

생성형 모델을 활용한 AI 헤어스타일러



개요

생성형 모델을 활용한 AI 헤어스타일러는 현대 사회에서 외모가 자신감과 사회적 인상에 미치는 영향을 고려하여, 사용자가 다양한 헤어스타일을 가상으로 시험해 볼 수 있는 기술 개발을 목표로 한다. 이 기술은 다양한 각도와 조명하에서도 사용자의 헤어 텍스처를 자연스럽게 보존하며 변환할 수 있어야 하며, 실시간으로 헤어스타일을 변경해보는 경험을 제공한다. 이를 위해 생성형 모델인 GAN을 활용, 고급 이미지 변환 기술을 적용하여, 헤어스타일의 세밀한 부분까지 효과적으로 재현하고자 한다. 프로젝트는 헤어스타일 데이터베이스 구축, 모델 학습 및 최적화, 그리고 웹 서비스 개발을 통해 사용자 친화적인 인터페이스를 제공할 예정이다. 본 프로젝트는 사용자에게 더욱 풍부하고 개인화된 미용 경험을 제공을 통해 외모 관리를 위한 효과적인 서비스를 제공하는 것이 최종 목표이다.

목차

1. 연구 배경

- a) 과제 배경
- b) 과제 목표

2. 문제 상황 및 요구 조건 분석

- a) 문제 상황 분석
- b) 요구 조건 분석
- c) 제약 사항 및 대책

3. 시스템 구조

- a) 모델 학습 파이프라인
- b) 모델 서빙 파이프라인

4. 개발 환경 및 사용 기술

- a) 개발 환경 및 언어
- b) 사용 기술

5. 일정 및 역할 분담

- a) 개발 일정
- b) 역할 분담

6. 참고 문헌

1. 연구 배경

a) 과제 배경

현대 사회에서 개인의 외모는 자신감과 사회적 인상을 형성하는 데 중요한 역할을 한다. 이 중 헤어스타일은 개인의 외모 변화에 크게 기여할 수 있는 요소로, 적합한 헤어스타일을 선택하는 것은 개인의 이미지를 긍정적으로 변화시킬 수 있는 가장 직접적인 방법 중 하나이다. 그러나 실제로 헤어스타일을 변경하기 전에 이를 시뮬레이션할 수 있는 효과적인 방법이 부족하여, 많은 사람이 미용실에서의 실패를 경험하곤 한다.

기존에 존재하는 헤어스타일을 적용해보는 서비스는 적은 선택지로 분류된 헤어스타일을 단순히 얼굴 이미지 위에 씌어보는 형식으로 제공하고 있다. 이는 실용적이지 못하고, 사용자가 스타일을 미리 적용해본다는 느낌을 받을 수 없고, 원하는 스타일링 결과를 얻지 못하는 경우가 허다하다. 또한, 기존의 서비스는 라이브 스트리밍 형태의 결과가 아니라, 단순히 이미지 형태의 결과만을 제공하기 때문에 실용성 및 효용성이 떨어진다. 이러한 문제를 해결하기 위하여 가상 환경에서 다양한 헤어스타일을 시험해 볼 수 있는 생성형 AI 모델을 활용한 실시간으로 적용되는 서비스 개발은 사용자의 요구사항을 충족할 수 있다.

이러한 서비스를 통해 사용자는 다양한 헤어스타일을 쉽게 시험해 보고, 자신의 얼굴형, 피부색, 그리고 선호에 가장 잘 맞는 스타일을 선택할 수 있게 된다. 이는 사용자에게 더 큰 만족감을 제공하며, 새로운 헤어스타일 시도 하는 것에 따른 불안감을 줄여줄 수 있다.

b) 과제 목표

본 프로젝트의 주된 목표는 자세, 조명, 환경에 적응하여 스트리밍 형태의 실시간 헤어스타일 변환 모델을 개발하는 것이다. 이 모델은 사용자가 가상으로 실시간으로 다양한 헤어스타일을 시험해 볼 수 있도록 하는 것을 목표로 하며, 사용자의 선택 경험을 강화하고, 실제 헤어스타일 변경 전에 충분한 예측을 제공하여 만족도를 높일 수 있도록 한다.

모델은 헤어스타일의 질감과 세밀한 부분까지도 자연스럽게 재현할 수 있어야 하며, 다양한 인종, 얼굴 형태 등의 개별 사용자 특성을 반영할 수 있는 기능도 갖추어야 한다. 이를 위해 딥러닝 기반 생성형 모델과 같은 최신 인공지능 기술을 활용하여 가장 현실적인 결과를 생성할 수 있는 모델 개발을 주요 목표로 설정한다. 또한, 이 모델을 활용하여 사용자가 쉽게 접근하고 사용할 수 있는 웹 기반 인터페이스를 개발하여 서비스를 제공할 예정이다.

2. 문제 상황 및 요구 조건 분석

a) 문제 상황 분석

현재 제공되는 다수의 헤어스타일 변환 앱과 서비스는 주로 정면 사진에 최적화되어 있어, 사용자가 다양한 각도로 촬영한 사진에 대해서는 정확도가 크게 떨어진다. 이러한 앱들은 헤어의 텍스처와 세부적인 디테일을 자연스럽게 보존하지 못하는 문제를 지니고 있다. 또한, 실시간 애플리케이션에서 사용자의 다양한 포즈와 표정 변화에 따른 헤어스타일 변환의 일관성을 유지하지 못하는 경우가 많아, 사용자 경험을 저하하는 주요 원인이 되고 있다.

b) 요구 조건 분석

1. 다양한 각도 지원:

사용자가 다양한 각도에서 촬영한 사진이나 비디오에서도 헤어스타일을 정확하게 적용할 수 있는 기술이 필요하다.

2. 텍스처 보존:

변환된 헤어스타일이 원본의 텍스처와 색상을 자연스럽게 재현하여 현실감 있는 결과를 제공해야 한다.

3. 사용자 친화적 인터페이스:

사용자가 쉽고 직관적으로 조작할 수 있으며, 결과를 신속하게 확인할 수 있는 인터페이스가 중요하다.

4. 실시간 처리 능력:

실시간 비디오 스트리밍에서 헤어스타일 변경을 지원하기 위해 빠른 처리 속도를 가진 고성능 이미지 처리 기술이 요구된다.

5. 다양한 헤어 스타일과 특성 지원:

다양한 피부색, 얼굴형 등과 머리카락 유형에 맞는 헤어스타일을 제공할 수 있는 광범위한 데이터 수집과 분석이 필요하다.

c) 제약 사항 및 대책

1. 계산 리소스:

생성형 AI를 활용한 고성능 헤어스타일 변환 모델은 높은 계산 리소스를 요구한다. 이를 위해

생성형 AI 모델에서 Diffusion 기반 모델들은 모두 여러 어간을 거쳐서 이미지가 생성되기 때문에, GAN(Generative Adversarial Network)보다 이미지 생성과정에 걸리는 시간이 적게는 100배, 많게는 1000배 이상 걸릴 수 있다. 이러한 요구 조건 및 제약 사항을 고려했을 때, 실시간 스트리밍을 위해서는 최근 성능이 좋은 Diffusion 기반 생성모델을 사용하기보다는, 짧은 생성 시간에 더 큰 가중치를 두어서, GAN을 기반으로 한 생성모델을 사용한다.

2. 데이터 개인정보 보호:

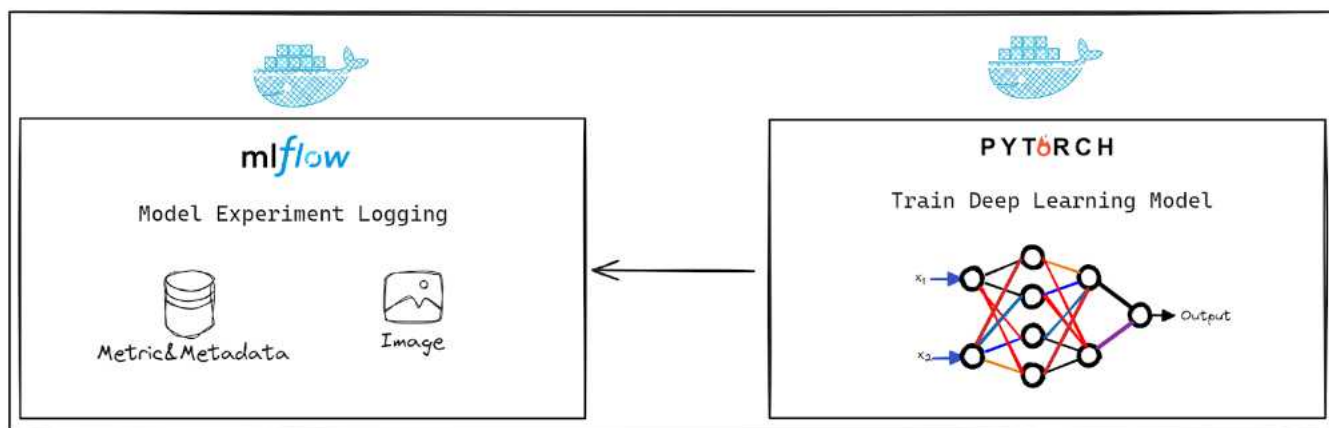
사용자의 사진이나 비디오 데이터를 처리할 때는 개인정보 보호법을 준수해야 한다. 이를 위해서 서비스 구현 과정에서 사용할 데이터는 주로 공공데이터(Open Data)를 사용하며, 데이터는 암호화 과정을 거친 후에 임시로만 저장되도록 한다.

3. 처리 성능 및 비용:

실시간 이미지 처리는 많은 컴퓨팅 자원을 소모하므로, 비용 효율적으로 처리할 수 있는 기술 개발이 필요하다. 필요에 따라 컴퓨팅 자원을 동적으로 할당받는 시스템을 구축하여 비용을 최적화할 수 있다. 일반적으로는 위와 같은 방식으로 처리 성능 및 비용 문제를 해결하지만, 본 프로젝트에서는 AI 모델 경량화 기술인 Quantization을 통해 기존 딥러닝 모델의 표현력을 유지하면서, 더 적은 비트를 사용하여 모델의 메모리 사용량을 줄이고, 이를 통해 추론 과정에서 동작 시간을 단축한다.

4. 시스템 구조

a) 모델 학습 파이프라인



1. 모델 학습 파이프라인 개요:

프로젝트의 모델 학습 파이프라인은 크게 데이터 수집, 데이터 전처리, 모델 훈련 및 모델 평가의 네 단계로 구성된다. 이 파이프라인은 다양한 인종과 헤어 타입을 반영할 수 있는 대규모 헤

어스타일 데이터셋을 기반으로, GAN을 활용하여 실제와 유사한 헤어스타일을 생성하고, 이를 통해 모델의 성능을 지속해서 향상하기 위한 학습을 진행한다.

2. 데이터 수집 및 전처리:

모델의 학습을 위해서는 풍부하고 다양한 헤어스타일 이미지 데이터의 수집은 필수적이다. 다양한 방법을 통해 수집된 데이터는 사전 확인을 통해, 헤어스타일이 명확하게 식별할 수 있고, 다양한 헤어스타일과 조건을 포괄적으로 포함하도록 한다. 데이터 전처리 과정에서는 이미지 크기 조정, 정규화, 데이터 증강 등을 통해 모델 학습에 적합한 형태로 변환한다.

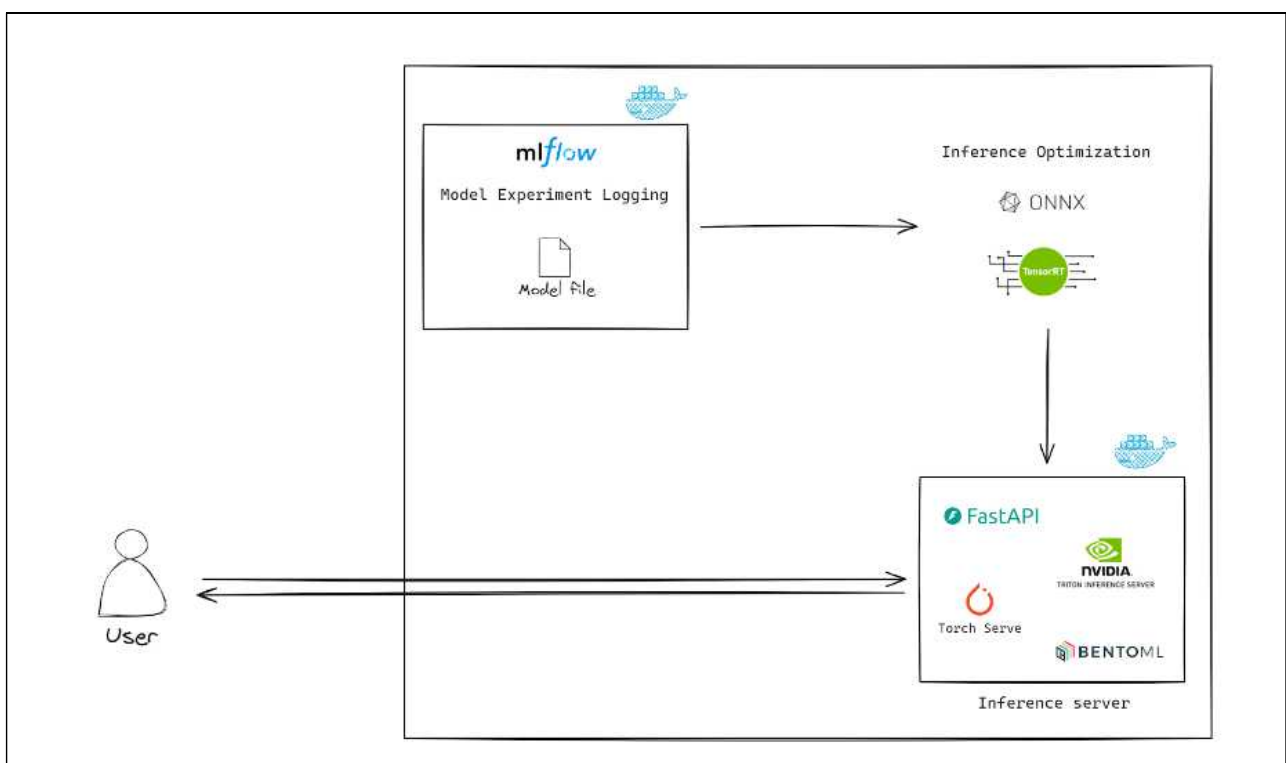
3. 모델 훈련

학습 데이터가 준비되면, GAN을 기반으로 하는 다양한 딥러닝 모델을 사용하여 헤어스타일 변환 모델의 학습을 진행한다. GAN은 실제와 구분이 어려운 헤어스타일 이미지를 생성할 수 있고, Diffusion 기반 모델보다 성능은 떨어질 수 있지만, 빠른 처리 속도를 통해 본 프로젝트에서 적합하다. StyleNet 등을 기반으로 하는 다양한 모델을 학습하여 결과를 비교하는 실험과 학습 과정에서는 손실 함수, 최적화 알고리즘, 학습률 등의 파라미터를 조정하며 최적의 모델 성능을 도출하기 위한 실험을 반복적으로 수행한다.

4. 모델 평가

모델의 훈련이 완료된 후에는 모델의 성능을 평가한다. 평가 기준으로는 이미지의 질적인 측면과 Reference style, image를 잘 적용하였는지를 판단하고, 본 프로젝트의 최종 목표는 생성형 모델이라는 점을 고려해서 변환된 헤어스타일의 다양성, 자연스러움 등을 종합적으로 평가한다. 이 과정에서 모델의 일반화 능력을 검증하고, 필요한 경우 추가적인 Fine-Tuning을 진행한다.

b) 모델 서빙 파이프라인



1. 모델 서빙 파이프라인 개요:

모델 서빙 파이프라인은 개발된 헤어스타일 변환 모델을 실제 환경에서 사용할 수 있도록 배포하고, 사용자의 요청에 따라 실시간으로 헤어스타일 변환 서비스를 제공하는 구조이다. 이 파이프라인은 API를 통해 애플리케이션과 데이터를 주고받고, 사용자의 입력 이미지 및 영상에 대해 실시간으로 헤어스타일을 적용하고 결과를 반환한다.

2. 기술 구성 방법론:

서비스의 핵심 기능을 지원하기 위해 다음과 같은 방법론을 후보군으로 지정하고, 테스트를 통해 가장 적합한 방식을 채택하여 모델을 서빙한다.

1) ONNX 변환 후 TensorRT로 최적화:

이 방법을 사용하여 모델의 실행 시간을 최적화하고 NVIDIA GPU에서의 성능을 극대화합니다. ONNX(Open Neural Network Exchange) 형식으로 변환된 모델은 Triton inference Server를 통해 모델 서빙 성능을 향상한다.

2) Torch Serve 사용:

PyTorch 기반 모델을 서빙하기 위해 Torch Serve를 사용한다. 이는 모델의 배포를 단순화하고 관리하기 쉽게 만들어 준다.

3) BentoML을 이용한 서빙:

BentoML은 다양한 ML 프레임워크를 지원하며, 모델 서빙을 위한 간단하고 효율적인 방법을 제공한다.

4) FastAPI 프레임워크를 이용한 웹 애플리케이션 서버 구축:

FastAPI는 사용자 친화적인 웹 인터페이스를 통해 서비스를 제공하며, 이를 사용하여 웹 애플리케이션 서버를 직접 제작하여 모델을 서빙하여 빠른 응답 속도와 동시성을 보장한다.

3. 서빙 아키텍처:

개발된 서비스는 mlflow를 통해 모델의 실험 및 로깅을 관리하고, 최종적으로 선택된 모델을 서버에 배포한다. 이 과정에서 다양한 서빙 기술 구성 방법론을 비교 및 테스트하여 최적의 방법을 선택한다. 최종 서빙 아키텍처로는 NVIDIA GPU를 활용하는 TensorRT와 Triton 서빙 조합이 성능 면에서 우수할 것으로 예상하지만, 실제 환경에서의 테스트를 통해 가장 적합한 서빙 방식을 최종적으로 결정할 예정이다.

최종 모델 서빙 파이프라인은 고성능과 사용 편의성을 동시에 제공하며, 다양한 플랫폼과 장치에서 헤어스타일 변환 기능을 효과적으로 사용할 수 있게 한다. 이를 통해 사용자는 언제 어디서나 자신의 외모에 새로운 헤어스타일을 적용해보고 최적의 스타일을 선택할 기회를 가질 수 있다.

5. 개발 환경 및 사용 기술

a) 개발 환경 및 언어

1. Host Machine:

- Host OS: Ubuntu 2.04
- Host CPU : AMD EPYC 7643 48-Core Processor
- Host GPU : A5000

2. Docker:

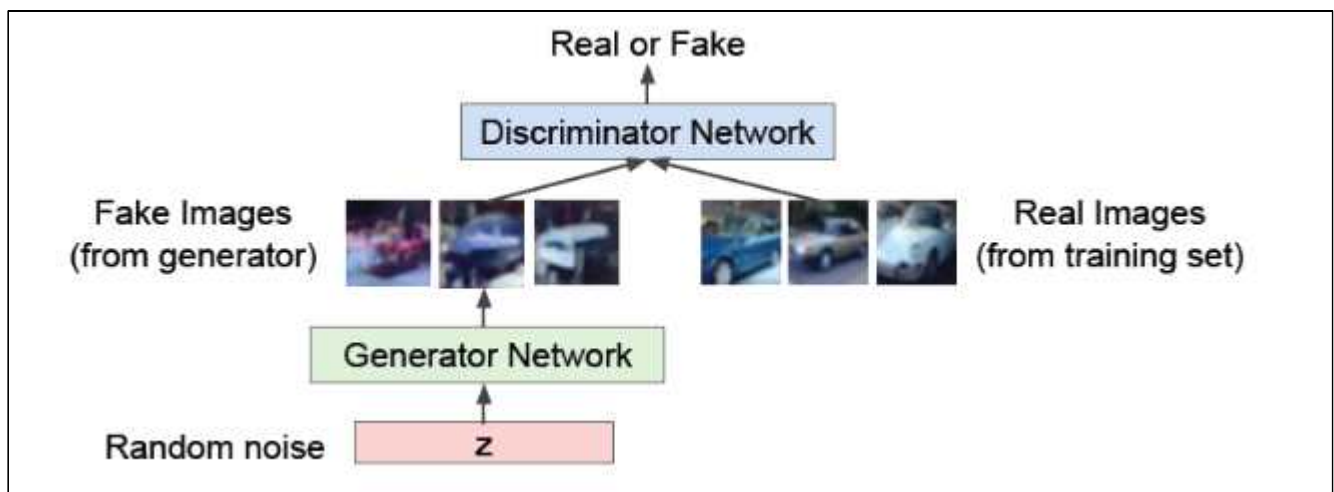
실험에 사용될 다양한 모델에 따라서 적합한 docker image를 사용한다.
(ex. pytorch/pytorch 2.3.0-cuda11.8-cudnn8-devel)

3. Language:

- Programming Language : Python
- DL FrameWork : PyTorch

b) 사용 기술

1. GAN (Generative Adversarial Networks):



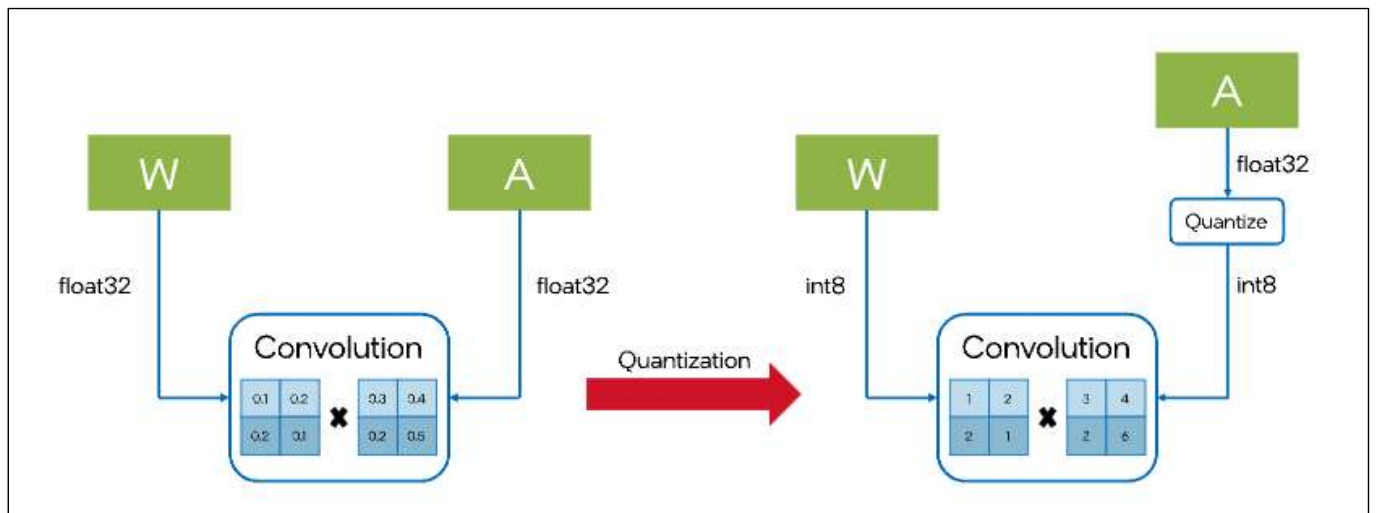
GAN은 비지도학습에 사용되는 대표적인 머신러닝 프레임워크의 한 종류이다. 2개의 신경망 (Neural Netork)으로 구성된 생성형 모델로, Generator은 실제 데이터와 유사한 새로운 데이터를 만들기 위한 학습을 진행하여 discriminator를 속이려고 시도하고, Discriminator는 실제 데이터와 generator를 통해 만들어진 가짜 데이터를 구별하기 위해 학습을 진행한다.

일련의 과정을 통해 서로 경쟁하면서 학습을 진행하면서, Generator은 더 실제와 같은 데이터를 생성하게 되고, Discriminator는 더 실제와 가짜 데이터를 잘 구별할 수 있게 된다. 즉, Generator가 실제 데이터 같은 데이터를 생성하는 생성형 모델(Generative Model)로서의 역할을 수행하게 된다.

추가로, 최근 성능이 개선된 Diffusion을 기반으로 하는 생성형 모델보다는 GAN을 채택하여 생성 시간 단축을 통해 실시간 스트리밍 결과를 제공할 수 있도록 한다.

본 프로젝트에서는 GAN을 사용하여 사용자 입력 이미지와 영상에 대해 다양한 헤어스타일을 자연스럽게 적용하고, 실제와 구별이 어려운 이미지 및 영상을 만드는 것을 목표로 한다.

2. Quantization:

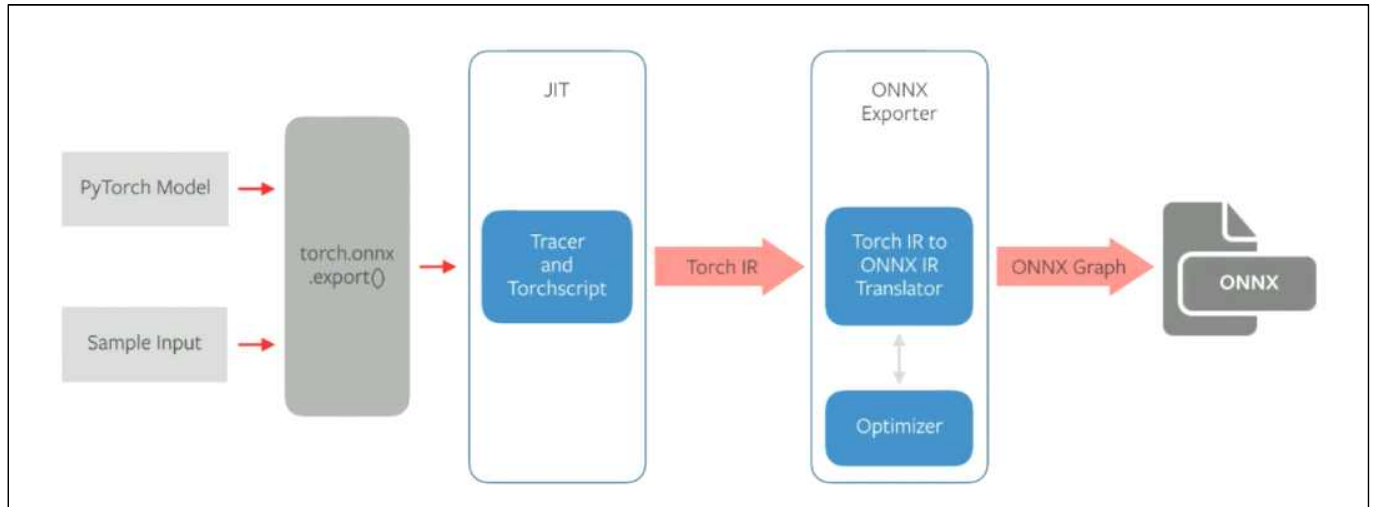


딥러닝 모델은 이미지 생성 분야에서도 좋은 성능을 보여주고 있다. 하지만 많은 메모리 공간과 연산량을 필요로 해서 효율이 떨어지는 문제점을 가지고 있다. 이 점은 본 프로젝트 구현에 있어서 큰 제약 사항으로 작용한다.

딥러닝 모델은 학습 시에 저장된 weight를 활용해서 수많은 연산을 해야 하는데, Quantization 기법을 활용하여, weight값 저장에 사용하는 비트의 수를 줄여서 모델 크기를 줄이는 방법을 통해 모델을 경량화한다.

기본적으로 숫자를 저장하고 연산할 때는 32-bit floating point (FP32) 자료형을 사용하는데, 추론 과정에서는 추론 속도의 가속화를 위해서 이를 8-bit int (int8)로 포팅시켜서 사용할 수 있다. 이렇게 자료형을 변환하여 모델을 경량화시키는 Quantization을 통해, 향상된 속도 대비 성능이 떨어지지 않도록 하는 것을 목표로 한다.

3. ONNX (Open Neural Network Exchange):



ONNX는 Tensorflow, PyTorch 등과 같은 서로 다른 딥러닝 프레임워크 환경에서 개발된 모델들을 서로 호환할 수 있도록 하기위한 프레임워크이다. 위와 같이 PyTorch에서 만들어진 모델을 ONNX 그래프로 export 하면, Tensorflow와 같은 다른 프레임워크에서도 해당 모델을 import 해서 사용할 수 있다.

단, 변환 과정에서 모델의 속도가 향상되지만, 성능 감소의 여지가 있고, 기존 프레임워크에서의 레이어가 달라질 수 있다는 점을 유의하여야 한다.

4. Existing Models:

GAN을 기반으로 한 생성형 모델 중에서 헤어스타일을 변형시켜주는 다양한 모델들이 존재한다.

1) MichiGAN:

MichiGAN은 헤어의 형태, 구조, 외관, 배경을 포함한 주요 시각적 요소를 명확히 구분하고, 사용자 입력을 다양한 조건 모듈을 통해 이미지 생성 파이프라인에 통합하여 사용자가 직관적으로 헤어스타일을 조작할 수 있도록 설계되었다. MichiGAN은 복잡한 헤어의 기하학적 특성과 외관을 다루는 데 있어서 뛰어난 사용자 제어력과 성능을 제공한다.

2) Barbershop:

Barbershop은 GAN 기반 모델로, 다중 이미지의 특징을 하나의 일관된 이미지로 유연하게 결합할 수 있는 새로운 latent space와 GAN-embedding 알고리즘을 제공한다. Barbershop은 세부 사항을 보존하고 공간 정보를 인코딩하는 데 효과적이다.

3) InterfaceGAN:

InterFaceGAN은 GAN의 latent space에 인코딩된 다양한 의미를 해석하여 픽셀 단위로 얼굴 편집을 가능하게 한다. Linear transform 후에 분리된 표현을 학습하여, 다양한 얼굴 속성을 보다 정밀하게 조절할 수 있도록 한다.

6. 일정 및 역할 분담

a) 개발 일정

Task	5월	6월	7월	8월	9월
Model Searching					
Model Experiments					
Model Transform					
Model Serving					
Construct WebUI					
Final Service Modifications					

b) 역할 분담

이름	역할 분담
박시형	<ul style="list-style-type: none">- 프로젝트 총괄 및 관리- 모델 최적화 및 수정- 모델 학습 파이프라인 구축
한지훈	<ul style="list-style-type: none">- 데이터 수집 및 전처리- 서비스 API 개발- 모델 경량화 적용 및 결과 분석
홍진욱	<ul style="list-style-type: none">- 서버 구축 및 관리- 서빙 방법론 실험 및 결과 분석- 모델 서빙 파이프라인 구축
공통	<ul style="list-style-type: none">- 관련 논문 분석 및 모델 탐색- 모델 실험 진행 및 결과 분석

7. 참고 문헌

a) Papers:

- Creswell, Antonia, et al. "Generative adversarial networks: An overview." IEEE signal processing magazine 35.1 (2018): 53-65.
- Polino, Antonio, Razvan Pascanu, and Dan Alistarh. "Model compression via distillation and quantization." arXiv preprint arXiv:1802.05668 (2018).
- Radford, A., Metz, L., & Chintala, S. (2016). Unsupervised Representation Learning with Deep Convolutional Generative Adversarial Networks. arXiv preprint arXiv:1511.06434.
- Karras, T., Laine, S., & Aila, T. (2020). StyleGAN2: Analyzing and Improving the Image Quality of StyleGAN. arXiv preprint arXiv:1912.04958.
- Ritschel, T., & Weyrich, T. (2015). Real-Time Hair Rendering on Mobile Devices. Journal of Computer Graphics Techniques.
- Zhu, Peihao, et al. "Barbershop: Gan-based image compositing using segmentation masks." arXiv preprint arXiv:2106.01505 (2021).
- Shen, Yujun, et al. "Interpreting the latent space of gans for semantic face editing." Proceedings of the IEEE/CVF conference on computer vision and pattern recognition. 2020.
- Tan, Zhentao, et al. "Michigan: multi-input-conditioned hair image generation for portrait editing." arXiv preprint arXiv:2010.16417 (2020).

b) Github Projects:

- MichiGAN: Multi-Input-Conditioned Hair Image Generation for Portrait Editing (SIGGRAPH 2020), GitHub Repository
(<https://github.com/tzt101/MichiGAN>)
- Hairstyle Transfer between Face Images, GitHub Repository,
(https://github.com/subrtade662/hairstyle_transfer)
- HeadNeRF, GitHub Repository
(<https://github.com/CrisHY1995/headnerf>)
- Style Your Hair: Official Pytorch Implementation, GitHub Repository,
(<https://github.com/Taeu/Style-Your-Hair>)
- Barbershop: GAN-based Image Compositing using Segmentation Masks, GitHub Repository,
(<https://github.com/ZPdesu/Barbershop>)
- Interpreting the Latent Space of GANs for Semantic Face Editing, GitHub Repository,
(<https://github.com/genforce/interfacegan>)
- Hairstyle Transfer, GitHub Repository,
(<https://github.com/Azmarie/Hairstyle-Transfer>)

c) Commercialized Services Research:

- Kittl, available at <https://www.kittl.com>
- Playground AI, available at <https://playground.com>