# 하이브리드 지능형 가상 캐릭터 동작 생성 개발

팀번호 : 15
팀 명 : HCB
분 과 : A
구성원 : 201724483 박현성
201724603 최우창
201724623 황원식

# 목차

# 1. 과제 배경 및 목표

## 1.1 과제 배경

인터넷 매체의 발전으로 인해 메타버스라는 새로운 형식의 사회를 만들어냈다. 메타버스란 가상현실, 증강현실의 상위 개념으로 현실을 디지털 기반의 가상 세계로 확장해 가상 공간에서 모든 활동을 할 수 있게 만드는 시스템이다. 메타버스와 가상 캐릭터는 불가분의 관계로 메타버스가 발전하고 대중화됨에 따라서 그 세계에서 살아갈 가상 캐릭터 역시 점차 기술적으로 진화하게 될 것이다.

가상 환경에서 더 자연스러운 동작을 재현하기 위해 주변 상황과의 연계를 통한 하이브리드 지능형 가상 캐릭터의 연구 및 개발이 필요하다. 하이브리드 지능형 가상 캐릭터란 음성, 텍스트 등의 대화 상황을 인지하여 얻은 데이터를 학습된 모델에 적용해 자연스러운 대화 상황을 위한 음성 및 동작을 생성하고 오류 상황 검출 시 컨트롤 주체를 변경할 수 있는 캐릭터를 의미한다. 이를 구현하기 위해서는 텍스트 분석을 통한 대화 모델, 텍스트 기반 동작 생성 모델, 오류 상황 발생 시 이를 검출할 수 있는 모델까지 총 3가지 해결과제가 존재하는데 이 중 2번 과제인 텍스트 기반 동작 생성 모델을 해결하려고 한다.

이번 과제를 통해 사용자-캐릭터 간의 상호작용 수준을 높여 가상환경상에서의 실존감과 응용서비스에 대한 몰입감을 높이고자 한다. 이는 언컨택트 시대 주목받고 있는 가상공간에서의 자연스러운 상호작용에 이바지할 수 있을 것으로 기대한다.

## 1.2 과제 목표

본 졸업 과제는 하이브리드 지능형 가상 캐릭터의 한글 텍스트 기반 동작 생성을 개발하는 것을 목표로 둔다.

**■ 한글 텍스트 트레이닝 데이터 셋 구축**

➜ 유튜브 영상에서 얻을 수 있는 영상과 자막을 이용하여 동작 생성 모델에 필요로 하는 데이터 셋을 구축한다.

**■ 한글 텍스트 Word Embedding 개발**

➜ 워드 임베딩(Word Embedding)은 단어를 벡터로 표현하는 방법으로, 워드 임베딩을 통

해 구한 임베딩 벡터(Embedding Vector)를 통해 모델을 학습시킨다.

**■ 가상 캐릭터 음성과 동작을 동기화**

➜ 가상 캐릭터 음성과 동작 동기화를 위해 텍스트의 단어별 시작 시각 타임 스탬프 할당이 필요하다.

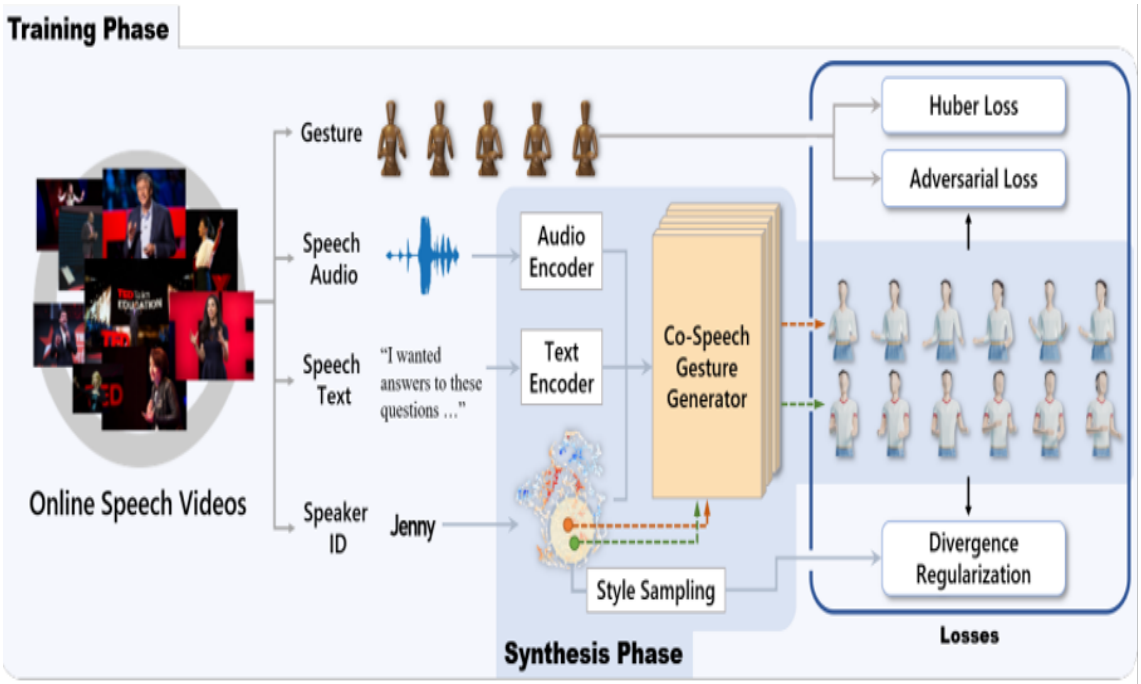
**■ 한글 텍스트 기반 가상 캐릭터의 동작 구현**

➜ 한글 텍스트 기반 동작 모델의 결과값을 Unity Programming에 적용시켜 가상 캐릭터가 동작 애니메이션을 수행할 수 있도록 한다.

➜ 애니메이션 결과를 분석하여 개선점을 찾아 가상 캐릭터가 자연스러운 동작을 수행할 수 있도록 한다.

# 2. 기존 사항 분석 및 전체 시나리오

## 2.1 기존 사항 분석




[그림 1] Speech Gesture Generation

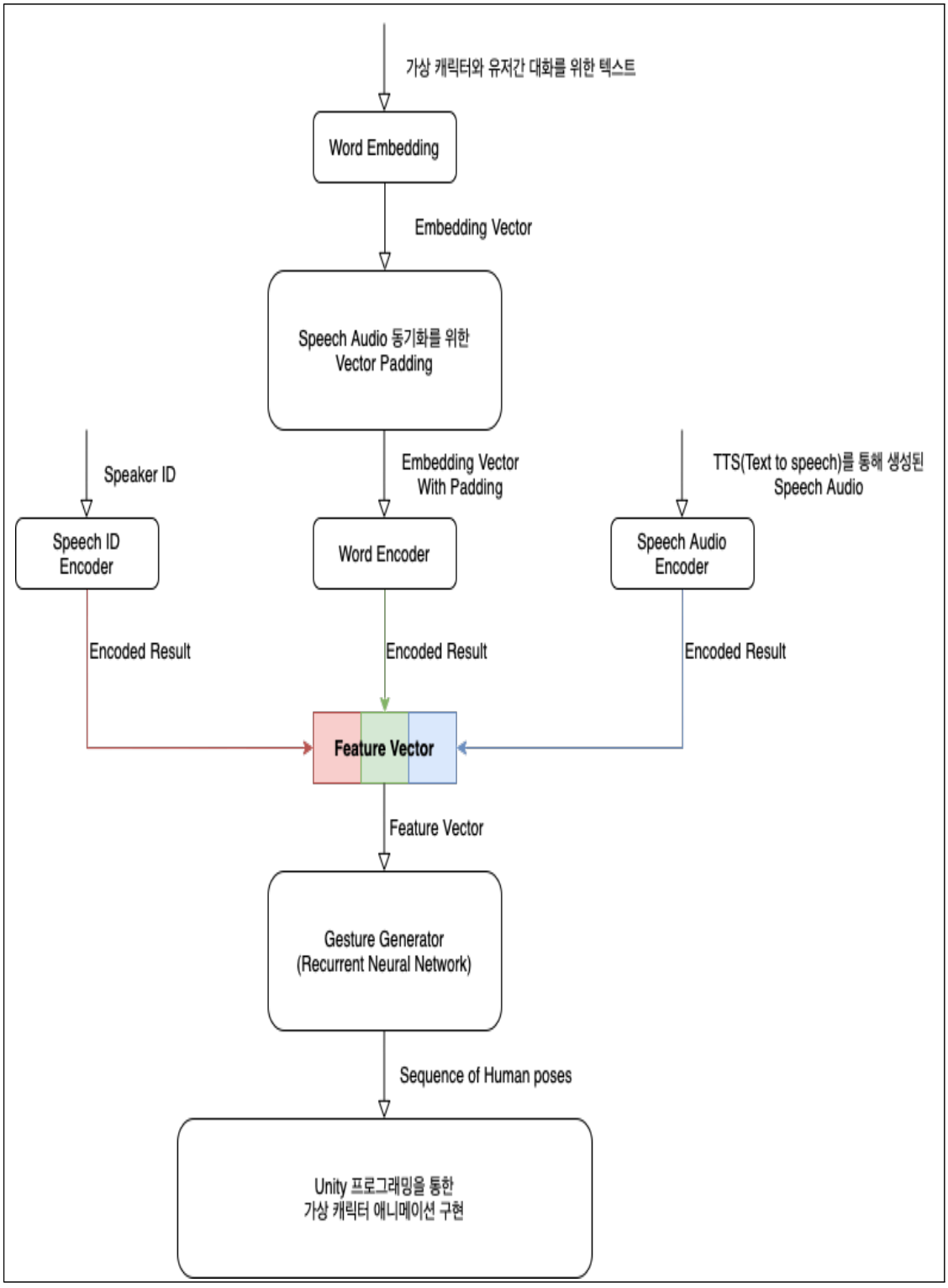
본 주제를 수행하기에 앞서 논문 조사를 통해 Text to Gesture에 관한 자료조사를 진행하였다. 그 중 Github에서 Text, Speech Audio, Speaker ID, Gesture 4가지를 통해 학습하여 텍스트에서 동작을 생성하는 프로젝트를 찾을 수 있었다.

이 프로젝트에서는 Speech Audio, Speech Text, Speaker ID 가 입력으로 주어지면 프레임 단위로 동일한 시간해상도를 가지는 Gestures Sequence를 출력한다. 따라서 텍스트로만 작동하기 위해서는 Google TTS 같은 Text To Speach API를 통하여 가상의 Audio를 만들어 입력으로 주어야 하는 필요가 있다.

결과로 나오는 Gesture는 10개의 Joint(척추, 목, 코, 머리, 양쪽 어깨, 양쪽 팔꿈치, 양쪽 손목)로 구성되어 9개의 방향벡터(부모 Joint에서 자녀 Joint로 향하는)로 나타내어져 있다. 9개의 방향벡터는 각각 척추-목, 목-코, 코-머리, 목-양쪽 어깨, 양쪽 어깨-양쪽 팔꿈치, 양쪽 팔꿈치-양쪽 손목의 방향을 가진다. 우리는 위의 모델을 이용해 한글로 동작 생성하는 것이 목표이기 때문에 Text Encoder의 동작을 분석하였다.

Text, Audio, Speaker ID는 서로 다른 Encoder로 가공된다. Text Encoder에서는 먼저 Text가 출력으로 나올 Gestures Sequence와 동일한 시간해상도를 보장하기 위해 Gesture Sequence와 같은 크기가 되도록 Text 사이에 Padding Token을 넣는 과정을 거친다. 예를 들어 I love you라는 텍스트에서 I와 love 사이에 잠시 멈추는 시간이 있을 때 생성된 Gesture 수가 5개이면 I ⋄ ⋄ love you 로 패딩 된다. 패딩 된 Word Sequence는 Word Embedding Layer에서 FastText를 통해 300개 차원을 가진 Word Vector로 가공된다. 이후에 이 Word Vector 들은 Temporary Convolutional Network(TCN)를 통해 32개 차원의 Feature Vector로 변환된다.

## 2.2 전체 시나리오




[그림 2] 시나리오 차트

# 3. 세부 과제 내용

## 3.1 과제 내용

**■ 한글 텍스트 트레이닝 데이터 셋 구축**

'세상을 바꾸는 시간'이라는 유튜브 채널에서 웹 크롤링을 이용해 영상과 자막 파일을 다운받는다. OpenPose 라이브러리를 이용해 다운받은 영상의 스켈레톤 데이터를 추출, PySceneDetect 라이브러리를 이용해 영상 장면을 분할한다. 스켈레톤 데이터와 분할된 장면을 비교 및 분석하여 필요하지 않은 장면을 삭제시킨다. 최종적으로 음성 및 동작 출력에 필요한 스켈레톤 데이터, 텍스트, 영상 장면만을 남긴다.

**■ 한글 텍스트 Word Embedding 개발**

FastText에서 157개 언어에 대해 사전 훈련된 모델을 배포한다. 배포되는 모델 중 한글어 모델도 존재하기에 이를 사용하려고 한다. FastText의 장점은 기존의 word2vector모델과 달리 데이터 셋이 충분할 때 내부 단어(subword)를 통해 모르는 단어에 대해서도 다른 단어와의 유사도를 계산하여 Word Vector를 얻을 수 있다는 점이다.

**■ 가상 캐릭터 음성과 동작을 동기화**

가상 캐릭터 음성과 동작 동기화에 필요한 Word Level Time Stamp Alignment 하기 위해 Gentle Forced Aligner를 사용했지만, 이 Tool은 영어에만 적용할 수 있어 이번 프로젝트에는 적합하지 않다. 따라서 우리는 한글 텍스트를 적용할 수 있는 Google STT API에 주목하였다. Google STT는 요청의 응답 텍스트에 시차(타임 스탬프)값을 포함하는데 제공된 오디오에서 인식되는 각 단어의 시작 부분과 끝부분을 표시한다. 인식된 텍스트에서 특정 단어를 검색하고 원본 오디오에서 찾아야 할 때 유용하다는 강점이 있다. 이를 이용하여 단어별 타임 스탬프를 할당할 것이다.

**■ 한글 텍스트 기반 가상 캐릭터의 동작 구현**

제스처 생성 모델의 결과인 상체 조인트 9개에 대한 방향 벡터들로 스켈레톤 파일을 만든다. 구현된 스켈레톤 파일은 가상 캐릭터 애니메이션에 사용된다. Unity에서 애니메이션 구현 시엔 Rigidbody라는 물리적인 속성을 사용해 자연스러운 움직임을 구현한다.

# 3.2 개발 환경 및 사용 기술

■ **PyTorch**


[그림 3] PyTorch

프로젝트 주요 개발 언어 중 하나로 데이터 분석 및 그래프 등의 시각화와 머신러닝을 위해 Python을 사용한다. 특히 머신러닝을 위해 PyTorch라는 오픈 소스 머신 러닝 라이브러리를 사용한다. Pytorch는 Python 환경에서 돌아가며 유용한 Python 라이브러리와 호환이 잘된다. 또한 Define and Run 메커니즘과 달리 Define by Run 메커니즘을 가진 PyTorch는 직관적인 모델링으로 인해 비교적 낮은 난도를 가졌다. 이러한 장점이 프로젝트를 진행하는 데 많은 도움이 될 것 같아 선택했다.

■ **OpenPose**


[그림 4] OpenPose

OpenPose는 인간 자세 예측의 한 분야로 카메라 가지고 사람의 몸, 얼굴, 손가락 마디를 정확하게 예측하는 라이브러리입니다. 동작 생성 모델의 학습을 위해 스켈레톤 데이터가 필요로 하게 되는데 OpenPose 라이브러리를 사용하여 이를 해결할 수 있다.

■ **PySceneDetect**


[그림 5] PySceneDetect

PySceneDetect는 비디오의 샷 변화를 감지하고 비디오를 자동으로 별도의 클립으로 분할하기 위한 응용 프로그램이자 파이썬 라이브러리이다. 독립 실행 파일로 단독으로

사용할 수 있으며, 다른 응용 프로그램들과 함께 비디오 처리 일부로 사용할 수 있고, 파이썬 API를 통해 다른 프로그램/스크립트에 직접 통합될 수 있다. 동작 애니메이션의 샷 변화의 감지 및 분류를 위해 PySceneDetect 라이브러리를 사용한다.

■ **FastText**


[그림 6] FastText

FastText는 Facebook의 AI Research lab에서 만든 단어 임베딩 및 텍스트 분류 학습을 위한 라이브러리다. FastText는 하나의 단어에도 여러 단어가 존재하는 것으로 간주하여 내부 단어를 고려하여 학습합니다. 본 프로젝트의 경우 동작 생성 이전 한글 텍스트 임베딩을 위해 FastText에서 제공하는 미리 학습된 한국어 모델을 사용한다.

■ **TTS**


[그림 7] TTS

TTS란 모델로 선정된 한 사람의 말소리를 녹음하여 일정한 음성 단위로 나눈 다음, 부호를 붙여 합성기에 입력했다가 지시에 따라 필요한 음성 단위만을 다시 합쳐 말소리를 인위로 만들어내는 기술이다. 동작 생성 모델의 오디오 입력과 가상 캐릭터의 음성 출력을 위해 사용된다.

■ **STT**


[그림 8] STT

STT란 사람이 말하는 음성 언어를 컴퓨터가 해석해 그 내용을 문자 데이터로 전환하는 처리를 말한다. 동작 생성 모델의 음성 출력과 동작을 동기화하기 위해 사용된다.

■ **Unity**

[그림 9] Unity

유니티는 3D 및 2D 비디오 게임의 개발 환경을 제공하는 게임 엔진이자, 3D 애니메이션과 건축 시각화, 가상현실 등 인터랙티브 콘텐츠 제작을 위한 통합 제작 도구다. 가상 캐릭터 생성과 애니메이션 동작 구현을 위해 Unity 3D를 이용한다.

# 4. 개발 일정 및 역할 분담

## 4.1 개발 일정

| 6월 | | | | 7월 | | | | | 8월 | | | | 9월 | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 1주 | 2주 | 3주 | 4주 | 1주 | 2주 | 3주 | 4주 | 5주 | 1주 | 2주 | 3주 | 4주 | 1주 | 2주 | 3주 | 4주 | 5주 |
| 기존 모델 분석 | | | | | | | | | | | | | | | | | |
| | 데이터 수집 | | | | | | | | | | | | | | | | |
| | | TTS, STT 기술 공부 | | | | | | | | | | | | | | | |
| | | Word Padding 기법 개발 | | | | | | | | | | | | | | | |
| | | | | | | 중간 보고서 작성 | | | | | | | | | | | |
| | | | | | | | 한글 Word Embedding, TTS, STT 적용 | | | | | | | | | | |
| | | | | | | | | | 데이터 학습 | | | | | | | | |
| | | | | | | | | | | | | Unity 가상 캐릭터 동작 구현 | | | | | |
| | | | | | | | | | | | | | | 최종 테스트 및 디버깅 | | | |
| | | | | | | | | | | | | | | | 최종 발표 및 보고서 준비 | | |

## 4.2 역할 분담

| 이름 | 역할 |
|---|---|
| **최우창** | - 한글 텍스트 기반 TTS, SST 적용<br>- 한글 Word Padding 기법 개발 |
| **황원식** | - 동작 생성 모델의 데이터 학습<br>- Unity로 가상 캐릭터 동작 구현 |
| **박현성** | - 모델 학습에 필요한 데이터 수집<br>- 한글 Word Embedding 기법 개발 |
| **공통** | - 논문 분석을 통한 기존 모델 학습<br>- API 및 관련 기술 공부<br>- 테스트 및 디버깅을 통한 성능 평가<br>- 보고서 작성, 발표 및 시연 준비 |

# 5. 참고 자료

## 5.1 참고 자료

1. 참고 논문 사이트
https://paperswithcode.com/task/gesture-generation

2. 참고 논문 Github
https://github.com/ai4r/Gesture-Generation-from-Trimodal-Context

3. FastText
https://fasttext.cc/docs/en/crawl-vectors.html

4. Google TTS
https://cloud.google.com/text-to-speech

5. Google SST
https://cloud.google.com/speech-to-text/docs/tutorials?hl=ko

6. HyperCLOVA
https://clova.ai/ko