Enabling Multilingual Communication: Automated Lip-synchronization Dubbing for Albanian Videos

ALDO DIKU

UNIVERSITY OF NEW YORK TIRANA

JULY 2025

This is to certify that I have read this project and that, in my opinion, it is fully adequate, in scope and quality, as a thesis for the degree of Bachelor of Arts in Computer Science.

(Title and Name)
(Project Advisor) _____ _____

This is to confirm that this thesis complies with all the standards set by the Department of Computer Science of University of New York Tirana.

Date: Seal/Signature:

I hereby declare that all information in this document has been obtained and presented in accordance with academic rules and ethical conduct. I also declare that, as required by these rules and conduct, I have fully cited and referenced all material and results that are not original to this work.

First Name, Last Name:

Signature:

ABSTRACT

ENABLING MULTILINGUAL COMMUNICATION: AUTOMATED LIP-SYNCHRONIZATION DUBBING FOR ALBANIAN VIDEOS

Diku, Aldo.
BA. in Computer Science
Thesis Advisor: Prof. Miralda Çuka
June 2025, 20 pages

Using available methods and tools (which i will mention when i gather all the tools and methods) to create lip-synced dubbed videos from an albanian video input and outputting a video of the same speaker in another language with the lips moving according to that languages movements.

Keywords: Lip-sync, dubbing, albanian language

First and foremost, I would like to express my sincere gratitude to my advisor, Professor Miralda Çuka, for their invaluable guidance, unwavering support, and insightful feedback throughout this project. Their expertise and encouragement were instrumental in shaping this work.

Finally, I am grateful for the support of my family and friends. Something something a little bit longer and better made.

Contents

3 Introduction

3.1 Background of the study

The increasing globalisation of online content consumption presents a significant challenge for video creators aiming to reach diverse audiences, thereby creating the need to produce multilingual versions of their work. Being able to make only one video and have it transform into multiple languages will help creators but also the target audience as well. Cutting down on the time and money it takes to make two or more videos with the same subject but in different languages. One such case is a youtube creator named der8auer from Germany, for the same topic he creates two videos, one in english and one in german. Sometimes his videos are quite long and having to do them again in another language will tire you out. Having a model that can take your video, translate it into another language and synchronize your lips to the movement of the language of your choosing would be a tool in your arsenal. Some of the usages of this lip-sync dubbing technology are: video dubbing and translation, real-time Face-to-Face translation that will be possible in the future and multilingual communication, gaming and virtual environments reduced cost and labor and time in this case, entertainment and content creation, speech recognition and lip-reading. Film, entertainment and media production. Education and training videos, especially in cases where there is a diverse workforce. It also has promising applications in areas for speech therapy, language learning and assistive communication devices for individuals with hearing impairments [6]. Advancements in Machine Learning (ML) have made it possible to have Automatic Speech Recognition (ASR)

3.2 How lip synchronization works

Phonemes are the distinct units of sound in speech, and visemes are visual representations of phonemes. Think phonemes as the audio and visemes as the video. Visemes serve to map sounds (phonemes) to corresponding mouth positions or shapes. The goal is to go from the audible to the visual (visemes). Visemes group similar-looking mouth shapes, which reduces the complexity required for animation. Relating the extracted audio features to the corresponding visual mouth shapes and facial expression. Mapping phonemes down to a smaller number of visemes gives artists fewer expressions to pose. The history of lip-syncing comes from animation and in particular two-dimensional cartoons where artists had to make facial movement according to the sounds of speech [14]. Lip-synchronization is a field that has been studied since 1994 [16], where Automatic lip-sync (ALS) could be used for animating cartoons realistically making the mouth movement more smoother and have a more natural feel and could also help in aiding people with hearing disability. Later in 1997 a new method was proposed where they used Fast Fourier Transform (FFT) for speech signal analysis, statistical measures called "moments" are used to describe the shape of the FFT. Mouth parameter measurements are used to measure the jaw position, the height of the maximum vertical opening between the lips and horizontal opening of the lips. This method did not require phonemic analysis or prior knowledge of the speech content. ObamaNet was an innovative architecture released in 2017, it used a time-delayed Long Short-Term Memory (LSTM) network to produce synchronized lip-synch videos. The network learned to map raw audio feature to corresponding mouth shapes based on hours of high-quality videos of a specific target person [32]. Convolutional Neural Networks (CNN) played an important role in lip-synchronization as it helped in feature extraction for

visual features, and image processing and generation. CNNs are often used with other models like LSTM blocks, or encoder and discriminator networks based on a Generative Adversarial Networks (GAN). A study by Pawar et al. (2024) focused on Marathi, an Indian language which has a shortage of datasets, similar to that of the Albanian language [24]. Integrating CNNs with adversarial training and including RNN-based architectures, specifically LSTM for lip-synch generation, Song et al. (2019), Sadoughi et al. (2021) and Li et al (2021) also used a combination of CNN, RNN, GANs and LSTMs [13] [19]. Another similar study by Exarchos et al. (2024) for lip-reading in Greek language tackeld the scarcity of datasets available. The study proposed a combination of 3D CNNs and LSTM networks for word recognition from lip movements.

4 Methodology

A key limmitation encountered was the lack of suitable Albanian video datasets, specifically open-license content with direct-to-camera speech, reflecting the broader challange of working with low-resource languages in multimedia applications. To address the first issue, custom video contennt was created featuring self-recorded footage in a one by one aspect ratio and lowest camera setting to save. The first step for this project was the data creation, seeing as there is not a lot of free and publicly available free-use content I can use we had to create the data myself. Videos were filmed 50 centimeters away from the recording device, different surrounding light, camera position (slightly down or up), shaved and unshaved faces were recorded to check the quality of the lip-synching models if they could simulate the bearded area around the lips. Five videos were created with moving backgrounds to check how the lip-synching model with handle changing background. One of the videos contains the head moving and lips being out of clear view, checking the limits of the model. Videos were recorded at 1440x1440 resolution (1:1 aspect ratio) at 30 frames per second, with durations ranging from 10 seconds up till 5 minutes. The study at (citation needed for the optimal length of time of the videos) found that the optimal length time was for videos with one word in them. To adapt to this, a script was made to create small chunks of videos to be closer to that optimal format. Videos could have been created by following and reading a script thereby avoiding the transcipiton step, however we want to create a workflow that will take any video of an albanian speaker and turn it into a dub lip synched video.

Audio was then extracted from the videos using ffmpeg package, Pulse Code Modulation, Signed 16-bit, Little Endian (pcm_s16le), a lossless, uncompressed audio codec format, 16000hz sample rate and monophonic, meaning single audio channel. which will be used for transcription and for the base of the voice cloning tool before inputing it into the final lip synchronization model. For the audio extraction ffmpeg library was used, and audio was extracted in a lossless codec, 16000hz sample rate and signed 16-bit Pulse Code Modulation (PCM), per Google Speech-to-text API documentation [5].

The audio gets sent through an Automatic Speech Recognition model, converting spoken language into written text. and gets saved in a text file, each line contains the name of the file transcripted and a colon separates it from the text. After several tests of three models, Google Speech to Text Chirp 2, Kushtrims and Neura, the latter one was selected as being the most accurate [5] [17] [22].

Transcripted text is translated to a target language that is supported by the voice cloning model.

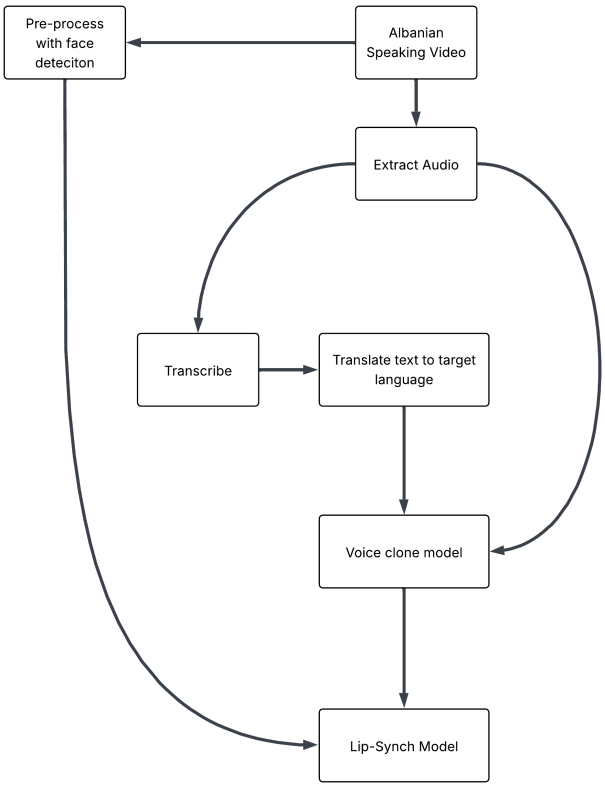

Figure 1: Work Flow

Using deep-translator python package and utilizing google translate as its core. Audio of the same file is used as reference to make a voice clone using OpenVoice, in the target language, a check is made if the cloned output is longer or shorter than the reference audio [25]. Another clone is made with the speed of the audio modified to be as close as possible to the original. OpenVoice offers emotion and accent cloning offering flexible voice control. The translated text will be used as the base of what the voice clone should output. OpenVoices Zero-shot cross-lingual voice cloning can generate speech that can be not present in the multi-lingual training dataset used by the model. There is "native" support and generalisation of voice clones. The native support is offered for languages like english, spanish, french, chinese, japanese and korean in the second version of OpenVoice.
The reference video is processed to be cropped in a 512x512, 25 frames per second format with the face centered, this was made using a face detection model which would detect faces in every frame of the video, from there we take the center and crop accordingly. Googles mediapipe framwork was used for the face detection task [9]. This format is a pre-requisite before inputting videos into the final lip-synching model. Pre-emptive tests were done on the videos to see if the face detector worked as intended and could detect a face in every frame of the videos.

After translating, the translated text and the original audio are used to make the voice clone. We need the original audio of the transcript because of emotions displayed in the audio can be used by the voice cloning tool.
Cloned voice and the original video from where the cloned voice was based on are then used to train the lip synchronization model.

One of the challanges of creating a lip synchronization in videos is the need for the lip movements to accurately align with a specified target speech segment, especially in multilingual and unconstrained environment. Visual data falling out of sunc with updated audio and inaccurate lip movements in target videos.
CNN and Generative Adversial Networks (GANs) to create the lip movements that sync up, this combined with a Discriminator Network which would try and detect the GANs fake lip movements and real ones, pushing each other to get better.

5 Audio transcription

The first step of the project is extracting audio from the videos and this was done using the open-source ffmpeg package. Going by the suggestions for optimizing transcription quality through googles speech-to-text API documentation [5], best practices, the audio was sampled at 16000 Hz, converted into a singned-integer bitrate, and in a lossless pcm_s16le format. One group of audio files were made in a lossless format, signed 16bit PCM (linear16), 16000hz sampling rate. While the other folder, was made with the same codec but with 48000hz sampling rate.
There is no open-source ASR model for Albanian language, and the amount of Albanian audio dataset is very low. Florijan (2023) managed to collect 40 hours of diverse accents, dialects and speech styles of Albanian audio, the purpose was to exceed an initial accuracy target of 45% [26]. This data was filtered out of 133 hours available, going through validation accuracy above 0.9 used for training. The base model used for this was Deep Speech 2 by Baidou [10], consisting of a combination of CNNs and RNNs. The model did managed to reach an accuracy of 46.3%, passing the threshhold but the model is not suitable for real world application. To test if there was another better model suitable for the dataset used by Florijan we used the Whisper model by OpenAI [23]. At first we tested the biggest model out of the box to check its performance, There are seven distinct whisper models whos difference lies in the parameter size and support for english only. Whisper is trained on 680,000 hours of labeled audio data, 438,218 hours of English language only, 117,000 hours covering 96 languages, Albanian included, though it does not mention the amount of hours for each language. Audio was sampled at 16000hz in mono, similar to that of Googles documentation. Its English ASR performance was very close to that of a human. Models range from tiny (39 million parameters) to large-v3 (1550 million parameters) these models were tested, attempts to fine-tune the large-v3 model with the dataset from Florijan we fruitless as the training time for such large parameter model was over 177 hours on a RTX 3080Ti 12GB Nvidia graphics card and 32GB of RAM. All models were tested from ones with the biggest parameter to the smallest, the only model that we could feasibly train was the whisper-small with a little bit more than 8 hours. ASR systems are evaluated using a Word Error Rate (WER) score, measuring the number of errors (substitutions, insertions and deletions) needed to transform the output of the ASR system into the reference transcript or ground truth. A lower WER score indicates a better performance in 1 we can see the formula used for the calculation. The fine-tuned whisper-small model achieved a 61.52% WER, not usable in any application. Testing the accuracy of the whisper-large-v2 model as a baseline reference without any fine-tuning resultet in a 55.1% Word Error Rate (WER) accuracy, the larger whisper-large-v3 model did not have much change and managed to score a 49.45% WER, both of these not up to par. Facebooks Wav2Vec2 baseline model with 960 hours of training

data was tested and managed to achieve a 97.74% WER [3].

$$WER = \frac{S + D + I}{N} = \frac{S + D + I}{S + D + C} \tag{1}$$

Where:

- S is the amount of substitutions,
- D is the amount of deletions,
- I is the amount of insertions,
- C is the amount of correct words,
- N is the amount of correct words.

From this point onward the search continued into closed systems, offered comercially. Googles Speech to Text API offered support for Albanian language and had free credits for first time user. Their system gives a confidence score for each transcription section and confidence score in individual words aswell. After transcription, a subtitle file format was also available to check the timings of when each word or sentence is spoken, this is an increadibly usefull feature as we could use it to cut the videos into smaller chunks as Pawar et al. (2024) experienced accelerated training process and improved performance when utilising a "vVISWA" dataset composed of isolated words [24]. In the first test Googles Speech to Text API, using Chirp 2 as a model, managed to achieve a WER score of 10.7%. Kushtrims model based on the whisper-large-v3-turbo and fine-tuned with 200 hours of curated Albanian audio with diverse dialect, hosted on the Hugginface Spaces, the model weights were not opensource, nor was the dataset used for the training [17]. It did not offer a confidence score nor a SRT file with audio timings.The model could be accessed for inference using the API available from Gradio, which is how the transcriptions were processed, however the model was quite slow in times as more than one individual would try to transcribe audio at the same time, increasing the transcription time a lot, in some cases taking more than three hours for twelve seconds of audio. At a later point the model was updated with an additional 100 hours of Albanian audio. Both versions of the model were evaluated, in table 1 we can see the scores for all models, and Kushtrims managed to get a 16.46% WER. The other ASR model tested was Neuras [22], a comercially available product, we were granted access and free credits. Their website advertises the model as having a 93% accuracy, which a bit of an underestimation on their part as it scored a 3.09% WER which is equivalent to 96.91% accuracy. The first set of audio files contained casual speech and did not have language that would be deemed difficult to understand or hear, testing was done for ten files, comparing the ground truth to the hypothesis file.
WER as a metric has its limmitations, from the formula we can see that the substitutions, deletions and insertions have the same weight. A minor misspelling might have the same weight as a word that completly changes the meaning of the sentence. It also doesnt take in consideration a mistake in letters inside a word. For example, the word "duket" was incorrectly transcribed as "duhet", in this case the whole word is a mistake. All unneccessary elements are removed, punctuation, making numbers into 123. The core of the WER calculation is finding the minimum number of edits needed to transform the hypothesis (transcription) text into the

Table 1: Error Analysis for Challenging Albanian Scripts

File ID	Neura			Kushtrim			Google API		
	S	**D**	**I**	**S**	**D**	**I**	**S**	**D**	**I**
File 1	0	0	0	2	0	0	1	0	0
File 2	0	0	0	8	0	0	7	0	0
File 3	0	0	0	27	4	3	14	17	1
File 4	27	7	4	53	15	24	25	7	3
File 5	2	0	1	31	3	7	18	7	6
File 6	0	0	0	6	0	0	2	0	0
File 7	0	0	0	12	2	6	13	2	6
File 8	0	0	0	6	0	1	4	0	0
File 9	2	0	0	12	0	3	11	0	5
File 10	2	0	0	10	3	2	5	2	0
Total Errors	**33**	**7**	**5**	**167**	**27**	**46**	**100**	**35**	**21**
Overall WER	**3.09%**			**16.46%**			**10.70%**		

reference (ground truth) text. This is typically done using a dynamic programming algorithm, specifically the Levenshtein distance algorithm, applied at the word level.

A second test was made reading along a script made with the purpose of being a difficult dialog. Five new videos were created, audio extracted and we tested Googles Chirp 2, Kushtrims and Neuras model as having the best results for a further analysis. Table 2 shows the results from the tested audio, very different results shown from the first test, Neuras accuracy dropped close to their advertised numbers, Kushtrims model was the one that moved less, although we have to keep in mind that this was the model with 300+ hours of training data, dispite this the models performance decreased. An anomaly was detected with Kushtrims model output because the number of Insertions is 52, in the transcription there were added words "Piii" after the end of the audio file which was not present in the other models. Removing these anomalies the models performance improves to 13.01%, the number of insertions dropped from 52 to 30. Googles Chirp 2 had the best results, achieving a 5.17% WER score, this could be the result of having a larger and more diverse dataset compared to Neuras model which might have more audio content with everyday speech rather than scienfic language.

Implementation of Googles API was the most challanging of the three models, as there were many different configurations, and the results varied when accessing it through the API. While Google's web UI offered more consistent transcriptions, the process required manual effort. Having the better accuracy of all tested models on the majority of the dataset and because of the nature of the audio content of the dataset being conversational language, Neura was chosen as the transcription model to go forward with the project. **Later tests on Kushtrim model show a lot of errors on the base 63 audio file, repetions of words. Not sure if it should be included.**

Table 2: Second Analysis WER score

File ID	Neura			Kushtrim			Google API		
	S	D	I	S	D	I	S	D	I
File 1	3	0	1	4	0	0	4	0	0
File 2	8	0	1	4	0	0	4	0	0
File 3	5	0	0	6	2	52	2	17	1
File 4	5	1	1	4	1	1	5	1	1
File 5	5	0	1	6	0	1	2	0	2
Total Errors	**26**	**1**	**4**	**24**	**3**	**54**	**21**	**1**	**3**
Overall WER	**7.08%**			**18.49%**			**5.17%**		

6 Translation

Python package deep-translator was used to translate the text from the ASR model to the target languages. Deep-translator is a library that offers multiple translation models such as Google Translate, DeepL, ChatGPT and Microsoft Translator. ChatGPT requires an API key to make use of the Large Language Model (LLM). For our usecase, Google Translate was used as it is free and offers a good accuracy.

7 Voice Cloning

Voice cloning is a subtask of speech synthesis, using deep learning to create speech imitating a specific voice. Prioritizing the preservation of the identity of the target speaker. Modern systems use a modular architecture to achieve realistic speech synthesis that replicates a target speaker's voice [11]. Similar to lip-synch architectures, voice cloning also uses CNNs, GANs, Variatonal Autoencoders (VAEs), DIffusion Probabilistic models (DPM), RNNs and Transformer-based Models [2]. We characterise voice cloning into several approaches, speaker adaption where we fine-tune a Text-to-Speech (TTS) model to replicate a specific user's voice using limited data [2]. Few-Shot Voice Cloning (FS-TTS) follows the concept of the speaker adaption but specifies the amount of data required as it typically ranges from a few seconds to a maximum of five minutes. Zero-shot Voice Cloning (ZS-TTS) differs from the first two because it does not require fine-tuning for the TTS model. Instead it uses a speaker encoder which uses a short audio clip to generate speech with voice characteristics similar to the reference waveform [2]. Multilingual Voice Cloning focuses on TTS systems that can support multiple lanuages while maintaining speaker's characteristics. Subdivisions of multilingual voice cloning are cross-lingual which allows the model to clone a reference voice and generate speech in a new language even if it was not largly present in the dataset. Intra-lingual are models with voice cloning within the same language. The NAUTILUS system from [20] has the ability to clone untranscribed speech, using a small amount of untranscribed speech, cited as about five minutes. Experiments coparing performance with different amounts of adaptation data show that Mean Opinion Score (MOS) for naturalness and similarity can improve significantly from five seconds to 15 seconds of untranscribed audio, plateauing or showing only minor improve-

ment beyond that (e.g. 30 or 60 seconds) [30]. Noisy audio can negatively impact the quality of generated speech, a real-time voice cloning system notes that generated speech can have low intelligibility due to artifacts, such as murmurs, hums, or noisy audio [11]. Because of different trainning and test datasets different reasearch focus on metrics such as speaker similarity, speech naturalness, intelligibility. or computational efficiency. Azizah (2024) used WER and Character Error Rate (CER) to measure the accuracy of models, using Whiper to generate text from the speech and comparing it with the reference text [1]. This method of evaluation is used by several other models to measure their intellegibility [12] [4] [20] [1]. User opinions, surveys, interviews, social media analysis, AB preference test, Degradation Mean Opinion Score (DMOS), these are all subjective measures and used is several models evaluation, because in the end the voice is made for humans to hear. Yemin et al. (2025) which created a foundation model for real-time autonomous interaction and voice role-play [31]. OpenVoice acknowledges the inherent difficulty in objective numerical comparisons across many different studies due to a huge variety of datasets, scales, and core functionalities. Using the translated text into the language of our choice we can then try and use a voice cloning model to make the voice dubbing. A lot of models were available for this task, Nari Labs released Dia [21], Chatterbox from ResembleAI [29], OpenVoice [25], Viola [31]. Some models offered more options about voice control, OpenVoice offers tone color and style from a reference speaker, emotion, accent, rhythm, pauses and intonation. Chatterbox offers different values of exaggeration, pace and temperature. Pacing or audio speed is very crucial in our project because of the differences in length of the reference audio and the output audio in the target language may be shorter or longer. Chatterbox is a new Voice Cloning model, its main support is for English language and it is quite limited in duration as it cannot do more than 35-40 seconds before starting to lose quality. It offers options to control the voice cloning with CFG/Pace, random seed nad temperature, on the Huggingface Space it only allows for 300 characters maximum input which limits our abilities quite a lot. Trying different values for the pace does very little to the change of audio length as it mostly changes to although it is more natural than others, it still has issues as if the values of the pace change too much or the exaggeration value changes too much it can create artifacts, moments of silence of up to 4 seconds that did not exist in the reference audio. Chatterbox includes a built-in PerTh Watermarking for responsible AI usage.

Voice cloning can be of help to those with speech disability, as most of the models are trained on normal human voices. Azizah (2024) presented zero-shot voice cloning Tacotron-2 based TTS for people with dysphonia, the choice for zero-shot was because of the impossibility of training a TTS model using data from people who suffer from dysphonia [1].
Ethical questions regarding Voice Cloning are paramount, primarily surrounding individual rights, potential for misuse and societal impact. In a hugly digitalised world where people share a lot of video content online they might not know that their voice can be collected and used for trainning, raising questions about privacy infringement and data protection [12]. Issues with identity and authenticity will rise as synthetic voices become increasingly natual sounding. Potential for misuse and harm, malicious exploitation of voice cloning for deception, fraud, manipulation and financial scams [7].

8 Lip synch model

Before deep learning lip-sync was achieved using viseme-based and rule-based system, mapping phonemes to corresponding visual mouth shapes. These methods were more commonly used in cartoons as they lacked the realism and fluidity of natural speech [15] [18] [27]. Figure 2 shows the visemes for letters a, b, c, d, e, ë, f, g, j in the Albanian alphabet. Since every letter has a uniques sound in the Albanian language. Movement of the lips depend on the emotions

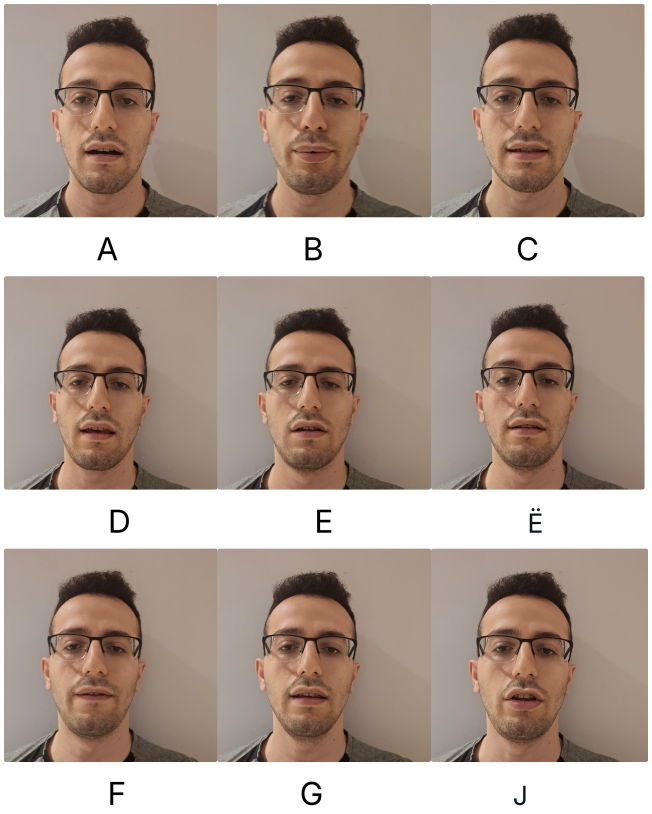

Figure 2: Visemes

of the speaker, researchers noticed that lip-syncing and broader facial animation are not the same across different emotional states like anger, sadness, happiness, surprise, fear etc. [**?**]. These emotional nuances can indeed be learned and trained by advanced models to be cloned or mimiced in generated outputs [28] [35]. Human evaluators consistentyl assess wether the generated output matches the emotional tone implied by the audio and they look for smooth transitions between emotinal expressions. Models like EmoTalk and EMOTE explicitly work to extract the emotions from the audio and integrate the effects of both emotion and speech to produce more nuanced emotional representations in 3D animations [**?**].

GeneFace++ is an improved version from GeneFace [34] [33], it progressed on three key areas, Audio-to-Motion Module, enhanced robustness (Landmark locally linear embedding), and improved efficiency (Instant Motion-to-Video Module). These additions aim to address the limitations of generalizability, video quality and system efficiency observed in the original GeneFace model. The nature of the architecture of the model, having a 3D model of the face can help with issues of Preparing the dataset before training was needed, the videos needed to

be scaled into 512x512, mediapipe [9] was used as a facedetector. An early issue came up as the face detector could not detect any face in all frames of the video, this was the same for all the videos in the dataset. Getting the metadata information out, we observed that videos had a rotation of 90 degrees. Mediaplayer applications would see this flag and add a 90 degree clockwise rotation to show the video correctly, but the face detection software would go through raw pixel data thus ignoring the flag and getting a sideways face. Most face detection models train on upright faces making them fail when faced with faces sideways or face-down. When applying a clockwise 90 degree rotation on the raw pixels of the video and changing the rotate flag to 0, the videos would come out as sideways, this might be beacuse ffmpeg might apply a correction rotation before the rotate command, thus we needed to apply two ffmpeg commands, one to rotate +90 degrees clockwise and set the rotate flag to 0, and the next command to rotate the video again with a -90 degrees counter-clockwise. The "min_detection_confidence" variable was also lowered from 0.7 as a default option to 0.6, with the new value the model was able to detect faces. Next step for processing data was audio feature extraction, using mel, f0, hubuert or esperanto. We extract individual image frames from the processed video, using lm2d_mediapipe we extract 2D facial landmarks from the video frames. These 2D landmarks are crucial for fitting a 3D Morphable Model (3DMM) in the next step. Fitting 3DMM, we take 2D facial landmarks and "fit" them in a 3D Morphable Model to the person's face in each video frame. This makes the model more suitable for usage on a specific person as we are training the NeRF model for the head and torso. This fits our dataset more as the videos are of a single person. The last step is about packaging and organizing the generated data into a binary format that is optimised for efficient loading and training of the subsequent neural network model. Handling of different audio and video lengths, although not specifically mentioned in the research we can infer based on the architecture, the model would likely continue to generate facial motion based on the remaining audio. We can test to see how the model performs with a different head and torso NeRF model using the one pre-trained by GeneFace++ or train them ourselves before inferencing. GeneFace++ is a model that works by training the Audio2Motion model which comes pre-trained in this case. We need to train Motion2Video renderer ourselves after the data has been prepared. Training is done for the head model and the torso model separatly. Finalizing all of these we can go to inference, we used the longest video in the dataset (five minutes) to train the head and torso model. The model for the head was stopped after one hour and 30 minutes after observing the lack of change in the total loss value. From Figure 3 we can see on graph Total Loss over Steps that the minimum value it reached (1.12 total loss) was at 15,000 and 20,000 steps. After it continued to fluctuate with a maximum of 20 and an average of 15.48. Mean Square Error Loss (MSE) and Super-Resolution Mean Squared Error Loss (SR MSE) go down at the begining of the steps and stay constant, while the Lambda Ambient has identical values to that of the total loss, this makes it the main factor contributing to the total loss value.

The models configuration was training for 250,000 steps. After 1 hour and 30 minutes of training the results were as shown in Figure 4. The total loss drops to the lowest value of 8.6979, after 20,000 epochs and continues rising till 384 in step 90,000. In Figure 5 we can see that MSE and SR-MSE keep going down consistently till step 30,000 where we can see fluctuations. The culprit of the high loss in our case is the lambda ambient loss, which we can see in Figure 6, as the line for the lambda ambient loss fits perfectly with the line of our total loss. This value measures how well the model is handling the ambient (non-facial, background, or global

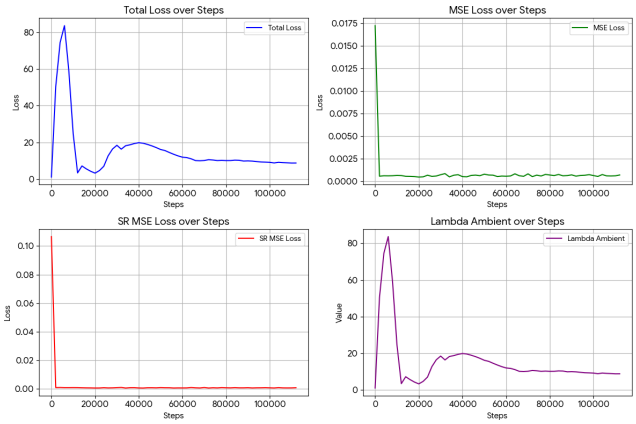

Figure 3: Head model training

lighting) aspects of the scene. Target ambient loss in the configuration files was changed from 1.0e-8 to 1.0e-6 which the authors say might be more suitable for men.

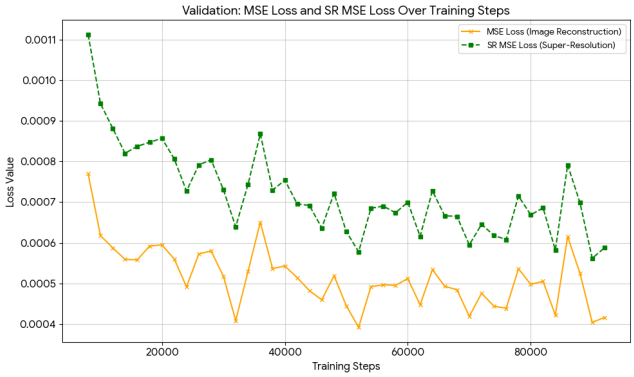

Figure 4: MSE, SR MSE loss

Figure 5: Total Loss

Figure 6: Lambda ambient Loss

Figure 7: Results from head model training

After the changes the ambientloss values were as shown in Figure 7. MSE and SR-MSE loss values were as shown in Figure 8. The model was trained for 198,000 steps before stopping as the loss values got below 0.001 as shown in Figure 9.

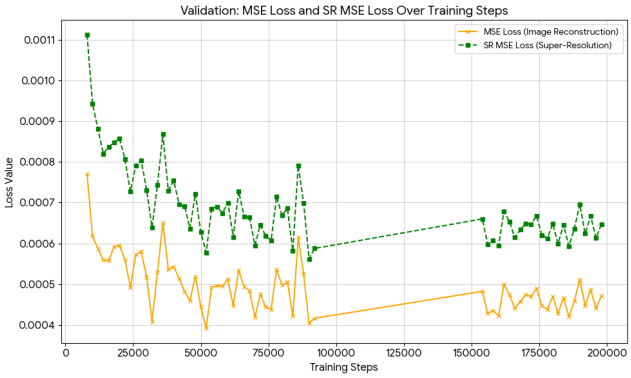

Figure 8: MSE second iteration

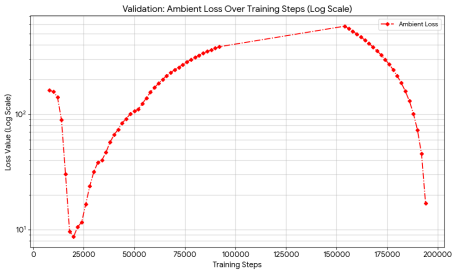

Figure 9: Ambient loss second run

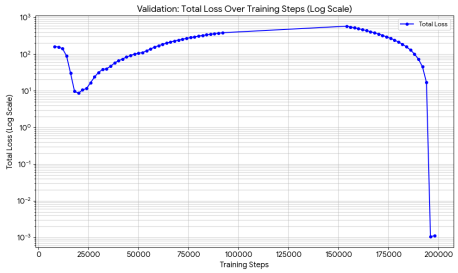

Figure 10: Lambda ambient Loss

Figure 11: Results from second training

The training of the Torso NeRF model is dependent on the performance of the pre-trained Head NeRF model. During the initial training of the Head NeRF model, suboptimal convergence behaviour rendered the resulting Head model unsuitable. Consequently, any Torso NeRF model trained using the badly optimised Head model is unreliable for the final generation pipeline.
A third training was done to the torso model after changes to the configuration files, lambda ambient value and eye blink dim changes from four to two. In Figure **??** we can total loss per training steps, the lowest total loss value occured at step 34,000. The training was stopped at 46,000 steps as the total loss was not decreasing more. At 8000 steps the models total loss is 0.104, showing a slight decrease to a minimum of 0.0768 at 38,000, before rising slightly again. It is largly plateauing within the range of approximately 0.076 to 0.112. This means the models overall performance on the validation set is no longer significantly improving.

Inference -

9 Future improvements

Creating an open-source albanian ASR model, using models like Kushtrims and utilizing the free credits from Google Speech-to-Text API, making use of the confidence score to save only audio that is above a certain threshold, to create a dataset that is more accurate and reliable.

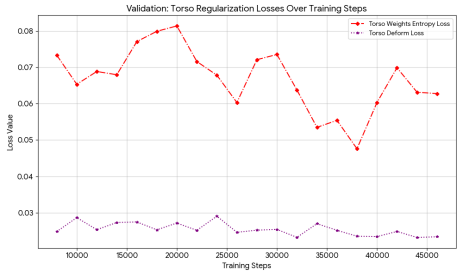

Figure 12: Weight Entropy loss

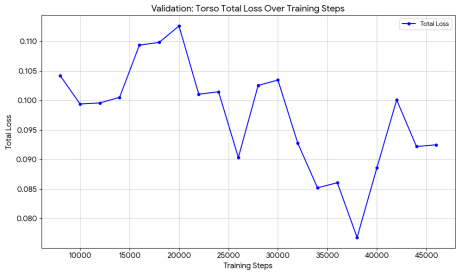

Figure 13: Total loss

Figure 14: Results from third torso model training

An ASR model of the target languagecan be used to verify the accuracy of the voice cloning model, by comparing the transcribed text with the translated text, for this we would need the target language ASR model to be very accurate.
LLMs can be used to translate the text, and could be of help when faced with discrepancy in audio timing of the reference audio and the cloned audio. Making the text shorter while keeping the meaning of the text the same. Changing the text to be more suitable for the voice cloning model. When faced with text that is of Albanian dialect, LLM could be used to transform the text to be more suitable for the voice cloning model.

10 Limitations

The project is limited to only the Albanian language as such the dataset is comprised of only Albanian speaking videos. Lack of open-source ASR models, and freely available datasets that fit into our requirements of having the person speak Albanian clearly, and facing the camera. Human evaluation bias in voice cloning, which is subjective and can be faulty or biased. Storage and computational power requirements, both local storage and cloud storage. Using Huggingface Spaces was problematic because the speed of audio transcription changes based on the availability of resources as everyone can use it, similar situation for the voice cloning models tried.
While Kushtrims model does say it is trained on Albanian audio with dialects, traditional translation models are not able to handle dialects, thus leading to a bottleneck in the project. Voice cloning models that could not produce audio more than 30 seconds created an issue in cases where the videos were longer than 30 seconds.
Different models had different ways to prepare the data for training, some models required the data to be in a specific format, creating a need to save the original data without changes then applying the format to create new data, increasing the amount of data stored.

10.1 Questions raised

List of questions:

- Do out lips move the same despite different emotions?

- Can we lip synch the same video with different emotions? Or just clone the video with the emotion expressed?
- How do we know if the transcription is correct programatically?
- How should the data be prepared for the lip-synchronization model?
- Video format, 30 frames per second or 25, quality of the video, resolution, etc.
- Audio format, mp3, lossless, bitrate, etc.
- How does the duration of video chunks used during preprocessing impact the performance of the final machine learning model, and what chunk length yields the best trade-off between information retention and computational efficiency?
- How are voice cloning models evaluated, as it feels like it should be done with a human evaluation and that is faulty most of the time?
- Which would be the best model for the lip-synchronization task based on the data we have?
- Best way to handle differences between original audio timing and the voice cloned sample timing. (slow down the voice? speedup)
- Is an LLM better than a traditional tool for translating? Neural Machine Translation (NMT) vs Large Language Models.

As for the optimal length of the video chunks, the study done by [24] faced challanges with lengthy continuous speech, inlcuding time-consuming merging issues and potential overfitting. This was addressed by using the "vVISWa" dataset which contains isolated words or indepentent speech. This improved the performance of the model a lot and reduced overfitting. However Goncalves et al. (2024) using the LRS3 dataset containing short sentences averaging 3.42 seconds, found that using such short sentences could lead to a drop in translation quality [8].

List of challanges:

- Low resources for the datasets in albanian, in video and audio.
- No open source models for the transcription of the audio from albanian.
- Voice cloning models being evaluated using a human evaluation and that is biased.
- Storage of data, both in the cloud and locally.
- Computing power, GPU, TPU, etc.
- Training time.
- If the spoken albanian is with a heavy dialect, an Albanian ASR trained with large number dialects is needed
- If the cutting of videos is not done correctly, it is possible to cut a word in half

- Translating the transcribed albanian dialect. How possible it is? NLP script transformation + translate*
- Voice cloning emotions.
- Video chunking first then transcription, or transcription (have to keep timings) then chunking - might be more possible and more cost effective than the NLP to find the problems in the language/chunked words but not for dialects.
- Could probably use the timings from google speech-to-text api (or any timings) to make the cuts since it offers that. Making the better option to be transcript first then cut/chunk second.
- Multiple Ethical challanges
- NLP + basic quantized LLM for better albanian understanding and translation?
-

List of improvements:

- Better dataset, quantity and quality, diversity (age, gender, moustache, skin color, lip color- lipstick, ethnicity, also in this case, dialect)
- Separation of dataset into single word video for better model performance.
- Albanian ASR open-source model, google speech-to-text API is paid service, and Kushtrims uses huggingface spaces which is slow
- Comparison between Google Speech-to-text API and Kushtrim ASR
- Addition on an NLP to handle transcription errors
- NLP for dialect speech, to transform it into something that can be translated, maybe LLM is better in this case.
- Better voice cloning. More emotional expression
- Better face reproduction from the emotions of the voice clone.
- Real time lip-synch dubbing
- Separating the dataset into two groups the new dataset with videos close to one word per video and see the results of that and the original dataset with longer videos.
- NLP can also be utilized in the audio length check, if we have too much of a difference in audio length, we could apply NLP-s to change the sentence into something shorter or longer but keeping the sentence about the same.

List of things to not forget to add:

- Mention confidence score in google api, it also offered timings like SRT file

- Kushtrim first 63 videos were on 200 hours of dataset, and the next 5 "hard" videos were on 300 hours. There can be a slight comparison on the accuracy. Should retest it.
- Write about the changes in translation from NTL to LLM being better, keeping context etc.
- Mention the fact that there is no open-source Albanian ASR model, no large open-source free-use dataset either.
- Lowering the confidence coeficient in the face detector as it was from 0.7 where it could not detect a face. no matter the image rotation and the issue with the -90 degrees in metadata.
- Also playing with the close face or far face, 0 or 1 values.
- In our case with GeneFace++, we are scaling the video in a 512x512 with the face being the center, this make the video much smaller, if we wanted a larger video, we would need to save both frames, scaled with GeneFace++ applied and the uncropped and paste the applied frame into the bigger frame.

References

[1] Kurniawati Azizah. Zero-shot voice cloning text-to-speech for dysphonia disorder speakers. *IEEE Access*, 12:63528–63547, 2024.

[2] Hussam Azzuni and Abdulmotaleb El Saddik. Voice cloning: Comprehensive survey, 2025.

[3] Alexei Baevski, Henry Zhou, Abdelrahman Mohamed, and Michael Auli. wav2vec 2.0: A framework for self-supervised learning of speech representations, 2020.

[4] Qi Chen, Yuanqing Li, Yuankai Qi, Jiaqiu Zhou, Mingkui Tan, and Qi Wu. V2c: Visual voice cloning, 2021.

[5] Google Cloud.

[6] Themis Exarchos, Georgios N. Dimitrakopoulos, Aristidis G. Vrahatis, Georgios Chrysovitsiotis, Zoi Zachou, and Efthymios Kyrodimos. Lip-reading advancements: A 3d convolutional neural network/long short-term memory fusion for precise word recognition. *BioMedInformatics*, 4(1):410–422, 2024.

[7] Genesis Gregorious Genelza. A systematic literature review on ai voice cloning generator: A gane-changer or a threat? *Journal of Emerging Technologies (JET)*, 4(2), 2024.

[8] Lucas Goncalves, Prashant Mathur, Xing Niu, Brady Houston, Chandrashekhar Lavania, Srikanth Vishnubhotla, Lijia Sun, and Anthony Ferritto. Improving lip-synchrony in direct audio-visual speech-to-speech translation, 2024.

[9] Google. MediaPipe Face Detector — Google for Developers. https://ai.google.dev/edge/mediapipe/solutions/vision/face_detector, 2024. Accessed: 2025-06-11.

[10] Awni Hannun, Carl Case, Jared Casper, Bryan Catanzaro, Greg Diamos, Erich Elsen, Ryan Prenger, Sanjeev Satheesh, Shubho Sengupta, Adam Coates, and Andrew Y. Ng. Deep speech: Scaling up end-to-end speech recognition, 2014.

[11] Weixin Hu and Xianyou Zhu. A real-time voice cloning system with multiple algorithms for speech quality improvement. *PLOS ONE*, 18(4):1–14, 04 2023.

[12] Onuh Ijiga, Idoko Idoko, L.A Enyejo, Omachile Akoh, Solomon Ugbane, and Akan Ibokette. ** Corresponding author: Onuh Matthew Ijiga Harmonizing the voices of AI: Exploring generative music models, voice cloning, and voice transfer for creative expression*, volume 11. JAETS, 03 2024.

[13] Diqiong Jiang, Jian Chang, Lihua You, Shaojun Bian, Robert Kosk, and Greg Maguire. Audio-driven facial animation with deep learning: A survey. *Information*, 15(11), 2024.

[14] Arien Kock Jonathan Gratch. An evaluation of automatic lip-syncing methods for game environments. Technical report, University of Twente, Department of Computer Science, University of SOurthern California, Institute for Creative Technologies, 2022.

[15] Jonathan Kock, Arien ; Gratch. An evaluation of automatic lip-syncing methods for game environments. Technical report, UNIVERSITY OF SOUTHERN CALIFORNIA LOS ANGELES, 2005.

[16] B.E. Koster, R.D. Rodman, and D. Bitzer. Automated lip-sync: direct translation of speech-sound to mouth-shape. In *Proceedings of 1994 28th Asilomar Conference on Signals, Systems and Computers*, volume 1, pages 583–586 vol.1, 1994.

[17] Kushtrim. Whisper large v3 turbo shqip. https://huggingface.co/spaces/Kushtrim/whisper-large-v3-turbo-shqip, 2025.

[18] John Lewis. Automated lip-sync: Background and techniques. *The Journal of Visualization and Computer Animation*, 2(4):118–122, 1991.

[19] Xiaohong Li, Xiang Wang, Kai Wang, and Shiguo Lian. A novel speech-driven lip-sync model with cnn and lstm. In *2021 14th International Congress on Image and Signal Processing, BioMedical Engineering and Informatics (CISP-BMEI)*, page 1–6. IEEE, October 2021.

[20] Hieu-Thi Luong and Junichi Yamagishi. Nautilus: a versatile voice cloning system, 2020.

[21] NariLabs. Dia. https://narilabs.org/, 2025.

[22] Neura. Neura.al. https://neura.al/, 2025.

[23] OpenAI. Robust speech recognition via large-scale weak supervision. Technical report, OpenAI, 2022.

[24] Diksha Pawar, Prashant Borde, and Pravin Yannawar. Generating dynamic lip-syncing using target audio in a multimedia environment. *Natural Language Processing Journal*, 8:100084, 2024.

[25] Zengyi Qin, Wenliang Zhao, Xumin Yu, and Xin Sun. Openvoice: Versatile instant voice cloning. *arXiv preprint arXiv:2312.01479*, 2023.

[26] Florijan Qosja. Development of an albanian language transcriber using artificial intelligence. https://github.com/florijanqosja/Albanian-ASR, 2023.

[27] Amirkia Rafiei Oskooei, Mehmet S. Aktaş, and Mustafa Keleş. Seeing the sound: Multilingual lip sync for real-time face-to-face translation. *Computers*, 14(1), 2025.

[28] Amirkia Rafiei Oskooei, Mehmet S. Aktaş, and Mustafa Keleş. Seeing the sound: Multilingual lip sync for real-time face-to-face translation. *Computers*, 14(1), 2025.

[29] ResembleAI. Chatterbox. https://www.resemble.ai/, 2025.

[30] Tasnima Sadekova, Vladimir Gogoryan, Ivan Vovk, Vadim Popov, Mikhail Kudinov, and Jiansheng Wei. A unified system for voice cloning and voice conversion through diffusion probabilistic modeling. In *Interspeech 2022*, pages 3003–3007, 2022.

[31] Yemin Shi, Yu Shu, Siwei Dong, Guangyi Liu, Jaward Sesay, Jingwen Li, and Zhiting Hu. Voila: Voice-language foundation models for real-time autonomous interaction and voice role-play, 2025.

[32] Supasorn Suwajanakorn, Steven M. Seitz, and Ira Kemelmacher-Shlizerman. Synthesizing obama: learning lip sync from audio. *ACM Trans. Graph.*, 36(4), July 2017.

[33] Zhenhui Ye, Jinzheng He, Ziyue Jiang, Rongjie Huang, Jiawei Huang, Jinglin Liu, Yi Ren, Xiang Yin, Zejun Ma, and Zhou Zhao. Geneface++: Generalized and stable real-time audio-driven 3d talking face generation. *arXiv preprint arXiv:2305.00787*, 2023.

[34] Zhenhui Ye, Ziyue Jiang, Yi Ren, Jinglin Liu, JinZheng He, and Zhou Zhao. Geneface: Generalized and high-fidelity audio-driven 3d talking face synthesis, 2023.

[35] Zhenhui Ye, Tianyun Zhong, Yi Ren, Ziyue Jiang, Jiawei Huang, Rongjie Huang, Jinglin Liu, Jinzheng He, Chen Zhang, Zehan Wang, Xize Chen, Xiang Yin, and Zhou Zhao. Mimictalk: Mimicking a personalized and expressive 3d talking face in minutes, 2024.