|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **프로젝트 계획서** | | | | | | | |
| 프로젝트  분야/주제 | Neural Style Transfer | | | | | 총 투입입원 | 1명 |
| 프로젝트 명 | 국문 | | 스트링 | | | | |
| 영문 | | Style Transfer for Recollection(STR) | | | | |
| 수행기간 | 2024. 03. 03. ~ 2024. 06. 18. | | | | | | |
| 참여자 | 소속 | | 참여자성명 | 학년 | 학번 | 이메일 | |
| 소프트웨어학과 | | 오세강 | 4 | 32212610 | chrisryn@naver.com | |
|  | |  |  |  |  | |
|  | |  |  |  |  | |
|  | | | | | | | |
| **프로젝트 개요** | |  | | | | | |
| 오늘날 우리는 디지털 기술을 이용하여 촬영한 사진에 다양한 필터를 통해 전혀 다른 분위기의 사진을 만들어낸다. 필터는 사진의 밝기, 명도, 채도, 선명도, 색감을 바꾸는 것뿐만 아니라, 사진을 마치 그림처럼 바꾸기도 한다. 그러나 이 필터들은 기존에 정해져 있는 방식대로만 작동하고, 내가 원하는 그림으로 바꿔주지는 않는다.  이와 더불어, 최근 인공지능을 공부하며 읽게 된 논문 <A Neural Algorithm of Artistic Style; Leon A. Gatys>을 통해 사진을 특정 화가의 화풍으로 바꿔 그리는 Neural Style Transfer에 관심을 가지게 되었다.  사용자의 사진을 유명 화가의 화풍으로 변경하는 ‘Enpainter’와 구글에서 출시한 ‘아트 트랜스퍼’ 등 Neural Style Transfer와 관련된 기존의 서비스들은 사진 전체를 특정 화풍으로 변경하는 서비스들이다. 이 프로젝트는 기존 서비스들과 달리 사진 속 인물들을 제외하고 효과를 적용하여 배경만 특정 화풍으로 변경해주는 새로운 서비스를 만들어보고자 한다.  Style Transfer for Recollection(STR)은 인터넷이 연결된 환경이라면 언제 어디서든 사용이 가능한 웹 기반 서비스로, 사진을 사용자가 원하는 특정 화풍으로 바꿔주는 서비스이다. 사진 속 인물은 segmentation 기법을 사용하여 구분하고, 인물을 제외한 배경을 변경한다. 기본적으로 제공하는 몇 가지 화풍 외에도 사용자는 자신이 사진에 적용하기 원하는 화풍을 직접 골라 적용시킬 수 있다. 사용자는 새로운 사진을 통해 자신이 원하는 작품 속에 인물이 들어간 듯한 느낌을 받을 수 있다. | | | | | | | |
| **기능** | |  | | | | | |
| **Front-end**   |  |  | | --- | --- | | 요구사항 명 | 메인 메뉴 | | 상세 요구사항  내용 | * 메인 화면에 프로그램에 대한 설명을 보여준다. * 화풍선택창으로 넘어갈 수 있는 버튼을 제공한다. | | 중요도 | * 1(매우 중요) |  |  |  | | --- | --- | | 요구사항 명 | 화풍선택창 | | 상세 요구사항  내용 | * 사용자가 바꾸고자 하는 원본사진을 업로드 할 수 있는 공간을 제공한다. * 업로드한 원본사진이 마음에 들지 않을 경우 다시 업로드할 수 있는 방법을 제공한다. * 제공하는 화풍 선택을 할 수 있는 스크롤 선택창을 제공한다. * 사용자가 바꾸고자 하는 화풍의 사진을 업로드 할 수 있는 공간을 제공한다. * 인물을 제외하고 화풍을 변경할지 인물을 포함하여 화풍을 변경할지 선택할 수 있는 체크박스를 제공한다. * 왼쪽에 업로드한 사진을 보여준다 * 오른쪽에 선택, 또는 업로드한 화풍을 보여준다. * 사용자가 변환 버튼을 누르면 서버로 사진을 전송한다. | | 중요도 | * 1(매우 중요) |  |  |  | | --- | --- | | 요구사항 명 | 결과 화면 | | 상세 요구사항  내용 | * 화풍이 변화된 결과물을 중앙에 반환한다. * 결과물을 다운로드 할 수 있는 버튼을 제공한다. * 처음으로 돌아갈 수 있는 버튼을 제공한다. | | 중요도 | * 1(매우 중요) |   **Back-end**   |  |  | | --- | --- | | 요구사항 명 | 업로드 사진 전송 | | 상세 요구사항  내용 | * 사용자가 업로드한 원본사진을 전송받는다. * AI서버로 사용자가 업로드한 원본사진을 전송한다. * 사용자가 인물의 화풍변경을 선택했는지 아닌지에 대한 정보를 AI서버로 전송한다. * 사용자가 화풍을 선택했는지, 화풍사진을 업로드 했는지 확인한다. * 사용자가 화풍을 선택했다면 해당정보를 AI서버로 전송한다. * 사용자가 화풍사진을 업로드 했다면 화풍사진을 AI서버로 전송한다. * AI서버로부터 전송된 화풍이 변경된 사진을 전송받는다. * 사용자 서버로 화풍이 변경된 사진을 전송한다. | | 중요도 | * 1(매우 중요) |   **AI 서버**   |  |  | | --- | --- | | 요구사항 명 | 화풍 변경 | | 상세 요구사항  내용 | * AI 서버로부터 사용자가 업로드한 원본사진을 전송받는다. * 사용자 서버가 화풍 정보를 전송했는지 화풍사진을 전송했는지 확인한다. * 사용자 서버가 화풍사진을 전송했다면 화풍사진을 학습하여 필터를 만든다. * 사용자 서버가 보낸 인물의 화풍변경 정보를 확인한다. * 인물의 화풍변경이 선택되었다면 화풍 필터를 통해 원본사진 전체의 화풍을 변경한다. * 인물의 화풍변경이 선택되지 않았다면 인물의 세그멘테이션을 진행하고 화풍 필터를 통해 원본사진 배경의 화풍을 변경한다. 이후 세그멘테이션된 인물과 화풍변경된 배경을 합성한다. * 사용자 서버로 화풍이 변경된 사진을 전송한다. | | 중요도 | * 1(매우 중요) | | | | | | | | |
| **추진 전략** | |  | | | | | |
| \* 기능 구현 배분   * Front-End(홈페이지 구축)   HTML, CSS, Javascript 웹 페이지 구축  웹페이지 배너 구성 및 디자인과 이미지 업로드 및 다운로드 등 서버와 전송 관련한 파트를 나누어 진행   * Back-End(사용자 서버 구축)   NAS 서버에 Docker 구축, Nginx & WSGI & Flask 구축을 통해 외부망과 연결   * AI 모델 구축   Python을 통한 AI 모델 구축  Neural Style Transfer를 통한 화풍변환 AI 모델 학습 및 변형  ViT-Adapter Segmenter의 학습 및 변형을 통해 인물 세그멘테이션 진행  결과적으로 사용할 Segmentation & Transfer 융합  \* 소스 코드 버전 관리 방안   * Git과 Github 활용 * branch를 활용하여 버전 관리: Github Flow 전략 * Main branch: 항상 배포가 가능한 branch * Feature branch: 새로운 기능 개발, 오류 수정 등의 경우 main에서 생성하는 새로운 branch   \* 추진 일정   |  |  |  |  |  |  |  |  |  |  |  |  |  |  | | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | | 주요내용 | 추진 일정 | | | | | | | | | | | | | | 3/26 | 4/2 | 4/9 | 4/16 | 4/23 | 4/30 | 5/7 | 5/14 | 5/21 | 5/28 | 6/4 | 6/11 | 6/18 | | SRS작성 |  |  |  |  |  |  |  |  |  |  |  |  |  | | 프로젝트 제안서 작성 |  |  |  |  |  |  |  |  |  |  |  |  |  | | 고수준 설계 |  |  |  |  |  |  |  |  |  |  |  |  |  | | 프로젝트 구현: Neural Style Transfer |  |  |  |  |  |  |  |  |  |  |  |  |  | | 프로젝트 구현: ViT-Adapter Segmenter |  |  |  |  |  |  |  |  |  |  |  |  |  | | 프로젝트 구현: Segmentation & Transfer 융합 |  |  |  |  |  |  |  |  |  |  |  |  |  | | 프로젝트 구현: NAS Docker 구축 |  |  |  |  |  |  |  |  |  |  |  |  |  | | 프로젝트 구현: Nginx & WSGI & Flask 구축 |  |  |  |  |  |  |  |  |  |  |  |  |  | | 프로젝트 구현: 웹페이지 배너 구성 |  |  |  |  |  |  |  |  |  |  |  |  |  | | 프로젝트 구현: 이미지 업로드 및 다운로드 구성 |  |  |  |  |  |  |  |  |  |  |  |  |  | | 테스트 |  |  |  |  |  |  |  |  |  |  |  |  |  | | 프로젝트 최종 구현 완료 및  보고서 작성 |  |  |  |  |  |  |  |  |  |  |  |  |  | | | | | | | | |
| **예상 최종결과물** | |  | | | | | |
| \* 연구개발 결과물의 특성   * 인터넷 접속이 가능한 디바이스, PC 또는 모바일 단말기에서 사용가능 * 웹 브라우저를 통한 실행, Chrome 최적화   \* 결과물 예시 | | | | | | | |
| **참고문헌** | |  | | | | | |
| 1. A Neural Algorithm of Artistic Style (Leon A. Gatys, Alexander S. Ecker, Matthias Bethge, 2015) 2. Deep Residual Learning for Image Recognition (Kaiming He, Xiangyu Zhang, Shaoqing Ren, Jian Sun, Microsoft Research, 2015) 3. An Image is Worth 16x16 Words: Transformers for Image Recognition at Scale (Alexey Dosovitskiy, Lucas Beyer, Alexander Kolesnikov, Dirk Weissenborn, Xiaohua Zhai, Thomas Unterthiner, Mostafa Dehghani, Matthias Minderer, Georg Heigold, Sylvain Gelly, Jakob Uszkoreit, Neil Houlsby, 2021) 4. Vision Transformer Adapter for Dense Predictions (Zhe Chen, Yuchen Duan, Wenhai Wang, Junjun He, Tong Lu, Jifeng Dai, Yu Qiao, 2023) | | | | | | | |