

Opencv를 통한 얼굴 이미지 오버래핑 촬영 어플리케이션

강원준

2009120245

응용프로그래밍
고려대학교

과제 목표

□ 해결하고자 하는 문제의 정의

Disclosure란 가수의 뮤직비디오(<https://www.youtube.com/watch?v=93ASUImTedo&feature=youtu.be&t=4m2s>)를 보다가 영감을 받아 기획 하였다. 뮤직비디오에 나와있는 것과 마찬가지로 동영상을 찍을 때 자동적으로 얼굴을 다른 이미지로 가릴 수 있는 어플리케이션을 개발하고자 한다. 보통 사적인 SNS보다 네이버 카페 같은 공개적 커뮤니티에 자신의 라이프나 전신 룩 스타일을 올리 고는 싶고 그렇지만 자신의 얼굴을 공개하고 싶지는 않은 사람들이 사진이나 동영상을 추후 작업 없이도 올릴 수 있도록 하고 싶었다.

□ 문제의 중요성 및 응용 분야

결국 얼굴을 인식해 그 위에 마스크나 다른 이미지들을 오버랩하는 것이 이 어플의 원리 이다. 얼굴을 인식하는 알고리즘은 `haarcascade.xml`의 오픈소스 알고리즘을 사용하 지만 다른 알고리즘들을 통해 동영상에서 특정 사물을 파악해 그 옆에 이미지 뿐만 아니라 특정 애니메이션을 오버랩한다면 가상현실의 효과를 낼 수 있을 것이라 생각된다.

기존 방법

□ 기존 방법 1

기존의 어플들을 보면 사진의 경우 이미 찍은 사진에 추후 처리 방식으로 이미지 스티커를 얼굴부위에 오버랩할 수 있도록 했다.

사진 촬영은 추후 처리 방식이 불편하지 않지만 동영상의 경우 이런 기능을 사용하기 위해 추후처리 방식이 크게 불편할 수 있다. 그렇기 때문에 기존에 동영상 같은 경우에는 이렇게 이미지 스티커를 중간에 넣어 동영상을 찍는 어플은 없었다. 또한 동영상의 경우 어떤 특정사물을 계속 인식하며 처리를 해줘야 하기 때문에 가상현실 어플 외에는 단순한 이미지를 올려 놓는 기능을 갖춘 카메라 어플을 찾을 수가 없었다.

그래서 저 뮤직비디오처럼 사람 얼굴에 이미지를 씌여주는 어플을 통해 더 다이나믹하고 재밌는 동영상을 만들 수 있을 것이라 예상했다.

구현 내용

□ 구현할 기존 해결 방법 (또는 제안하는 새로운 방법)

기존에 동영상에 스티커 이미지를 오버랩해 영상 녹화를 할 수 있는 어플이 없었다면 Hide Face는 OpenCV의 안드로이드 SDK를 사용하여 haarcascades.xml을 통해 얼굴을 인식하고 그 얼굴위에 재밋는 이미지를 올려서 다이나믹한 영상촬영을 손쉽게 할 수 있는 기능을 갖출 것이다. 또한 동영상 뿐만아니라 사진을 찍은 후 추후 처리 방식으로 이미지를 올려놓는 방식이 아니라 사전에 사진을 찍을 때부터 얼굴 크기에 맞춰 이미지 파일을 올려놔 투스텝이 아닌 원스텝의 촬영 어플을 구현할 생각이다. 처음에 들어갔을 때 재밋는 이미지를 2개 정도 준비해 선택해 촬영할 수 있도록 할 생각이다.

계획표

주 내용	5/1	5/2	5/3	5/4	6/1	6/2
안드로이드 카메라 및 동영상 동작 원리	O	O				
오픈시비 동작 원리			O			
최종 보고서 및 시연				O	O	O

성과 및 피드백

- ❑ OpenCV4Android 를 연동하고 이미지를 올려놓는 것이 생각했던 것보다 시간이 훨씬 많이 걸렸다. Opencv 예제들은 대부분 C++로 되어있고 안드로이드 예제가 많이 없었고 StackOverflow에만 조금 있어서 오래 걸렸다.
- ❑ 설치하는 방식도 버전별로 다 다르기 때문에 현재 나와 있는 3.0을 사용할 수 없었고 2.4.10을 사용해야 했다. 일반적인 Module Import로는 계속 로딩을 시킬 수가 없었다. 또한 Opencv는 라이브러리를 반복적으로 설치하기 원하지 않게 하려고 관련 어플이 있으면 바로 연동되어 앱스토어에서 OpenCV Help Manager라는 앱을 따로 설치하게 되는데 Shield의 버전과 맞지 않아 계속 에러가 났었다. 결국 코드 내에서 직접 Static Initialization을 하여 Help Manager없이 원래 프로젝트 라이브러리에서 모듈들을 가져오게끔 하였다. (https://www.youtube.com/watch?v=OTw_GIQNbD8) 이 유튜브를 따라해서 겨우 해냈다. 다른 포스팅은 하나도 되지 않았고 유일하게 이 유튜브가 도움이 되었다.
- ❑ 얼굴을 haarcascade.xml를 사용하여 트래킹하여 Core.Rect라는 함수로 해당 부위에 쉽게 네모 라인을 그릴 수 있었다. 그래서 이미지를 올려놓는 것도 어렵지 않게 바로 할 수 있을 줄 알았는데 이것도 시간이 너무 오래 걸렸다. 먼저 내가 이미지 프로세싱에 대해 기본지식이 없었던 이유가 가장 컸었고 관련 예제가 없었던 것도 컸었다. 특히, 이미지를 올려놓아도 이미지 Channel이 동영상은 CV8UC4였고 이미지 파일은 CV8UC3이었기 때문에 에러가 계속 났던 것인데 이런 것들을 처음 접하느라 처음부터 다시 공부했고 Channel을 변경해서 드디어 해냈다. 그래서 다음학기에는 미디어 처리나 멀티미디어 프로그래밍 수업을 꼭 듣고 싶다는 생각을 했다.
- ❑ 가장 아쉬운 부분은 완성도인데 원했던 기능들을 구현했더라도 UI 측면에서 잘 마무리를 하지 못했던 부분이다. 시간이 좀 더 있었으면 그리고 기말고사 전공시험 5개의 압박이 덜했으면 더 잘했었을 텐데 그 부분이 가장 아쉽다.

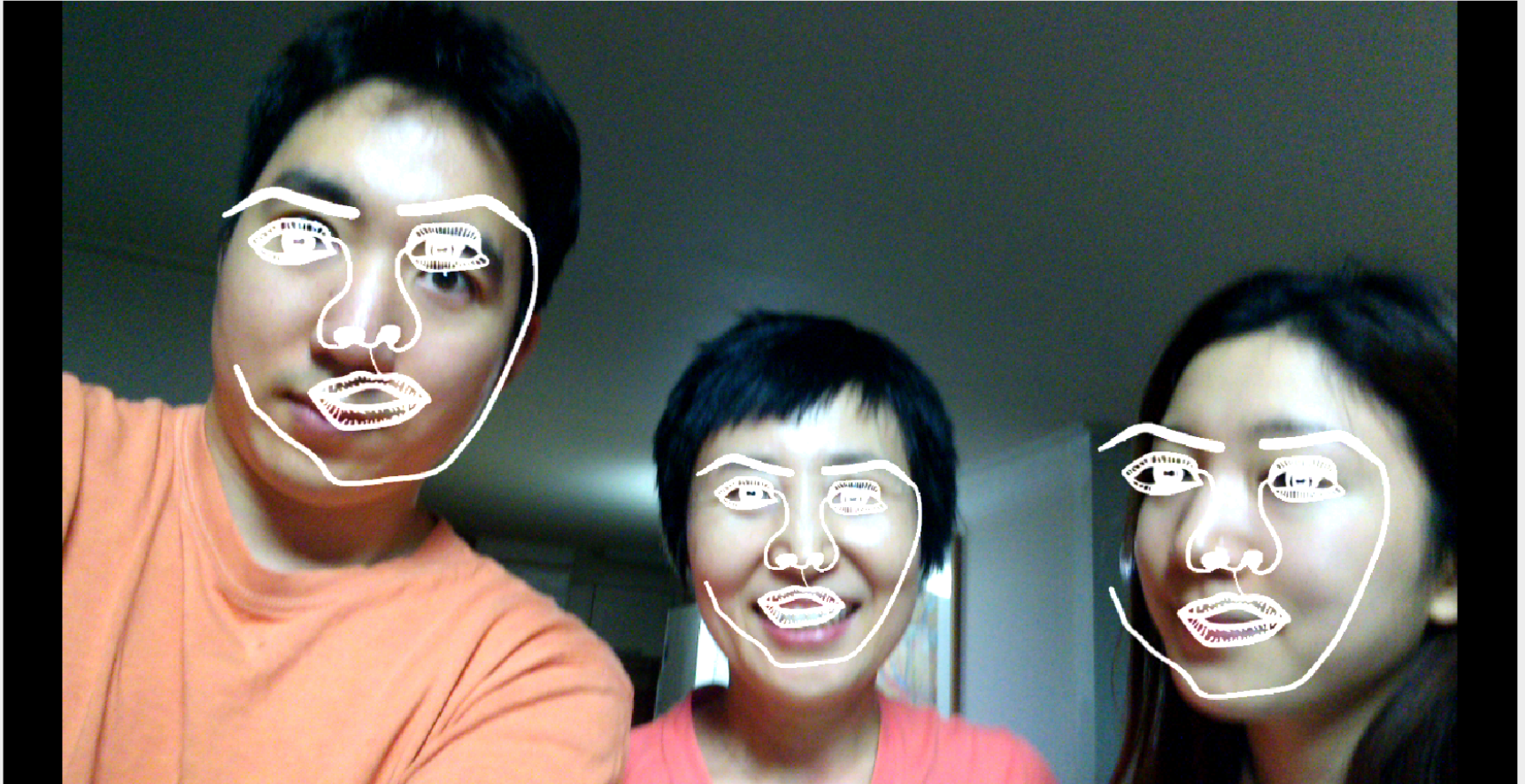
어플 설명

- 먼저 어플을 열면 카메라를 이용해 비디오 프리뷰가 나온다. 이 어플에 대한 설정 선택은 다 메뉴를 통해 올려놓았다. 메뉴들은 각각 앞쪽 카메라 사용, 뒤쪽 카메라 사용, 마스크 이미지 사용, 포커페이스 이미지 사용, 녹화 시작, 녹화 스톱.(front camera, backside camera, mask image, pokerface image, record start/record stop)이 있다. 녹화를 선택하게 되면 녹화가 시작되며 스톱을 하면 녹화가 끝난다. 해당 녹화 파일은 sd에 저장되게 했다. (이 부분은 MediaRecorder로 구현했는데 영상이 제대로 녹화되지 않았다. 프로세싱이 너무 느리기 때문에(framerate를 못따라간다고 한다) 녹화하기 위해서는 OpenCV를 Wrapper클래스로 가진 JavaCV라는 라이브러리를 따로 사용해야 했다. 하지만 시간이 없어서 이 라이브러리를 제대로 사용해 완성하지 못했다.)
- 이 어플의 단점은 빛이 없으면 얼굴을 트래킹할 수 없어 이미지가 띄어지지 않는다는 점이다. 오픈시비의 Haarcascade는 빛의 어둠과 밝음으로 사람 얼굴을 판별하기 때문이다. 또한 얼굴의 측면도 잘 판별할 수 없기 때문에 정면인 경우에 이미지 프로세싱이 가능하다.

화면 디자인

오후 11:04

Kre



참고 문헌

1. 안드로이드 프로그래밍 정복
2. Mastering OpenCV with Practical Computer Vision Projects
3. OpenCV Computer Vision with Python
4. <http://opencv.org/platforms/android.html>