CSCI 1430 Final Project Report:
Speaker Identification and Vocal Separation using the Discrete Wavelet Transform

Team name: Parrotfish
TA name: John Farrell.
Brown University
Interactive Demo GitHub Model Weights

Abstract

We propose an approach combining audio source separation targeting timbral differences with visual speaker identification, where both methods employ the Discrete Wavelet Transform.

1. Introduction

Audio source separation and specifically speech separation has strong potential for use in making media more accessible to hearing-impaired individuals, especially in conjunction with visual speaker identification. This is especially applicable in the context of modern media, where podcasts have become incredibly common as a content form intended both for education and entertainment. This comes as just one example of the movement towards decentralizing media away from big companies, and while this is in many ways a good thing, it also means that the content creators of today often may not have the budget and capacity - and regulatory mandates - to ensure that accessibility standards are being met, meaning that certain demographics may receive less consideration in content production.

By creating a speech separation model that specifically focuses on timbre in conjunction with a visual speech identification algorithm, we aim to bridge that gap, reducing the overhead for independent media producers to implement accessibility features such as speaker-identified captioning even in cases where an audio contains simultaneous speaking.

2. Related Work

1. **Wave-U-Net** [6]: In 2018 Stoller et al. proposed the "Wave-U-Net," which has since become the most influential audio separation model of all time. At a high level, their approach is based on the Short-Time Fourier Transform (STFT), which is a time-frequency representation of audio in which overlapping segments of the waveform have their Fourier transforms arranged in a time-series. This allows for a great insight into not only which frequencies are present within an audio clip, but when they occur. This, coupled with a CNN-based U-Net architecture allows the model to learn the timbres of each instrument and segment the spectrogram based on each of the learned instrument fingerprints. Although this model is influential, STFT-based methods have a glaring issue: they typically ignore phase information. In other words, the Wave-U-Net only learns STFT magnitudes and assumes the phases are shared between the mixture audio and each source, which is fundamentally incorrect. Although this dramatically simplifies the approach, interaction between sources will almost always an effect on phase, so even if the model perfectly learns to segment the magnitude spectrum, the resulting reconstructed audio will never sound perfectly separated.
2. **Multiresolution Deep Layered Analysis (MRDLA)** [3]: T. Nakamura and H. Saruwatari attempted to rectify this fundamental issue with an STFT-based approach by switching to a different time-frequency representation of audio: wavelets. Wavelets are small packets of frequency information localized in time at different scales, which makes them inherently good at capturing both high frequency detail and low frequency localization. Unlike the STFT, phases are baked-in to wavelets, which allows for a fully end-to-end separation model. Furthermore, certain types of wavelets have anti-aliasing filters baked in, and an extension of MRDLA allowed for learnable wavelets. This approach achieves better results than the Wave-U-Net with fewer parameters.
3. **Streaming Multi Speaker ASR with RNN-T** [5]: Amazon researchers Skylar et al. developed a DPNN-style architecture for text-to-speech with multiple simultaneous speakers. Their approach relies on using a

recurrent neural network transducer (RNN-T) and dual-path architecture to achieve multi-speaker automatic speech recognition (ASR) that works in real-time and is extremely lightweight. This model is present on the Alexa smart device. The key distinction in training is the use of Permutation Invariant Training [8], which allows the model to learn on the best permutation of the predicted speakers. Since it is impractical to explicitly label two out of potentially thousands of speakers, the model must learn to distinguish between speakers without having an expected order. The issue with PIT is that is it has factorial complexity scaling with the number of speakers, as you must check every permutation of true and predicted speaker order. Furthermore this model only works for two speakers.

4. **Directed speech separation for ASR of long-form conversational speech** [4]: Paturi, et al., another team of Amazon researchers, takes a different approach to multi-speaker separation for ASR, as they correctly point out that a major issue to most separation pipelines is that to limit model size, training clips are limited to one-two seconds. This requires a "stitching step" at the end. For instrument separation, this is not an issue, as model predicted vocals, drums, bass, etc. will always appear in the same indices. However, with PIT and speaker separation, this is not the case. As a consequence, this means there is no guarantee that the relative order of speakers will be preserved across audio frames, which can hamper the model's effectiveness when dealing with dynamic speaker changes across long-form audio (we also encountered this problem as our approach focused on one second audio). Paturi et al. used transformers to get more long-form context and avoid the need for PIT. The abstract for this paper essentially gives a point-for-point outline of the limitations of our model, and unfortunately we did not see this paper until after we already decided with stick with a segmented CNN approach.

3. Method

The goal for our model was to take in an input video and return both the identified speakers and the frames in which they are speaking as well as their individual separated audios. This means that we are essentially solving two problems: speaker identification, which is a computer vision problem; and speech separation, which is an audio processing problem. The latter is astronomically more difficult.

Preprocessing and the Data Pipeline: For the audio model, we wanted to have end-to-end audio separation. In other words, the model would take in audio and return audio.

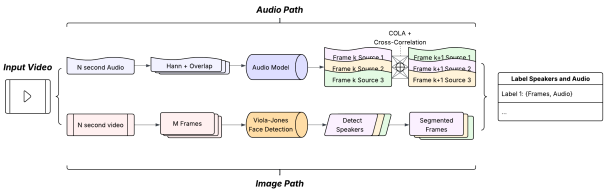

Figure 1. Control Flow of AV paths

To do this, we started with two datasets from Meta: Casual Commands, which has several thousands of one-minute video recordings of single speakers in multiple languages (mainly Portuguese) and the Fair-Speech Dataset, which includes around 50k 2-8s segments of people issuing voice commands in English. We used pydub to extract the audio from casual commands and chopped up all the audio in both datasets into windowed 1s segments of waveform audio resampled to 16kHz. We then hashed each segment into 500 directories for better randomization and efficiency. We then compressed each folder with tf.records and uploaded them to Google Cloud Storage for fast loading into Google Colaboratory. We used tf.dataset to dynamically batch these 1s segments into (1-3) speaker mixtures. Starting with approximately 500k 1s isolated segments, we could synthesize near-limitless data, so the model never saw the same data twice. We trained the model on an Nvidia A100 in Colab with a batch size of 16 and callbacks for early-stopping, intermediate saving to Drive, and reducing the learning rate. We then retrained the model with a lower initial LR and a batch size of 32, which we replicated from [3] and [3].

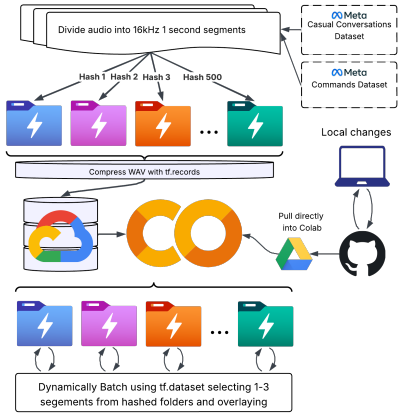

Figure 2. Diagram of data pipeline to Colab

Separation: We adapted our wavelet instrument separation model from CSCI 1470 (Spring 2024) which was inspired by MRDLA [3]. This did require some fairly substantial modifications, as we had to convert the loss function and dataset structure to work with PIT and we modified the skip connections to allow for gating. Furthermore, we use a

different US and DS block structure where we have different processing paths for the approximation and detail wavelet coefficients. The reason we chose wavelets over STFT-based methods follows from section 2, as not only do wavelets contain phase information and have built-in anti aliasing filters, but the structure of the Discrete Wavelet Transform (DWT) and its inverse (IDWT) mirrors that of the more traditional U-Net convolution and transposed convolution. That is that the DWT takes in a sequence of shape (features, channels) and returns one of (features // 2, channels$\times$2) where the first channel corresponds to the low frequency information (detail coefficients) and the second corresponds to the high frequencies (detail coefficients). As mentioned in 4, we need an additional stitching step for the model to work with longer audio. The stitching step involved using Hann windows with 25% overlap to eliminate edge effects then using the method of Constant-Overlap Add (COLA) to perfectly reconstruct a signal after the model has been run on each segment. However, as there is no guarantee that speaker order is preserved between segments, we use the Hungarian algorithm to match successive segments based on the similarities between their overlapping sections.

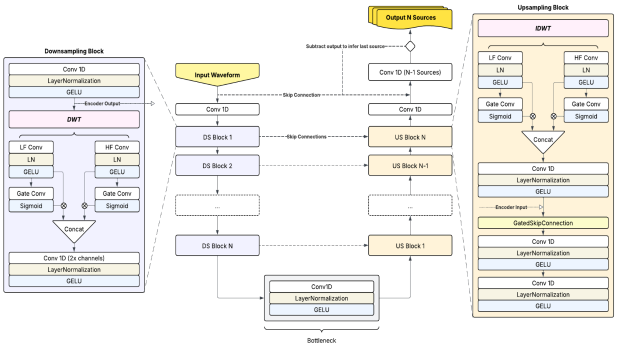

Figure 3. Wavelet-U-Net model architecture.

Speaker Identification: In our visual speaker identification system, we implemented facial movement tracking using pretrained OpenCV Haar cascade classifiers [1] based on the Viola-Jones [7] detection framework. The algorithm identifies regions of interest (ROIs) for faces in each video frame, then locates mouth regions within the facial boundaries. After collecting ROI data for the entire video sequence, we analyze each frame by comparing it with a window of three frames before and after. This comparison uses multiple metrics between corresponding ROIs: contour area changes, centroid movement (derived from image moments), and shape variations (calculated using Hu moments). We determined thresholds for each metric to detect significant mouth movements indicative of speech, and optimized them based on qualitative evaluation. When all metrics exceed their thresholds, the system identifies and records the corresponding face as an active speaker in the analyzed frame.

4. Results

Overall the initial results were extremely promising, especially on one second audio segments. For longer clips, the drawbacks to stitching as mentioned in 4 hamper the quality of the audio model. We firmly believe that wavelets are the superior choice to STFT approaches for speech separation, as they capture the timbral differences between speakers over time without the phase limitations of the STFT. Furthermore, DWT layers can be implemented efficiently as Finite Impulse Response (FIR) filters with a lifting scheme [2], which integrate well with tensorflow's support for efficient convolution.

```
approx = tf.nn.conv1d(
             channel_inputs,
             self.dec_lo_filter,
             stride=2,
             padding='VALID')
```

We used the Pywavelets package and the Debauchies 4 tap (db4) wavelet for pre-computed filter coefficients. We chose this wavelet because it has a good frequency response for speech [2].

Although using Viola-Jones through openCV was extremely efficient, the image model struggled to correctly identify speakers at off-angles and when the faces were far away. We suggest using a more sophisticated approach for this task in the future. Despite being primarily trained on

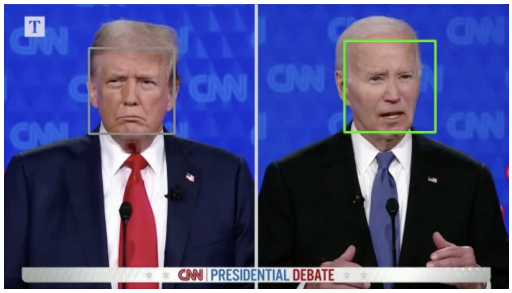

Figure 4. Viola-Jones speaker identification during the 2020 Presidential Debate

Portuguese audio, the speech model does a surprisingly good job generalizing to English speech, but can struggle if the relative differences in volume are too great between speakers. We also might have trained too much on single speaker (identity) data, as the model tends to do a great job isolating the loudest speaker but has a quite a bit of bleed in channels for quieter speakers. However, the model does a good job at predicting silence in the other channels when there are one and two speakers. The model also does equally well on masculine and feminine voices, which we attribute to our data being composed of majority feminine speakers.

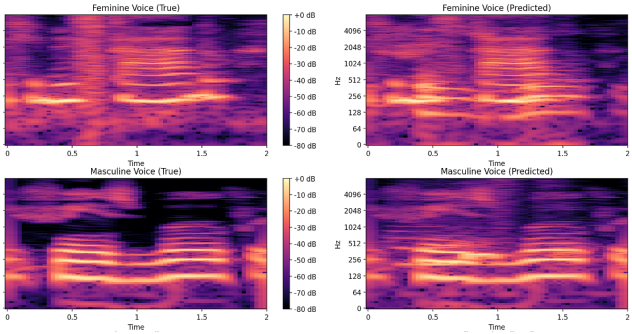

Figure 5. Mel-Spectrograms of Separated Audio Segments *Left:* True spectrograms. *Right:* Predicted Spectrograms

Parameters	Validation Loss (PIT MSE)
127M (residual)	3.37×10^{-4}
127M (one-pass)	3.51×10^{-4}
17M (one-pass)	3.95×10^{-4}

Table 1. asdfasdffasdadwasd FIX THIS

4.1. Technical Discussion

The model's performance suggests that wavelets are great for human speech separation, but a one-second segment approach fundamentally holds back the models generalization to longer segments. Even using the Hungarian algorithm, the channels would often switch between whoever is speaking the loudest.

The final size of our main model was around 127M parameters, but we experimented with a smaller version that only required 17M (a similar size to MRDLA and Wave-U-Net [6]) and achieved slightly poorer results. However, it still performed remarkable well given the nearly $8\times$ reduction in model complexity.

We also experimented with a residual output layer that predicts the first two sources and infers the third as the residual between the original input and the two predicted sources. This enforces that the output sums to the original signal, but over-training on single speakers led the model to overweight the contribution of the residual output even in cases with multiple speakers. Forgoing this in favor of a single-pass approach did a better job at distinguishing speakers at different volumes, but did not sum to the original signal.

5. Conclusion

Our work demonstrates the effectiveness of using the Discrete Wavelet Transform for both audio source separation and visual speaker identification. The wavelet-based approach provides distinct advantages over traditional STFT-based methods by preserving phase information and incorporating anti-aliasing filters, resulting in more natural-sounding separated audio. While our model achieves promising results on short audio segments, the inherent limitations of segmented processing highlight the need for transformer-based architectures to better handle long-form content and speaker transitions. Despite these challenges, our experiments confirm that wavelets are well-suited for capturing the timbral differences between speakers. Future work should focus on improving the speaker identification component beyond Viola-Jones, addressing the channel-switching issues in longer audio segments, and exploring alternative wavelet families to further enhance separation quality. This research demonstrates meaningful progress toward making media more accessible to hearing-impaired individuals through automated speaker-identified captioning, even for content with simultaneous speech.

6. SRC

As we began working on our project, we made sure to center the SRC implications of our idea at every step, ensuring that we didn't turn a blind eye to other important ethical considerations in the name of promoting accessibility. As such, the critiques that we received from our peers in our SRC report had a significant influence over the direction of our project.

1. **Overrepresentation of English speakers**

   First and foremost, we were careful to ensure that English speakers were not overrepresented in our audio data. In doing so, we actually found that we had significantly more Portuguese audio clips in our dataset than English, in addition to a wide variety of other languages including Tagalog, Spanish, Indonesian, Hindi, and many more, including a significant share of clips highlighting individuals speaking in their non-native languages from many regions of the world.

2. **Data & Privacy**

   Furthermore, in response to the second critique regarding the privacy of individuals, we made the decision to move away from our initial deep learning approach to speaker recognition and instead use an approach based purely on feature detection using Haar cascades and the Viola-Jones algorithm. By doing this, we were able to significantly reduce the risk of potential privacy issues that come with storing massive amounts of data containing people's faces and voices together, especially with the limitation of using externally hosted data storage. We additionally chose to change the direction of the image processing aspect of our project, in part to allow for this. We initially planned to create a facial recognition model to be used to count unique faces that appear throughout each frame within a section of a video, in order to input into the model to determine

how many output files the model should produce. However, we were able to avoid the necessity of the face and video data for this task as we were able to build an architecture for our audio model that allowed us to separate into an arbitrary number of stems (our model had a maximum of 3 as limited by our resources, but the architecture is extensible). As a result, the only data being stored beyond inputs to the trained model (which are not stored anywhere permanently, and can be confined to the local environment of the user) is the audio data, consisting of a series of thousands of 1-second clips completely isolated from their context which are already publicly available via Meta. Finally, we used a Google Cloud Storage to store all of our data, a production-grade cloud storage service that ensures the security of data.

3. **Surveillance & Misuse**

   We recognize the potential of our project to be used in surveillance applications and certainly would not condone this use of our technology. With that said, the most likely misuse of this concept that is discussed in our critique in a surveillance application would be to isolate an audio source within an extremely crowded environment that would not be audible to the human ear. Our project does not focus on isolating an imperceptible audio source from background noise, and thus would not be capable of achieving this; in its current state, our model is trained to isolate up to 3 overlapping speakers that are all at least mostly audible for the purpose of furthering accessibility, especially in conjunction with other technology, such as a speech-to-text model for speaker-isolated captioning.

References

[1] Modesto Castrillón-Santana, L. Antón-Canalís O. Déniz-Suárez, and J. Lorenzo-Navarro. Face and facial feature detection evaluation. In *Third International Conference on Computer Vision Theory and Applications, VISAPP08*, January 2008. 3

[2] Tomohiko Nakamura, Shihori Kozuka, and Hiroshi Saruwatari. Time-domain audio source separation with neural networks based on multiresolution analysis. *IEEE ACM Trans. Audio Speech Lang. Process.*, 29:1687–1701, 2021. 3

[3] Tomohiko Nakamura and Hiroshi Saruwatari. Time-domain audio source separation based on wave-u-net combined with discrete wavelet transform. *CoRR*, abs/2001.10190, 2020. 1, 2

[4] Rohit Paturi, Sundararajan Srinivasan, Katrin Kirchhoff, and Daniel Garcia-Romero. Directed speech separation for automatic speech recognition of long-form conversational speech. 2022. 2

[5] Ilya Sklyar, Anna Piunova, and Yulan Liu. Streaming multi-speaker asr with rnn-t, 2021. 1

[6] Daniel Stoller, Sebastian Ewert, and Simon Dixon. Wave-u-net: A multi-scale neural network for end-to-end audio source separation. *CoRR*, abs/1806.03185, 2018. 1, 4

[7] P. Viola and M. Jones. Rapid object detection using a boosted cascade of simple features. In *Proceedings of the 2001 IEEE Computer Society Conference on Computer Vision and Pattern Recognition. CVPR 2001*, volume 1, pages I–I, 2001. 3

[8] Dong Yu, Morten Kolbæk, Zheng-Hua Tan, and Jesper Jensen. Permutation invariant training of deep models for speaker-independent multi-talker speech separation. *CoRR*, abs/1607.00325, 2016. 2

Appendix

Team contributions

Matthew McQuistion Matt was responsible for adapting the wavelet-based audio model from CSCI 1470 to work with the new task of speech separation. This required fundamental changes to the network architecture along with implementing permutation invariant training. Architecture changes involved both adding gated skip connections and modifications to the wavelets. Matt switched the model to compute the DWT/IDWT directly with the builtin tensorflow convolution. PIT changes included a substantially more involved loss function and changes to the output layer (both residual and one-pass) along with developing the smaller version of the model. Matt made the poster along with the visualizations for network architecture, control flow, and pipeline. Matt also wrote the logic for the stitching step with the Hungarian algorithm, extending the model to work with longer segments of audio with windowing and COLA. Lastly, Matt wrote the logic for model retaining, allowing us to load in a model and retrain using a different config file.

Rayhan Meghji (Capstone) Rayhan was responsible for the image model implementation with Viola-Jones and wrote the logic for full end-to-end video to segmented video + sources logic and the gradio interface through Huggingface. This required migrating the codebase and model to work with Huggingface spaces and GitHub LFS. In addition to shared responsibilities with model training and data processing, Rayhan revamped the model loading and saving scheme, which allowed for much cleaner loading and iteration. They also handled the compression/decompression to tf.records, dynamic batching logic, and the cloud data pipeline which proved challenging to get to work with XLA. Rayhan also handled several critical reworks of the project structure and synchronizing GitHub to work with google Colab through Dive.