11th ISCA Speech Synthesis Workshop (SSW 11)
26-28 August 2021, Budapest, Hungary

Lipsyncing Efforts for Transcreating Lecture Videos in Indian Languages

Mano Ranjith Kumar M, Jom Kuriakose*, Karthik Pandia D. S, Hema A. Murthy*

Department of Computer Science and Engineering, Indian Institute of Technology, Madras

manoranjith@smail.iitm.ac.in, jom@cse.iitm.ac.in,
karthik.pandia@gmail.com, hema@cse.iitm.ac.in

Abstract

This paper proposes a novel lip-syncing module for the transcreation of lecture videos from English to Indian languages. The audio from the lecture is transcribed using automatic speech recognition. The text is translated and manually curated before and after translation to avoid mistakes. The curated text is synthesized using the Indian language end-to-end-based text-to-speech synthesis systems. The synthesized audio and video are out-of-sync. This paper attempts to automate this process of producing video lectures lip-synced into Indian languages using different techniques.

Lip-syncing an educational video with the Indian language audio is challenging owing to (a) the duration of Indian language audio being considerably longer or shorter than that of the original audio, (b) the extempore speech causes the audio in the source videos to have long silences. Any modification to the speed of audio can be unpleasant to listeners. The proposed system non-uniformly re-samples the video to ensure better lip-syncing. The novelty of this paper is in the use of HMM-GMM alignments in tandem with syllable segmentation using group delay, as visemes are closer to syllables. The proposed lip-syncing techniques are evaluated using subjective evaluation methods. Results indicate that accurate alignment at the syllable level is crucial for lip-syncing.

Index Terms: Automatic dubbing, Lip-syncing, Transcreation, Group delay.

1. Introduction

The medium of instruction for higher education globally is predominantly English. Educational content that is available online has been growing exponentially in recent times. Language is the major barrier to use these resources in places like India, where people use almost 1652 different languages and 22 official languages to communicate. Recent advancements in speech recognition (ASR), machine translation (MT), and speech synthesis (TTS) have enabled us to develop systems that automatically dub educational videos in Indian languages. This paper proposes and analyses various lip-syncing methods that improve the quality of transcreation of lecture videos to Indian languages.

Automatic dubbing is an extension of speech-to-speech translation [1]. It involves (i) transcription of speech from the video, (ii) translation of the transcribed text into the target language, (iii) synthesizing target language audio, and (iv) lip-syncing synthesized audio with the original video. In automatic dubbing, generally, the machine translation is done carefully, such that the number of syllables in the source and target audio is almost matched. For example, dubbing of movies preserves the duration of the video and forces the audio to align within the duration of the video. This is attained by manually curating the translated text to match the video duration and lip movements in such a way that the lip-synced video does not sound unnatural. The lip-syncing techniques proposed in this paper does not force the synthesised audio length to match the video length. Instead, the video is re-sampled to match the audio duration. In lecture videos, conveying the underlying concepts is prioritized more than matching the syllable rate, so that the conveyed information is not lost in the process. Changing the speech rate to match the duration of the video is also not preferred since our analysis showed that the lip-synced videos with variable speech rate are not preferred for by the viewers. Adding to this, unlike English, Indian languages are word order free, and hence the length of the translated text is generally longer or shorter than that of source English text. Due to this, the synthesized audio is significantly longer or shorter than that of the source audio. Hence, we prefer transcreation of video to target language over simple dubbing by preserving the information conveyed in the source video lectures.

The previous works in lip-syncing focused on different techniques for aligning source video and target audio or text. In recent work by [1, 2], TED talks are automatically dubbed using a prosodic alignment module that aligns the speech segments from source audio with the machine-translated text. The attention mechanism is used in [3] to find a plausible phrasing for the translated text, which is synthesized and added to the source video. Other than finding alignments, changing the speed of synthetic speech to fit into the subtitle duration is attempted in [4, 5]. An end-to-end audio-visual translation system is trained on thousands of hours of data from all domains in [6], and adapted to a specific domain and speaker. Many works in the literature, including [7, 8] suggest that there is a co-relation between visual speech unit (visemes) and phonetic speech unit (phonemes). Syllables are combination of phonemes. Using syllable level boundaries for lip-syncing makes sure that the visemes are not spliced in the video.

This is the first attempt in the literature to transcreate educational lectures from English to Indian languages. Lip-syncing video with the Indian language audio is challenging due to the longer duration of the translated, and synthesised audio. The domain of educational videos also make it more challenging, as the original speaker in the lecture videos have long silences when the lecturer is dis-fluent. The transcreated video's naturalness depends not only on exact alignments of synthesized audio with lip movements but also on the long pauses, head and hand movements, and expressions of the speaker in the original video. Hence in our lip-syncing method, these attributes of the source video are preserved in the final video.

The paper proposes lip-syncing methods using word-level alignments on the synthesized audio. A group delay (GD) based syllable segmentation is used to fine-tune the word boundaries, which further improves the naturalness by preserving the viseme units is proposed. The lip-syncing developed in this pa-

*Equal contribution by both authors.

10.21437/SSW.2021-38

per attempts to attain isochrony [9], where audio is synchronized with the speaker's lip movements. These systems are evaluated using subjective evaluation method; mean opinion score (MOS) to show that word-level alignments are very important for better lip-syncing output.

The rest of the paper is organized as follows. In Section 2, the video transcreation system is discussed. Different lip-syncing systems proposed along with preliminary lip-syncing systems are detailed in Section 3. Section 4 explains the evaluation setup for these systems, followed by results and discussions in Section 5 and Section 6 concludes the paper.

2. Pipeline of Video Transcreation System

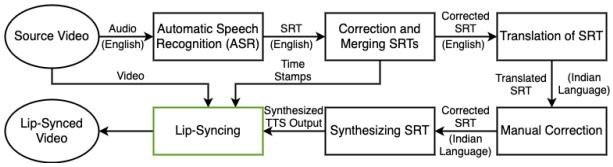

Figure 1: *Flowchart of video transcreation system. Lip-syncing module attempted in this paper is shown in green color.*

The flowchart for video transcreation is given in Figure 1. The input lecture video is split into segments by detecting long silence regions using speech activity detector of Kaldi toolkit [10], and these segments are transcribed using the ASR system[1] [11] to create the Sub-Rip Text (SRT) file. Technical lectures contain words that are domain-specific and includes a significant amount of mathematical equation and notations. These are preserved, manually verified and corrected before translating to the target language, as shown in the flowchart (Figure 1). The machine translation to Indian languages is done using language translation APIs [12, 13]. The translated text after manual verification is synthesized using the end-to-end (E2E) [11, 14] based TTS system proposed in [15]. The TTS models are trained on Indic TTS [16] data-set. The synthesized audio along with the source video and subtitle file, is given as input to the lip-syncing system. The lip-syncing system is described in detail in Section 3.

3. Lip-Syncing System

The lip-syncing techniques discussed in this paper segment the TTS synthesized audio and aligned them with the source video by re-sampling the video. In the following sub-sections, we discuss two simple preliminary systems based on re-sampling, silence detection, and two proposed systems using ASR alignments obtained from TTS synthesized audio and group delay-based segmented ASR alignments.

3.1. Preliminary Systems

As an initial attempt at lip-syncing, we did a simple re-sampling of the video segments. The trailing and beginning silences of the synthesized audio are discarded before re-sampling. The source video is then re-sampled to match the audio segments' duration by interpolating the video to match the duration of synthesized audio. The flow chart of this method is highlighted in red color in Figure 2. This attempt is a basic approach to lip-syncing without using any silence alignments or word boundaries. This method does not detect long-pause regions in the

[1] https://asr.iitm.ac.in/NPTEL/Transcribe/

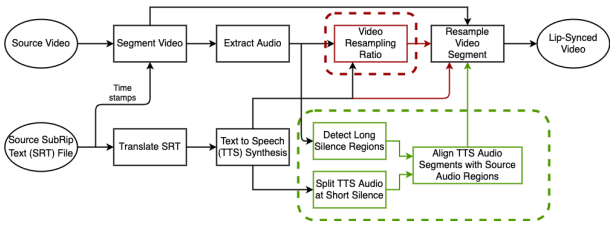

Figure 2: *Flow chart of preliminary methods attempted. Pipeline for re-sampling based lip-syncing is shown in red color and pipeline for silence detection based lip-syncing is shown in green color.*

source video within a SRT segment. Hence, in the transcreated video, there are sections where the lecturer's lips are not moving while synthesized audio is playing. The final video also has sections where the video speeds up or slows down very aggressively due to the large difference between source audio and target audio duration. This is mainly due to the duration differences between sentences in English and Indian languages and the long pauses in source video within the SRT segment.

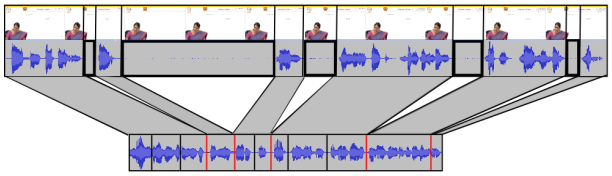

Figure 3: *Alignment of audio after silence detection. Box with black border shows long-silence regions in source video. Red line denotes location of spliting in TTS audio. The alignment between video and audio is also shown.*

To address this issue, transcreated videos need to ensure that the audio plays only when the lips are moving. The lip movements of the speaker should synchronize with the regions of speech in a source audio segment. Therefore in the next attempt, we try to detect the silence regions and keep it as it is and align the synthesized audio with the regions where the speaker is speaking as shown in Figure 3. The flowchart of this system is given in Figure 2, where the green box highlights the flow of this attempt. We try to detect the long silence regions in the audio from the source video, and short pauses in the TTS synthesized audio and try to align them using Algorithm 1. The algorithm tries to map the TTS segments with the segments of the source audio. This is done by finding the silence regions in the source audio and splitting the TTS audio in the same ratio as in the source audio. This process of splitting and aligning the TTS audio is initially done for the longest silence region in the source audio, followed by the second-longest silence, and so on recursively until the TTS audio is completely mapped with the source audio. Refer Algorithm 1 for more details. The same algorithm is used in Proposed System 1 and Proposed System 2 for aligning the source video with the TTS audio.

Silence detection works well in finding the silence regions. However, since the silence detection only look for the silence regions, the silence detection will often detect short silence regions in the middle of certain words that are often occur together. This can lead to the system splicing the synthesized audio in the middle of those words and inserting silences. Even

Algorithm 1 Mapping of source intra-SRT-segment alignments to TTS intra-SRT-segment alignments

Input: *src_segs(1:N):- List of source intra-SRT-segments of length N obtained after silence detection of a source audio segment*

tts_segs(1:M):- List of TTS intra-SRT-segments of length M after silence detection after TTS audio segment (Obtained from word boundaries of ASR incase of Proposed System 1 and Proposed System 2)

Format *of src_segs(1:N) & tts_segs(1:M):-*
<start-time> <end-time> <duration> <label>
where (label==0) => silence & (label==1) => speech

Output: *map(,):- List of mappings between src_segs indices & tts_segs indices*

procedure ALIGN_SEGMENTS(*src_segs(1:N),tts_segs(1:M)*)
▷ *This function returns alignment of source segements and target segments*

variables
map(a,b):- represents that index a in src_segs is aligned with index b in tts_segs.
end variables

if *length(src_segs)==1* ***OR*** *length(tts_segs)==1* ***then***
▷ *This is the base condition for recursion*
return
else
src_split = SOURCE_SPLIT_INDEX(*src_segs(1:N)*)
speech_ratio =
SPEECH_RATIO(*src_segs(1:N),src_split*)
tts_split =
TTS_SPLIT_INDEX(*tts_segs(1:M),speech_ratio*)
return ALIGN_SEGMENTS(*src_segs(1:src_split), tts_segs(1:tts_split)*) + ***map***(*src_split,tts_split*)
+ ALIGN_SEGMENTS(*src_segs(src_split:N),tts_segs(tts_split:M)*)

though the lip will sync with the video, the synthesized audio being played will lose its understand-ability considerably in this case. We address this issue of splicing the audio between words in Proposed System 1 using word-level alignments.

3.2. Proposed System 1: Word level alignment method

Silence boundaries do not have information of the target text and hence can have intra-word splits. Word level alignments are required to avoid splitting between words. Word level boundaries are obtained using hidden Markov model - Gaussian mixture model (HMM-GMM) based ASR system [10]. For training the ASR systems, a common label set (CLS) [17] lexicon representation of the transcription is obtained using the unified parser [18] for Indian languages. The word-level alignments obtained are used to align the synthesized audio at the word level with the source video. The word-level alignments using HMM-GMM ASR are only obtained for the synthesized audio since silence regions are better-detected using signal processing techniques using short-term-energy (STE) for source audio. The detection of silence in the source is very important for better lip-syncing. The flowchart of Proposed System 1 is highlighted in red in

procedure SOURCE_SPLIT_INDEX(*src_segs(P:Q)*)
▷ *This function returns a index of largest silence region in source segments*

variables
sil_dur_list:- list of duration of silence segments in src_segs.
end variables

sil_dur_list = (src_segs(P:Q)(label==0).duration)
max_dur = max(sil_dur_list)
return *max_dur*

procedure SPEECH_RATIO(*src_segs(P:Q),src_split*)
▷ *This function returns ratio of speech on both sides of silence region*

variables
speech_dur:- duration of speech before split.
full_dur:- duration of speech in whole src_segs.
end variables

speech_dur = $\sum$ *(src_segs(P:src_split)(label==1).duration)*
full_dur = $\sum$ *(src_segs(P:Q)(label==1).duration)*
return *speech_dur/full_dur*

procedure TTS_SPLIT_INDEX(*tts_segs(P:Q),speech_ratio*)
▷ *This function returns index of split in TTS segments,which is close to speech ratio*

variables
X:- list of indices of tts_segs(P:Q) segments.
tts_ratio(x):- TTS speech ratio at split index x.
end variables

for x `in` X ***do***
tts_ratio(x) =
$\sum$ *(tts_segs(P:x)(label==1).duration)* /
$\sum$ *(tts_segs(P:Q)(label==0).duration)*
return *Index x for min of (abs(speech_ratio − tts_ratio(X)))*

Figure 4. An example for ASR alignment obtained is given in Figure 5.

The system detects the words correctly, but in some cases, the beginning or ending of the word can be misaligned by few milliseconds. While this is not an issue in ASR results, the misalignment can be due to the word ending not being an ending of a syllable. These boundaries may result in clicking noises, which can sound unnatural. Proposed System 2 tries to correct these boundaries to obtain better word-level boundaries by using group delay and energy-based correction.

3.3. Proposed System 2: Word alignments correction using group delay (GD) & spectral flux (SF)

Minimum phase group delay (GD) of short-term-energy (STE) and sub-band spectral flux (SBSF) together are used for identifying syllable boundaries. The algorithm for finding sylla-

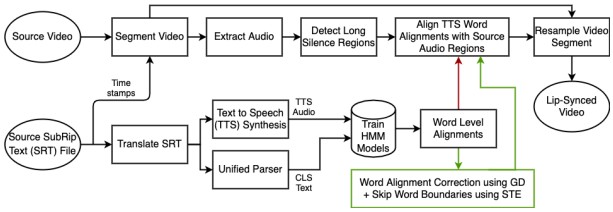

Figure 4: *Flowchart of proposed lip-syncing methods. Pipeline for Proposed System 1 is shown in red color and pipeline for Proposed System 2 is shown in green color.*

ble boundaries using GD and SBSF is explained in [19]. The word boundaries obtained from the HMM-GMM ASR model is corrected using the syllable boundaries obtained from GD and SBSF. After alignment correction, the short-term energy (STE) is calculated at these boundaries, and high energy boundaries are excluded from alignments. Similar to Proposed System 1, the silence detection algorithm is used to find the long silence regions inside the source audio segments, and corrected word boundaries of synthesized audio are aligned with source audio segments using the Algorithm 1.

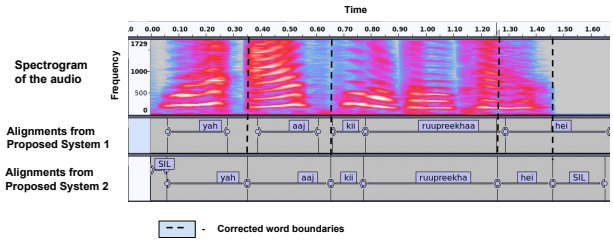

Figure 5: *Word level alignment of Hindi utterance "Yah aaj kii ruupreekhaa hei" (English: This is today's outline) obtained using Proposed System 1 and Proposed System 2.*

During the speech, some words are articulated together (Ex. metro network, deep learning). Splitting the synthesized audio in between those words can lead to clicking sounds. This is primarily owing to the co-articulation between adjacent syllables that make up a word(s). In Indian languages, it is very common to find that the word morpheme at the end of one word is merged with the succeeding word. Clearly, more than one syllable or syllables across word boundaries is perhaps associated with a viseme. We attempt to correct this by using word boundaries and syllable boundaries, where the short-term energy is very low, suggesting that the co-articulation is small.

4. Evaluation

In addition to the systems discussed in Section 3, we compare our systems with an additional system that transcreates video using Google cloud API[2]. This system uses ASR, MT, and TTS modules from Google cloud APIs, to transcreate videos. This system will be henceforth referred to as Baseline System. Unlike the systems discussed above, in the baseline system, the TTS audio is generated based on the duration of the video segment. By controlling the speaking rate in the TTS module, the duration of audio is matched with the source video.

Since it was evident that proposed systems are better than preliminary attempts, we evaluated only Proposed System 1 and 2 against the Baseline System. These three systems were evaluated using, mean opinion scores (MOS) test. The configuration of the test is given in the following subsection. Samples of final trans-created videos for all preliminary, proposed, and baseline systems are given in this link: https://www.iitm.ac.in/donlab/preview/lip_sync_21/index.html

[2]https://github.com/google/making_with_ml/tree/master/ai_dubs

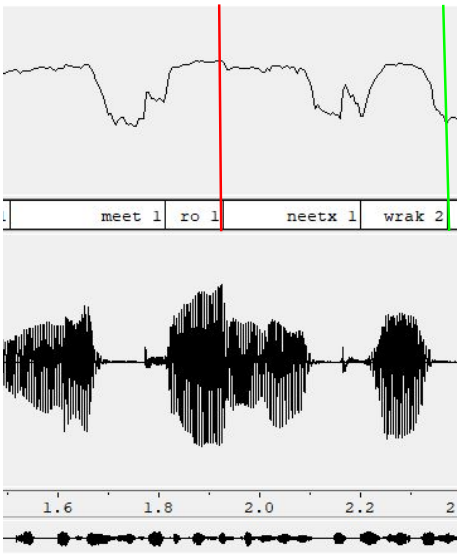

Figure 6: *The synthesized waveform of "metro network" and it's corresponding syllable boundaries (Metro ("meet","ro"), Network ("neetx","wrak")) along with STE plot. The red line corresponds to the word boundary where the energy is high and the green line corresponds to the word boundary, where the energy is low.*

4.1. MOS Test

The mean opinion scores (MOS) test is originally designed to evaluate TTSes. A set of 15 evaluators were asked to evaluate 5 lip-synced videos of approximately one and half minutes to two and half minutes, in comparison with the corresponding source video. Two male and two female speaker video segments are chosen for evaluation from Proposed System 1 and Proposed System 2, along with a Baseline video segment. The video segments are chosen at random from the whole lecture. The speakers are not repeated in the 5 randomly chosen video segments to make sure that the viewer does not get used to the speaker. The evaluators are asked to rate the naturalness of the video, including the lip movement. The videos are played at random to the evaluators. The evaluators are asked to give a rating from 0 to 100 for each video segment, where 0 being no synchronization between the audio and the video, and 100 being perfectly lip-synced.

5. Results and Discussion

As seen in Table 1, the Proposed System 2 has highest MOS score of *82.55*, whereas Proposed System 1 has a MOS score of *75.64*. The lip-synced videos using Proposed System 2 are rated higher due to the correction of word boundaries using group delay-based segmentation, along with the splitting of synthesised audio at low energy word boundaries. The importance of correcting the word boundary using group delay-based segmen-

Table 1: *Mean opinion score (MOS) of Hindi language lip-syncing for Proposed System 1 and Proposed System 2 along with Baseline System.*

Systems	MOS Score
Proposed System 1: Word-level alignment method	75.64
Proposed System 2: Word alignments correction using group delay (GD) spectral flux (SF) correction	**82.55**
Baseline System: Google Cloud API based application: ML-Powered Translation	62.98

tation is shown by an example in Figure 5. The figure shows spectrogram of Hindi utterance "*yah aaj kii ruupreekhaa hei*" (English: "*This is today's outline*") in common label set (CLS) format defined in [17]. Word boundaries of the word "*aaj*" and "*ruupreekha*" are more accurate in Proposed System 2 after boundary correction. Silence regions are also recognised with very high precision after correction, compared to the ASR boundaries obtained using Proposed System 1 as shown in Figure 5.

The word boundaries don't always correspond to the decay of the speech signal but can also be an onset for the next word. This can be due to the co-articulation of two words together. This can be seen in Figure 6, where the words "*metro network*" are pronounced together. The STE value at the end of the word "*metro*" (refer red line) is higher than that of the STE value at the end of the word "*network*" (shown in green). Thus, splitting in-between "*metro network*" is avoided. Splitting the signal between these words will lead to unnatural artifacts like clicking sounds. Using STE to find these word boundaries has also contributed to the higher performance of Proposed System 2 over Proposed System 1.

The Baseline system has a lower MOS score of *62.98* compared to Proposed System 1 and 2. The Baseline system increases or decreases the rate of speech based on the length of the video segment to which it has to be merged and hence has a varied rate of speech throughout the lip-synced video. This can also be due to the fact that the sentences in Indian languages are generally longer in comparison to those in English. The variable speech rate can be unpleasant to the viewers. In terms of constant speech rate, the Proposed System 1 can be considered as a baseline, and results show that the Proposed System 2 provides an improvement.

Signal processing techniques like group delay and STE can be used in tandem with the machine learning methods (ASR) to find accurate word-level boundaries. The results show that the improvement in word-level alignments has significantly improved the quality of the lip-synced video.

6. Conclusions

Professional dubbing is an expensive and labour intensive process. This work proposes novel techniques to improve the naturalness of auto-transcreated videos. The discussed results shows that syllable level segmentation (Proposed System 2) provides an absolute improvement of *6.9* over simple ASR word level alignment based technique (Proposed System 1). Using visual speech units (visemes) along side syllable segmentation may further improve the observed results for lip-syncing.

7. Acknowledgements

We want to extend our gratitude to Speech Lab, Indian Institute of Technology, Madras (IITM) for their help in generating the SRT files for the lecture videos. We would like to thank the Department of Science and Technology (DST), the Ministry of Electronics and Information Technology (MeitY), Office of the Principal Scientific Adviser (PSA) to the Government of India, for funding the projects, "Text to Speech Generation with chosen accent and noise profile for Aerospace and Industrial domains" (CSE1819172MIMPHEMA), "Natural Language Translation Mission" (CS2021012MEIT003119), "Speech to Speech Machine Translation" (CS2021152OPSA003119), respectively.

8. References

[1] M. Federico, R. Enyedi, R. Barra-Chicote, R. Giri, U. Isik, A. Krishnaswamy, and H. Sawaf, "From speech-to-speech translation to automatic dubbing," *arXiv preprint arXiv:2001.06785*, 2020.

[2] M. Federico, Y. Virkar, R. Enyedi, and R. Barra-Chicote, "Evaluating and Optimizing Prosodic Alignment for Automatic Dubbing," in *Proc. Interspeech 2020*, 2020, pp. 1481–1485. [Online]. Available: http://dx.doi.org/10.21437/Interspeech.2020-2983

[3] A. Öktem, M. Farrús, and A. Bonafonte, "Prosodic phrase alignment for machine dubbing," *arXiv preprint arXiv:1908.07226*, 2019.

[4] J. Matoušek and J. Vít, "Improving automatic dubbing with subtitle timing optimisation using video cut detection," in *2012 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2012, pp. 2385–2388.

[5] "How to dub a video with ai youtube," https://youtu.be/T2TAAHmNBnE, 02 2021, (undefined 4/5/2021 0:57).

[6] Y. Yang, B. Shillingford, Y. Assael, M. Wang, W. Liu, Y. Chen, Y. Zhang, E. Sezener, L. C. Cobo, M. Denil *et al.*, "Large-scale multilingual audio visual dubbing," *arXiv preprint arXiv:2011.03530*, 2020.

[7] S. Taylor, "Discovering dynamic visemes," Ph.D. dissertation, University of East Anglia, 2013.

[8] H. L. Bear and R. Harvey, "Phoneme-to-viseme mappings: the good, the bad, and the ugly," *Speech Communication*, vol. 95, pp. 40–67, 2017.

[9] F. C. Varela, "Synchronization in dubbing," *Topics in audiovisual translation*, vol. 56, p. 35, 2004.

[10] D. Povey, A. Ghoshal, G. Boulianne, L. Burget, O. Glembek, N. Goel, M. Hannemann, P. Motlíček, Y. Qian, P. Schwarz, J. Silovský, G. Stemmer, and K. Vesel, "The kaldi speech recognition toolkit," *IEEE 2011 Workshop on Automatic Speech Recognition and Understanding*, 01 2011.

[11] S. Watanabe, T. Hori, S. Karita, T. Hayashi, J. Nishitoba, Y. Unno, N. E. Y. Soplin, J. Heymann, M. Wiesner, N. Chen, A. Renduchintala, and T. Ochiai, "Espnet: End-to-end speech processing toolkit," *CoRR*, vol. abs/1804.00015, 2018. [Online]. Available: http://arxiv.org/abs/1804.00015

[12] Y. Wu, M. Schuster, Z. Chen, Q. V. Le, M. Norouzi, W. Macherey, M. Krikun, Y. Cao, Q. Gao, K. Macherey *et al.*, "Google's neural machine translation system: Bridging the gap between human and machine translation," *arXiv preprint arXiv:1609.08144*, 2016.

[13] V. Mujadia and D. Sharma, "NMT based similar language translation for Hindi - Marathi," in *Proceedings of the Fifth Conference on Machine Translation*. Online: Association for Computational Linguistics, Nov. 2020, pp. 414–417. [Online]. Available: https://www.aclweb.org/anthology/2020.wmt-1.48

[14] R. Prenger, R. Valle, and B. Catanzaro, "Waveglow: A flow-based generative network for speech synthesis," in *ICASSP 2019-2019 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2019, pp. 3617–3621.

[15] A. Prakash, A. L. Thomas, S. Umesh, and H. A. Murthy, "Building multilingual end-to-end speech synthesisers for indian languages," in *Proc. of 10th ISCA Speech Synthesis Workshop (SSW'10)*, 2019, pp. 194–199.

[16] A. Baby, A. L. Thomas, N. Nishanthi, T. Consortium *et al.*, "Resources for indian languages," in *Proceedings of Text, Speech and Dialogue*, 2016.

[17] B. Ramani, S. Lilly Christina, G. Anushiya Rachel, V. Sherlin Solomi, M. K. Nandwana, A. Prakash, S. Aswin Shanmugam, R. Krishnan, S. Kishore, K. Samudravijaya, P. Vijayalakshmi, T. Nagarajan, and H. A. Murthy, "A common attribute based unified HTS framework for speech synthesis in Indian languages," in *Speech Synthesis Workshop (SSW)*, 2013, pp. 291–296.

[18] A. Baby, N. Nishanthi, A. L. Thomas, and H. A. Murthy, "A unified parser for developing indian language text to speech synthesizers," in *International Conference on Text, Speech, and Dialogue*. Springer, 2016, pp. 514–521.

[19] S. A. Shanmugam, "A hybrid approach to segmentation of speech using signal processing cues and hidden markov models," Ph.D. dissertation, MS Thesis, Department of Computer Science Engineering, IIT Madras, India, 2015.