

이게 나...? : 유명인을 닮은 내 얼굴 만들기 프로그램

팀 명 룡딩동

팀 원 김경연

지도교수 신현준

멘 토 정태홍(스노우)

개발 동기 및 목적

AI 프로필 열풍!

지난 해, 여러 카메라 플랫폼에서 내놓은 AI 아바타, 프로필, 이어북 등이 SNS 상에서 크게 유행했다. 이러한 트렌드에 맞추어 '이미지 프로그래밍을 통한 고객 맞춤형 이미지 생성' 을 주제로 삼았다.

*AI 프로필이란?

사용자가 사진을 입력하면 다양한 스타일에서 자신의 얼굴이 적용된 이미지를 생성해주는 시스템



AI 프로필
30개 테마로 만드는 당신을 위한 특별한 프로필
시를 통해 "나"를 표현해 보세요.

같은 테마끼리 묘하게 비슷한 인상

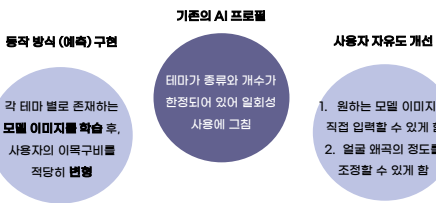


출처 유명자 인스타그램

출처 스노우 실행장면, 박영수

시프로필을 직접 사용하고 관찰한 결과

사용자의 이목구비가 어느 정도 반영되었으나, **얼굴형이나 눈 간격 등 디테일에 의해 나타나는 고유한 인상은 사용자가 아닌 테마에 따라 달라짐**을 확인할 수 있었다. 이에 따라 시스템 동작 방식을 예측, 사용자 자유도를 개선한 프로그램을 목표로 하였다.



개발 내용

01 Mediapipe로 두 이미지의 face landmarks 추출

Source Model Mediapipe의 sample code 에서 pipeline graph 수정 sample 수정

(Webcam) Video Image Video Normalized LandmarkList

02 Simple Linear Regression

두 이미지의 얼굴 크기와 위치를 맞추기 위하여 2x3 coefficient를 구한 뒤,

$$Ax = B$$

478x2 Source 478x3 Model

원본 이미지에 coefficient 적용하여 SLR, 받아온 map으로 원본 이미지 remap

03 Radial Basis Function

두 이미지의 랜드마크 분포를 근사하기 위하여 dx=model.x-src.x dy=model.y-src.y 과 kernel function으로 weight를 구한 뒤, ThinPlateSpline, Gaussian 두 방법

원본 이미지에 weight 적용하여 RBF, 받아온 map으로 원본 이미지 remap

04 Opencv.highGUI

Trackbar 추가하여 왜곡의 정도 조정
슬라이더의 값은 Src 좌표와 RBF map의 비율로, 0에 가까울수록 src 값, 100에 가까울수록 RBF 결과값

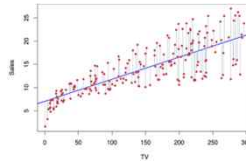
주요 기술

사용 도구



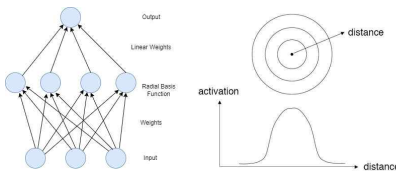
- Mediapipe (Google의 AI Framework)를 활용한 C++ 개발을 위해 빌드 도구로 Bazel을 사용하였다.
- 이미지처리와 행렬 연산을 위한 openCV (라이브러리)를 사용하였다.

Single Linear Regression



두 데이터 사이 최소 자승법을 적용하여 가장 적합한 일차 함수 식을 추정하는 것으로, translation과 scaling을 위한 coefficient를 구하는 데 사용하였다.

Radial Basis Function



Input space 데이터 간의 Euclidean 거리에 의존하여 계산되는 함수로, RBF의 합은 데이터 분포를 근사하는 데 유용하게 사용된다.

결과 및 분석

Source	Model	Result 1 (Gaussian)

- 이미지 두 장을 입력하면 자동으로 계산하여 src의 이목구비를 변형한 이미지를 출력한다.
- command line argument로 이미지 경로를 입력할 수 있으며, 슬라이더를 사용하여 직접 왜곡의 정도를 조정할 수 있다.
- RBF kernel로 Gaussian function을 사용했을 때 자연스러운 결과를 얻을 수 있었다.

Result 2 (ThinPlateSpline)



- RBF kernel로 Thin Plate Spline을 사용했을 때는 해당 kernel의 특성상 배경까지 왜곡되는 현상이 나타났다.

- ✓ 이를 보완하기 위해 배경 합성 기능을 추가할 예정이다. 배경 이미지로는 시프로필처럼 샘플로 제공하거나, 기존에 있던 চে형 미반영 문제를 개선하기 위하여 사용자 입력 이미지를 사용하려고 한다.
- ✓ 더 빠르고 자연스러운 변형을 위하여 Delaunay Triangulation을 구현할 예정이다.

오픈소스 URL

https://github.com/kky107102/mediapipe/tree/kky_face_deformation

