BUILD SPEAKING HUMANOID

- TUSHAR ARORA

Keyframe
Lips Mapping

Blender

APPROACH

Text-to-Speech (TTS)

- Choose TTS Framework: Coqui TTS, Mimic TTS, Mozilla TTS, Praat, Mary TTS, Festival, etc.
- Dataset If we make from scratch: LibriSpeech, LJSpeech datasets for English
- Select a Pre-trained Model: Tacotron2, WaveGlow, WaveNet, FastSpeech2

What i chose and why?

- I choose the coqui framework with tts_models/en/ljspeech/tacotron2-DDC model because of its easy of use, opensource, fine-tuning, voice cloning, performance, community, and documentation.

Problem in Coqui Framework

- Given that installing and using Coqui TTS with pip might still be considered as utilizing a pre-built API

Alternative from Strach

- So without any API, we make it from scratch using pytorch audio with an English dataset named LJspeech(2.6 GB). Train with tecotrone2, and Implement Train WaveGlow for converts mel-spectrograms to waveforms then make pipeline and make model

Real life cases

- If we are making an advertisement project, do we need a personality person’s voice dataset or do we need to do finetuning in the coqui framework

Approach like Synthesia.io

- Text Preprocessing: Remove unwanted characters, format the text
- Text-to-Speech (TTS) Engine
- Prosody and Voice Modeling: Prosody (rhythm, stress, and intonation). Voice characteristics such as pitch, speed, and timbre are also modeled based on the selected voice profile (e.g., male, female, or specific accent).
- Neural Network-based Synthesis: WaveNet or Tacotron
- Post-processing and Audio Enhancement: clarity, remove artifacts, and adjust its volume and tone to match the desired effect.
- Integration with Avatar Animation: Synchronized with the avatar's lip movements and facial expressions.

- One more thing we can use is Convert audio to individual phonemes for lips mapping like “ah," "oo," "th”, etc.

- For example, for the phoneme "ah," you would create a keyframe where the character's lips are open in a blender, and for the phoneme "oo," you would create a keyframe where the character's lips are rounded.
- Match Keyframes to Speech Audio
- Smooth Out Transitions
- Fine-Tune the Animation

APPROACH

3D MODEL

- 3D Character Creation and Rigging: Use Daz 3D or Blender to create a realistic 3D model of a humanoid. Rig the model with appropriate bones and controls to enable facial expressions and body movements. Create blendshapes for various facial expressions (happy, sad, angry, etc.).
- Pre Build Model Website: Makehuman, MBlabs, Blenderkit, CCO model

Alternative Solution

- We can use DeepFake
- Deep Learning Technique

Example of Zomato GENAI

- Let’s take the example of “Zomato GENAI Advertisment”
- https://www.youtube.com/watch?v=SNfls_EeX34

- Data Collection: Replicate human speech patterns, facial expressions, and gestures.
- Model Training using (CNNs) for facial expression recognition
- Fine Tuning

- CNN & RNN are Used for image generation, facial expression recognition, and speech analysis.
- Conditional GANs for condition inputs
- NLP for analyzing the speech audio for synchronizing lip
- Video Editing

Combination

- Use Blender or Similar Technology for 3D Modeling: Modeling, Rigging and Animation
- Deep Fake Techniques: Face Swap, Voice Cloning
- Deep Learning Techniques: Motion Recognition, Speech Recognition, Interactive Dialogue

Another way to make a 3d Rigged Animation Model Using Blendshape & Livelink face App

- https://www.youtube.com/watch?v=oIuFg8GyShI&t=23s

Lip Movement Synchronization

- Use phoneme-to-viseme mapping to synchronize lip movements with speech using praat to create phonemes now we need to map phoneme to viseme. CMU Pronouncing Dictionary
- Deep Learning: Trained on vast amounts of video data containing speech and corresponding facial movements & Audio Waveform Analysis.
- Tools:
- For speech synthesis, DeepSpeech or Tacotron.
- For facial expressions and lip synchronization, you can use Blender or similar tools.

Let’s see How synthesia.io Work

- Text-to-Speech (TTS)
  - Tools: Tacotron, WaveNet
  - Technology: Deep Learning, NLP
- Phoneme and Viseme Mapping
  - Tools: Speech recognition algorithms
  - Technology: Phoneme extraction, Viseme mapping
- 3D Facial Animation
  - Tools: Blender, Maya
  - Technology: Blend shapes, Skeletal animation, GANs, SyncNet
- Rendering and Compositing
  - Tools: RenderMan, Arnold, Unreal Engine, After Effects, Nuke
  - Technology: Real-time rendering, Post-processing
- Synchronization and Final Output
  - Tools: Adobe Premiere Pro, Final Cut Pro
  - Technology: Video editing, Quality assurance

Let’s see How synthesia.io Work

- Text-to-Speech (TTS)
  - Models: Tacotron 2, WaveNet, FastSpeech.
  - Libraries: Mozilla TTS, TensorFlow TTS, OpenTTS.
- Phoneme Extraction
  - Tools: Montreal Forced Aligner, CMU Sphinx, custom phoneme extractors.
  - Libraries: Phonemizer.
- Viseme Mapping
  - Custom algorithms for mapping phonemes to visemes.
  - Libraries: viseme-mapping tools.
- 3D Model Animation
  - Software: Blender, Maya, 3ds Max.
  - Libraries: Rigging and animation plugins, custom scripts.
- Facial Expression and Emotion Integration
  - Tools: Facial action coding system (FACS) based rigs, emotion detection algorithms.
  - Libraries: Dlib, OpenCV for emotion detection.
- Background and Contextual Animation
  - Software: Blender, After Effects, custom scene generators.
  - Libraries: Animation and rendering plugins.
- Rendering and Post-Processing
  - Software: Blender (Cycles, Eevee), Arnold, After Effects.
  - Libraries: Rendering and post-processing plugins.

Blender Rhubarb Lipsync

https://www.youtube.com/watch?v=6MN6eui4zpE