

얼굴 표정 변화에 따른 감정 필터 제시

이동원, 201710877, 융합전자공학전공

최성우, 202011096, 융합전자공학전공

윤성민, 202011083, 융합전자공학전공

요약

본 팀이 선택한 주제는 얼굴의 표정 변화에 따라 각기 다른 감정 필터를 제시하는 것이다. 먼저 주어진 사람 이미지에서 얼굴 영역을 검출한 후, 외부 API 프로그램을 활용해 얼굴 표정에서 보이는 가장 적합한 감정을 판단한다. 이후 감정 판단이 완료된 이미지 속 사람의 감정을 강조하는 필터 효과를 적용해주는 것까지 프로젝트를 구성하였다.

해당 DSP 시스템은 기본적으로 MATLAB R2023b 버전을 활용하며, 정밀한 표정 감지를 위한 API 프로그램인 'AWS Rekognition'(이하 'AWS')을 활용하기 위해 python 코드를 활용해야 한다. 이 외에 필터 실행을 위한 추가 툴박스들의 설치도 요구된다.

소개 및 개발목적

사람의 표정을 감지하는 영상처리 시스템은 여러 분야에 활용될 여지가 있는 기술이다.

본 프로젝트의 개발 목적은 특정 분야에 도움이 되는 시스템을 만들기 위함은 아니다. 기본적으로는 불특정다수의 사용자에게 재미를 느끼게 하기 위한 엔터테인먼트 계열의 놀이용 필터 적용이 그 목적에 있다고 볼 수 있다. 해당 아이디어는 예전부터 AWS가 넷상에서 밈으로 소비되던 부분에서 얻었고, 이를 'SNOW' 같은 실시간 영상처리 필터에 표정 감지를 통한 자동 적용이 가능하다면 더 재미있지 않을까? 하는 호기심에서 시작되었다.

추가적으로 활용될 여지를 고려해보면, 아직 감정의 표현이 미숙한 유아들에게는 도움이 될 것 같다고 생각한다. 유아들에게 자신의 감정을 필터를 통한 시각적 효과로 강렬하게 느끼게 해줌으로서 자신의 표정을 능숙하게 컨트롤하는 교육에 쓰일 수도 있다고 본다.

본문 1. 외부 API 프로그램을 활용한 얼굴 표정 감지

프로젝트 코드 실행을 위한 사전 설정

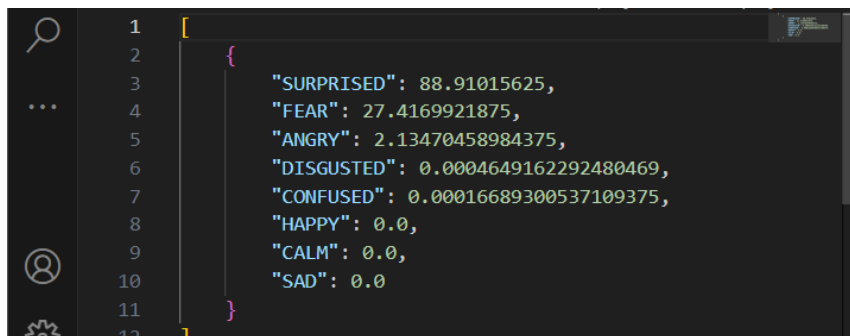
1. 설정 상세는 프로젝트 github 링크를 통해 확인할 수 있습니다.
 2. 해당 프로젝트는 MATLAB 애드온 Computer Vision Toolbox, Image Processing Toolbox를 필요로 합니다.
 3. MATLAB R2024a 버전 기준으로 MATLAB 환경 내에서 Python 3.9, 3.10, 3.11의 실행을 지원합니다.¹
 4. MATLAB에서 pyenv 함수를 활용하여 python 경로 설정을 진행합니다.
- 현재 작성된 코드는 python에 boto3 라이브러리를 필요로 합니다.

¹ https://kr.mathworks.com/support/requirements/python-compatibility.html?s_tid=srchtitle_site_search_1-python%20compatibility

5. AWS 서비스 이용을 위해 필요한 ACCESS_KEY 와 SECRET_ACCESS_KEY가 지급된 값이 맞는지 확인합니다.

Amazon Rekognition은 AWS에서 제공하는 클라우드 기반의 영상 분석 서비스이다. 딥 러닝 기반으로 구동되는 해당 서비스는, C++, Python, Java 등 10가지 이상의 프로그래밍 언어를 통한 개발 튜토리얼을 제공하여 다양한 환경에서 활용이 가능하다. Rekognition은 간단한 적용 방법과 사전 구축된 자체 API를 통해 별도의 기계학습 과정을 생략하고 이미지의 정보 분석을 빠르게 진행하는 데 도움을 준다.²

감정 분석 시스템의 전체적인 진행 과정은 다음과 같다. MATLAB 환경에서 MATLAB 코드 'image_filtering.m' 실행하면, sample 폴더에 저장된 파일명(image_path)의 이미지를 가져온다. 이후 Python으로 이루어진 코드 'detect_faces.py'를 호출하여 AWS가 판단한 감정들의 인식값 아래 <그림 1>과 같은 json 파일로 저장한다.



<그림 1> 'SURPRISED'로 판단된 이미지의 감정 percent 판단 json파일

이때 가장 큰 Confidence 값을 maxEmotion으로 설정한다. 위의 <그림 1> 같은 경우는 'SURPRISED' 수치가 89%로 가장 높은 수치이기에 'SURPRISED=maxEmotion'으로 판단하는 것이다. 다음으로 maxEmotion에 대응하는 준비된 filter 함수를 불러와 적용시킨다. 그러면 최종적으로 원본 이미지, filter 적용 이미지, maxEmotion 항목과 Confidence점수 figure에 출력된다. 각 filter 함수에 대한 구조는 아래의 본문 2에서 확인할 수 있다.

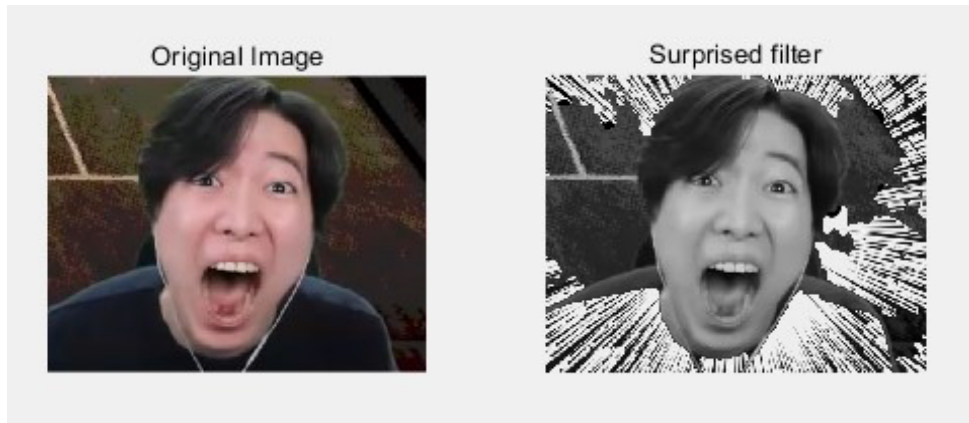
본문 2. 감정 필터 시뮬레이션 시행 및 결과

감정 필터 프로그램은 모두 '행복', '슬픔', '분노', '놀라움', '역겨움'의 5가지 감정만을 검출하여 이에 대응되는 필터 함수를 가져와 적용하도록 앞의 표정 검출 시스템에 설정되어 있다. 본래 API에서 지원하는 감정의 종류는 총 8가지이나, 본 프로젝트에서는 위의 5가지 감정의 json 파일 값만을 받아 필터에 적용하기로 기획하였다.

² <https://docs.aws.amazon.com/rekognition/latest/dg/what-is.html>

<https://docs.aws.amazon.com/rekognition/latest/dg/face-feature-differences.html>

감정 필터는 크게 3종류의 코드 구조를 기반으로 이루어져 있다. 놀라움 필터, 슬픔 필터, 기쁨 필터가 동일한 코드 구조를 기반으로 이루어져 있고, 각 감정 필터는 이미지와 최종 이미지 출력 부분에서의 차이점만이 존재한다. 그 외에 분노 필터 전용 코드와 역겨움 필터 전용 코드로 이루어져 있다.



<그림 2> 놀라움 필터(surprised_filter)

<그림 2>에서 볼 수 있듯이, 놀라움의 표정으로 판단된 위의 샘플 이미지를 불러와 실행한 결과 Canny edge 검출을 통해 정상적으로 얼굴을 제외한 배경이 크로마키 처리된 후 배경 이미지 'surprise_background.jpg'가 깔린 것을 볼 수 있다.

이를 실행하는 MATLAB 프로그램은 이미지를 최종 출력하는 메인 코드 이외에 4가지의 함수로 이루어져 있다. 각각 원본 전경 이미지의 edge를 검출하여 크로마키가 적용될 전경 이미지의 배경 부분을 제거하는 함수인 'removeBackground(image)', 배경 이미지를 읽어와 크로마키 처리된 전경 이미지와 합성시킨 결과를 반환하는 크로마키 합성 함수인 'chromakey (faceImage, backing)', 크로마키 마스크를 씌우는 생성 함수인 'createChromaKeyMask(image, chromaKeyColor)', 크로마키 마스크를 조정하여 전경 이미지와 배경이미지를 합성하는 기능을 수행하는 이미지 합성 함수 'blendImages(foreground, background, mask)'를 활용하는 프로그램을 활용하였다. 이후 최종 이미지 출력 전 전체 이미지의 RGB 값을 조정하는 코드를 살짝 넣어 주었다. 놀라움 필터와 같은 코드 구조가 적용된 감정 필터는 '슬픔 필터'와 '행복 필터'가 있다.

주의사항이 있다면, Object Detector류의 객체 탐지 프로그램을 따로 사용하지 않았기에 얼굴 부분의 edge가 제대로 감지되지 않아 얼굴 영역 일부에 배경이 침범하는 오류가 발생하는 경우도 있다. 이를 커버하기 위해 최종 출력되는 전체 이미지에 grayscale 처리를 하여 적용된 배경 이미지와 검출 대상의 색채가 어우러짐과 동시에 마치 만화장면 같이 보이도록 연출하였다. 이는 같은 함수를 공유하는 '슬픔 필터(sad_filter)'에서도 배경의 색채인 파란색과 맞춰 최종 이미지에 RGB값 중 파란색을 증폭시켜 배경과 조화를 이룸과 동시에 피부 톤까지 푸르게 변형시킴으로써 더욱 슬픈 분위기를 주도도록 연출하였다.



그림 3. 분노 필터(angry_filter)

<그림 3>에서 볼 수 있듯이, 분노 필터에 적용되는 코드의 알고리즘은 앞의 놀라움 필터와 조금 다르다. 일단 Canny 엣지 검출을 활용한 얼굴검출 알고리즘과 달리 분노 필터에서는 Computer Vision System Toolbox에서 지원하는 ‘vision.CascadeObjectDetector’를 사용하여 이미지나 비디오 속 얼굴 감지를 수행한다. 해당 감지기는 AdaBoost 알고리즘을 사용하는 비올라-존스 객체 감지 프레임워크를 기반으로 한다. , Canny 엣지 검출도 같이 활용하여 얼굴 윤곽선이 검출한 후, 모폴로지 연산을 통한 노이즈 제거한다. 최종적으로 검출된 얼굴 윤곽선의 내부 영역의 RGB 채널 중 빨간색 수치만을 증폭시키면 얼굴 피부 톤이 붉어져 마치 열이 오른 것 같은 만화적 효과를 보여준다.

추가적으로 크로마키를 통한 이미지 합성을 사용하지 않는 대신 사거리 마크(angry mark)를 배경 투명화 처리 후 얼굴 상단에 위치하도록 설정하여 또 다른 만화적 효과를 연출하였다.



그림 4. 역겨움 필터(disgusted_filter)

<그림 4>에서 볼 수 있듯이, 해당 필터는 필터가 적용될 대상 이미지 외의 배경 이미지나 추가 이미지를 필요로 하지 않는다. 단순히 색 반전 효과와 이미지 전체에 웨이브 효과를 적용하는 간단한 코드만을 필요로 하기 때문이다. 이미지 내에서 특정 대상(얼굴)을 감지할 필요도 없기 때문에 추가 함수 또한 필수적이지 않다. 단지 이미지의 색상 반전 명령어인 `imcomplement(img)`와 웨이브 효과에서의 진폭과 주파수를 조절하는 파라미터 값만 간단히 설정해줌에 따라 혼란스럽거나 역겨움의 감정을 과장되게 표현할 수 있다.

결론

본 프로젝트는 얼굴에서 드러나는 개개인의 표정을 판단함으로서 그 사람의 감정을 얼마나 제대로 파악하는지, 파악한 감정을 적절한 형태로 표현했는지에 의의를 둔다. 물론 아직은 JPG, PNG 등 정적인 이미지 파일 형태만이 적용 대상이 되어 여러 분야에 활용되기엔 한계점이 분명하다.

그러나 본 프로젝트의 알고리즘을 활용하여 비디오나 실시간 영상처리 중에 자동으로 표정을 감지한 후, 감정의 필터링 처리가 가능하도록 발전시킨다면, 서론에서 말한 유아용 표정 교육뿐만 아니라 의료 계열에서의 환자 상태 파악, 또는 범죄/수사 계열에서의 용의자 심리 파악 등 다양한 분야에 있어서 큰 도움이 될 여지가 있다고 판단하였다.

참고 및 인용 사이트

- [1] <https://aws.amazon.com/ko/rekognition/>
- [2] https://kr.mathworks.com/support/requirements/python-compatibility.html?s_tid=srchtitle_site_search_1_python%20compatibility
- [3] <https://docs.aws.amazon.com/rekognition/latest/dg/what-is.html>
<https://docs.aws.amazon.com/rekognition/latest/dg/face-feature-differences.html>
- [5] https://github.com/TyPe6000/imageprocessing_project_24-1