Real-Time Facial Detection Talkbox: Integrating Facial Landmark Detection with Vocal Formant Simulation

ZhengHao Wang
Peabody Institute of Johns Hopkins
zwang369@jh.edu

Ted Moore
Peabody Institute of Johns Hopkins
tmoore97@jh.edu

ABSTRACT

The Real-Time Facial Detection Talkbox (RFDT) is a multimodal system that integrates real-time facial landmark detection with vocal formant simulation, enabling talkbox-like control of a real-time audio signal. Using Google's MediaPipe Face Landmark technology, RFDT tracks the user's mouth landmarks and maps mouths shapes to vocal formants which are simulated using filters in SuperCollider. Communication between Python and SuperCollider is facilitated via OSC (Open Sound Control), ensuring seamless real-time interaction.

Author Keywords

formants, real-time processing, computer vision, digital music instrument

Figure 1: A guitar player (Kaifeng "Kafoona" Huang) using the *Real-Time Facial Detection Talkbox* (RFDT) in real time while playing guitar.

1. INTRODUCTION

This paper introduces the *Real-Time Facial Detection Talkbox* (RFDT)**,** a multimodal system that integrates facial landmark detection with vocal formant simulation to enable talkbox-like control of a real-time audio signal. RFDT combines two core technologies to create a novel interaction between physical gestures and synthesized sound: (1) MediaPipe to track facial landmarks in real-time and extract the user's mouth shape, and (2) a filter-based simulation of vocal formants using SuperCollider. [3] MediaPipe (running in Python) [5] communicates facial landmarks to SuperCollider via OSC (Open Sound Control), [8] enabling dynamic real-time interaction between the tracking system and the sound synthesis engine. Future research will focus on enhancing the recognition accuracy, exploring additional applications for the interface, improving the realism of filter-based vocal formant simulation, and extending the system to simulate consonants.

There are many previous research projects that simulate human voice synthesis, including synthesizers that simulate the entire vocal tract [1] and replicate the characteristics of human speech [2], provide a detailed 3D vocal tract model for simulating speech production by adjusting anatomical

parameters, [10] combine biomechanics and acoustics to model and simulate the dynamics of vocal tract muscles in real time while integrating neural networks with physical models to simulate the neural and anatomical processes behind speech generation. [11-13] RFDT is unique from this previous research its use of real-time interaction through facial tracking. This direct interaction provides an immediate and expressive way for the performers to manipulate formant filters, emulating the intuitive musical approach of pre-existing talkbox technology.

2. DATA PROCESSING PIPELINE

Creating the system pipeline involved extracting facial landmarks, transmitting them to SuperCollider, and visualizing the data in real time. These processes are performed by the system and are therefore invisible to the user. This section describes how the system processes and transmits these signals.

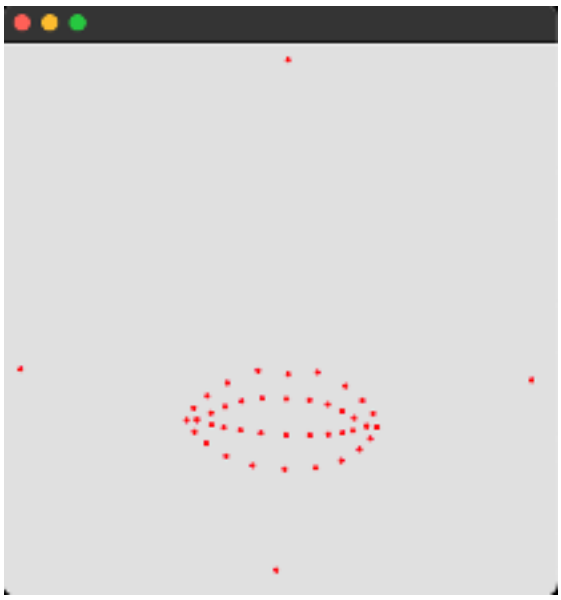

Figure 2: Two-dimensional visualization of signals sent by MediaPipe to SuperCollider, representing tracked facial landmarks used for vowel synthesis in RFDT.

2.1 MediaPipe Facial Recognition in Python

The *Real-Time Facial Detection Talkbox* (RFDT) employs MediaPipe Face Landmark in Python to detect and track 468 facial landmarks from webcam input, focusing on a predefined subset around the mouth. The x and y coordinates of these key points are extracted and sent as OSC (Open Sound Control) messages to SuperCollider using the *pythonosc* library. [14]. Once received in SuperCollider, the OSC messages are parsed to extract the coordinates, which are normalized to a 0-to-1 range to ensure consistent scaling across devices. These normalized landmarks are then visualized in real time using a custom *UserView* object in SuperCollider, where each point is displayed as a red dot, forming a dynamic representation of the mouth's shape. This real-time visualization enables performers to intuitively monitor how their mouth movements are translated into data for filter control.

3. VOCAL FORMANTS AND FILTER SIMULATION

Vocal formants are the resonant frequencies that result from the interaction between the human vocal cords and the vocal tract. When a person speaks or sings, the vocal cords generate a fundamental frequency, which are then shaped by the vocal tract to create specific sounds. The vocal tract then amplifies certain frequencies while attenuating others, and the selective amplification this produces is how human vowel sounds are formed. [9]

The two most important formants for vowel distinction are **F1** and **F2**:

F1 is influenced by the openness of the vocal tract. For instance, an open mouth, as in the vowel **/a/**, corresponds to a higher F1 frequency, while a more closed mouth, as in **/i/**, results in a lower F1 frequency.

F2 is related to the front-back positioning of the tongue. A front tongue position, as in **/i/**, leads to a higher F2, while a back tongue position, as in **/u/**, produces a lower F2. [9]

Each vowel has a unique combination of F1 and F2 frequencies based on the position of the lips and tongue. For example:

/a/: High F1 (~800 Hz), Low F2 (~1200 Hz)

/i/: Low F1 (~300 Hz), High F2 (~2500 Hz)

/u/: Low F1 (~300 Hz), Low F2 (~900 Hz)

These formants are the acoustic signatures of vowels. Higher-order formants (F3, F4, etc.) also contribute to the timbre of the voice but are less critical for vowel identification. [9]

3.1 Simulating Vocal Formants

Simulating vocal formants is crucial for recreating Talkbox-like vowel sounds. By modulating the frequencies of F1 and F2, it is possible to mimic the acoustic characteristics of vowels. To replicate the Talkbox-like sounds, RFDT uses band-pass filters [7] in SuperCollider, which selectively amplify specific frequency ranges while attenuating others. This process emulates the resonance of the vocal tract. Key aspects of the filter design include:

1. Mapping Mouth Shapes to Formants**:**

F1 is derived from the vertical distance between the upper and lower lips. A larger distance corresponds to a higher F1, while a smaller distance results in a lower F1.

In the vocal tract **F2** is determined by tongue position, however, MediaPipe's Facial Landmark tracking cannot determine tongue positions. Therefore, we've chosen as a proxy to calculate the horizontal distance between the corners of the mouth which perceptually relates to F2. A wider mouth corresponds to a higher F2, and a narrower mouth results in a lower F2.

These distances are normalized using the `(linlin)` object to map onto the typical frequency ranges of F1 (300–1000 Hz) and F2 (800–4000 Hz).

2. **Real-Time Interaction**:

The OSC messages sent from Python continuously update the F1 and F2 values, allowing the performer's mouth movements to directly influence the generated sound. This creates a seamless interaction between physical gestures and simulated vowels.

FUTURE RESEARCH

The system currently supports vowel classification of the five primary vowels (A, E, I, O, U), but these classifications are not yet utilized in the filtering process. In future studies, these vowels may be integrated into the filtering mechanism to simulate the effects of a talkbox more accurately. Future research aims to enhance the realism of filter-based vocal formant simulations, expand the capability to simulate consonants, and develop a synthesizer that can model the entire vocal tract, including both vowels and consonants, for more realistic speech synthesis. Additionally, real-time consonant recognition functionality will be explored.

REFERENCES

[1] IMAGINARY. Pink Trombone. Web: https://github.com/IMAGINARY/pink-trombone, 2024.

[2] H. Ishizuka, T. Saito, and Y. Arai. IEEE 802.15.6 channel model. Web: **https://ieeexplore.ieee.org/document/6289140**, 2012.

[3] McCartney, J. (2002). Rethinking the computer music language: Super collider. *Computer Music Journal*, *26*(4), 61-68.

[4] Google. MediaPipe Face Landmarker. Web: https://ai.google.dev/edge/mediapipe/solutions/vision/face_landmarker, 2024.

[5] Python, Why. "Python." *Python releases for windows* 24 (2021).

[6] Python OSC - Open Sound Control server and client implementations in pure Python. https://pypi.org/project/python-osc

[7] Hosken, Dan. *An introduction to music technology*. Routledge, 2014.

[8] Freed, Adrian. "Open sound control: A new protocol for communicating with sound synthesizers." In *International Computer Music Conference (ICMC)*. 1997.

[9] Peter Ladefoged, Richard Harshman, Louis Goldstein, Lloyd Rice; Generating vocal tract shapes from formant frequencies. *J. Acoust. Soc. Am.* 1 October 1978; 64 (4): 1027–1035. https://doi.org/10.1121/1.382086

[10] P. Birkholz, "VocalTractLab: A 3D model of the vocal tract for speech synthesis and research." https://vocaltractlab.de, 2024.

[11] A. Lau and S. Fels, "ArtiSynth: A biomechanical modeling toolkit for the vocal tract," University of British Columbia. https://www.artisynth.org, 2024.

[12] N. M. Stephens, P. Ahmed, and J. Keller, "TorchDIVA: A hybrid neural and physical model for speech synthesis." PLOS ONE, vol. 18, no. 1, 2024. https://journals.plos.org.

[13] Y. Zhang, X. Li, and A. Sharma, "Material Point Method Vocal Tract Simulation: Modeling the biomechanics and aerodynamics of speech production," Proc. Int. Phonetic Assoc., 2023. https://www.internationalphoneticassociation.org.

[14] Attwad, 2024 python-osc (V1.9.0) https://github.com/attwad/python-osc. URL