

## 얼굴 표정 인식 프로그램

이 프로그램은 제가 2014년 교환 학생 당시, Computer Vision & 3D Reconstruction 수업의 최종 프로젝트로서 제출했던 프로그램입니다. 당시, 표정을 제대로 인식하는데 성공하여, 교수님과 다른 학생들에게 큰 호응을 얻었던 프로젝트입니다. Python을 기반으로 작성되었고, OpenCV 라이브러리의 Optical Flow의 분석의 기능을 사용하여, 얼굴의 감정 변화를 잡아내는 프로그램입니다.

### 주요 기능

- 컴퓨터의 웹 캠을 통하여 실시간으로 사용자가 어떤 표정을 짓고 있는지 파악한 후, 상단에 Happy 혹은 Sad라는 메시지를 출력하는 프로그램입니다.
- 배경이 복잡하거나 조명이 어두운 공간에서는 얼굴의 인식이 힘들므로 표정 역시 판독하기 힘들기에, 단순한 배경에 밝은 조명 아래서만 제대로된 판독이 가능합니다.

### 구현 방법 및 과정

1. 우선 OpenCV 라이브러리의 기능을 이용하여, 화면에 있는 사용자의 얼굴 영역을 인식합니다.
2. 그 중에 눈과 입을 인식합니다. 이 역시, 라이브러리 튜토리얼로 공개되어 있는 xml코드와 OpenCV에 이미 존재하는 라이브러리 기능을 활용합니다.
3. 입으로 인식된 영역에서 광학 흐름 분석(Optical Flow)을 통해 어떤 모션들이 발생하는지 관찰합니다.
4. 왼쪽 윗방향( $\nwarrow$ )과 오른쪽 윗방향( $\nearrow$ )으로의 모션들의 갯수를 기억합니다.
5. 또한 왼쪽 아랫방향( $\swarrow$ )과 오른쪽 아랫방향( $\searrow$ )으로의 모션들의 갯수를 기억합니다.
6. 4번과 5번을 비교하여, 4번이 5번에 비해 훨씬 클 경우( $\nwarrow$ 의 개수 >  $\swarrow$ 의 개수) Happy, 반대일 경우 Sad로 판독합니다.

(Happy와 Sad가 둘 다 되지 않던 상황은 Neutral이라고 판독했었지만, 최종 버전에선 삭제했습니다.)

### 팀 내부에서 본인의 역할

- 저는 처음 프로젝트를 시작할 당시, 이 아이디어를 내고 팀원을 모집하였습니다.
- 프로젝트가 시작된 후에는, 핵심 알고리즘의 구현을 맡아 프로젝트를 작업했습니다.
- 제가 작성한 알고리즘은 OpenCV 라이브러리를 통하여, Optical Flow를 화면에 출력해 내는 것과 그 결과물들을 바탕으로 Happy와 Sad를 판독하는 알고리즘입니다.

### 한계 및 발전 방향

- 앞서 설명했듯이, 제한적인 상황에서만 구현되는 것이 이 프로젝트의 한계점이라 볼 수 있습니다.
- 처리 속도가 예상했던 것 보다 오래 걸려, 빠른 시간 내에 표정이 바뀐다면 정확도가 떨어집니다. 따라서 앞으로 프로그램 구동 속도를 개선 시킬 필요가 있습니다.
- 처리 속도 상의 문제로, 애초에 기획했던 눈 영역의 분석이나 다른 얼굴 영역(예를 들면 볼이나 광대)의 분석을 하지 못하였습니다. 입꼬리가 올라간 상황이라도, 슬픈 표정 일수도 있기에 종합적인 분석이 필요합니다.