다.

### 3. 결과

본문 3 절의 알고리즘을 바탕으로 Visual Studio 2013 을 이용하여 시중에 배포된 Open CV 2.4.10 를 적용하는 과정을 통해 코드를 제작하였다. 스티칭 과정에서 쓰이는 특징점 추출 알고리즘은 OpenCV 에서 제공하는 ORB 알고리즘과 flann 매칭 기법을 사용하였다.

아래 그림은 초기에 오류가 있었던 frame 들의 seam line 을 수정한 후 적용한 영상이다. 수정한 해당 영상에서는 92 번째 frame 에서 93 번째 frame 으로 이동할 때 순간이동 현상이 발생하지 않았다. 이에 따라 순간이동 현상이 지속되었던 93 번째 frame 부터

168 번째 frame 까지 전체 영상을 수정해본 결과 모든 부분에서 오류가 발생하지 않음을 볼 수 있었다.



그림 7. 오류 수정 영상 (92nd frame)



그림 8. 오류 수정 영상 (93rd frame)

초기 Image stitching 을 이용한 Video Stitching 과정에서 발생한 오류 영상을 본 논문에서 제안하는 방법으로 진행한 경우 오류가 수정된 영상의 모습으로 영상이 출력된 것을 확인할 수 있다.

### 5. 결론

영상을 Stitching 하는 기술을 적용한 분야가 증가함에 따라 더 빠르고 정확하게 영상을 시청자들의 편의에 맞추어 제작해야 한다. 뿐만 아니라 시청하길 원하는 사람들의 Stitching 에 대한 관심과 기대치 또한 증가하고 있다. 본 논문에서 제안하는 기술은 그 요구에 만족시키는 방향을 제시하였다. 해당 기술을 적용함으로써 블렌딩된 파노라마 영상의 이질감을 최소화하여 사용자에게 더욱더 현장감 넘치는 미디어 기술 서비스를 제공 가능할 것으로 보인다.

한편 본문에서 제안하는 기술은 Seam line 의 큰 변동에 해당 기술을 적용하기에는 적합한 것으로 보인다. 그러나 미세한 seam line 의 변동까지 완벽하게 잡기에는 여전히 오류가 존재한다. 추후 진행할 연구는 미세한 seam 의 변동까지 잡아주는 현재보다 더 상위 기술을 적용하는 방식을 고안하여 적용할 예정이다.

#### 참고문헌

[1] Dr. Mayada F. Abdul-Halim & Noor Adnan Ibraheem, UTILIZING GENETIC ALGORITHMS FOR

2D TEXTURE SYNTHESIS, 2015, 3pg

[2] Kyung-Ju Park, LV-RV Modeling Using a Parameterized Blended Model, 2007, 3pg

[3] Sangwoo Ahn & Sungwoong Ra & Jihun Cha, A study on design of panorama video rendering systems using ITF, 2pg