Opti-Speech-VMT: Implementation and Evaluation

Hiranya G. Kumar[1(✉)], Anthony R. Lawn[1,2], B. Prabhakaran[1], and William F. Katz[1]

[1] University of Texas at Dallas, Richardson, TX 75080, USA
{hiranya,bprabhakaran,wkatz}@utdallas.edu
[2] Topaz Labs, 14285 Midway Road, Suite 125, Addison, TX 75001, USA

Abstract. We describe Opti-Speech-VMT, a prototype tongue tracking system that uses electromagnetic articulography to permit visual feedback during oral movements.Opti-Speech-VMT is specialized for visuomotor tracking (VMT) experiments in which participants follow an oscillating virtual target in the oral cavity using a tongue sensor. The algorithms for linear, curved, and custom trajectories are outlined, and new functionality is briefly presented. Because latency can potentially affect accuracy in VMT tasks, we examined system latency at both the API and total framework levels. Using a video camera, we compared the movement of a sensor (placed on an experimenter's finger) against an oscillating target displayed on a computer monitor. The average total latency was 87.3 ms, with 69.8 ms attributable to the API, and 17.4 ms to Opti-Speech-VMT. These results indicate minimal reduction in performance due to Opti-Speech-VMT, and suggest the importance of the EMA hardware and signal processing optimizations used.

Keywords: Speech · Tongue · Visual feedback · Electromagnetic articulography · Avatar · 3D model · Latency

1 Introduction

In previous work, we described Opti-Speech, a technique for animating a 3D model of the human tongue in real time [1]. The system was designed for research and training in second language learning and for clinical applications in speech language pathology. We used motion capture data from an electromagnetic articulography (EMA) system and an off-the-shelf 3D animation software (Maya) to create the visual representation. The goal of the Opti-Speech project was to create a real-time tongue representation with the necessary resolution for designating tongue shapes and positions common to a variety of speech sounds. In our application, "joint" positions and rotations were used to drive a hierarchical rig of virtual joints that in turn deformed a geometric mesh of a virtual tongue. Based on practical considerations (number of EMA sensors that can be comfortably placed on the lingual surface) and prior research on the number of

M. Ur Rehman and A. Zoha (Eds.): BODYNETS 2021, LNICST 420, pp. 233–246, 2022.
https://doi.org/10.1007/978-3-030-95593-9_19

EMA sensors that can effectively describe speech sounds [2,3], we determined that five markers could provide the necessary resolution for identifying tongue shapes and positions common to a variety of speech sounds. We created a flexible rig of joints in Autodesk Maya to allow the markers to drive a polygonal tongue that was then brought into Autodesk MotionBuilder software. Through a custom plugin for MotionBuilder, the motion data from our EMA system is streamed in real-time and constrained to the marker setup of the rig. The resulting movement of the tongue mesh allows the subject to watch a 3D model of their own tongue movements in real time (Fig. 1).

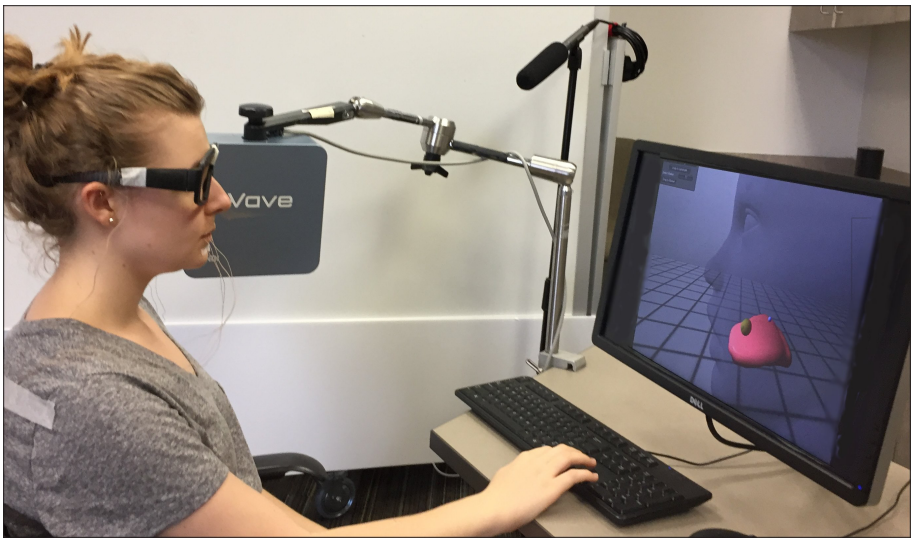

Fig. 1. A participant using the Opti-Speech system with NDI WAVE hardware.

Several studies were conducted using the first prototype system, including training [4–6], and visuomotor tracking [7,8] paradigms. These studies revealed the limitations of our first prototype Opti-Speech system, including an inability to introduce moving targets and to present different trajectories for these moving targets. These additional features are useful for studies of speech motor control, including comparisons of speech and non-speech movements. In addition, these features will help in clinical studies that present more sophisticated moving target patterns for patients to emulate.

In this paper, we introduce Opti-Speech-VMT (Opti-Speech for Visual Motor Tracking experiments) providing: (i) a variety of static and moving targets, (ii) different trajectories for the moving target, including curved and custom trajectories. Since the latency of the system could have effects on users in speech and tracking experiments, we investigated the Opti-Speech-VMT and total system latency periods. The results indicated that Opti-Speech-VMT contributed

minimal latency (17.464 ms) and that total system delay was substantially larger (87.319 ms). Most importantly, these latency periods fall below the range reported to have potential adverse effects on speech performance.

2 Related Work

Although the effects of visual feedback on speech have been extensively reviewed in several studies [9–18], few studies have focused on the framework used to obtain the feedback. Most of the studies on effects of visual feedback on speech learning have relied on ultrasound imaging [9–12]. Ultrasound (US) imaging allows a participant to directly visualize the interior of the oral cavity as a two-dimensional image. It comes with a few advantages: the output from an US system doesn't need processing to be used as visual feedback for the participant, the detection system itself is non-intrusive, and the equipment is portable. In addition, there have been recent developments in tongue tracking and segmentation in US images. Laporte et al. (2018) [13] developed an US tongue tracking system that uses simple tongue shape and motion models with a flexible contour representation to estimate the shape of the tongue. Karimi et al. (2019) [14] discuss an algorithm requiring no training or manual intervention which uses image enhancement, skeletonization and clustering to come up with a set of candidate points that can then be used to fit an active contour to the image, subsequently initializing a tracking algorithm. Mozaffari et al. (2020) [15] make use of Deep Learning techniques (Encoder-decoder CNN models) for automatic tracking of tongue contours in real-time in US images.

A disadvantage of US feedback systems is that the imaging provided is typically low-resolution, monochrome, and noisy, and therefore not very intuitive. In addition, the visual feedback cannot be customized by the user to add functionality such as interactive feedback, adding targets, manipulating input data, etc., which limits usability for different types of speech experiments.

EMA-based systems have also been used for visual feedback studies [16–18], although to lesser extent than US-based systems, due to their lack of portability and high costs. Shtern et al. (2012) [16] and Tilsen et al. (2015) [17] describe EMA real-time feedback systems for articulatory training. Real-time kinematic data from the EMA system is used to create an interactive 3D game for speech learning/rehabilitation, based on the Unity Game Engine.

Suemitsu et al. (2015) [18] make use of EMA hardware for real-time visual feedback to support learning the pronunciation of a second language. Specifically, they studied the production of an unfamiliar English vowel, (/æ/), by five native Japanese speakers. An array of EMA sensors was used to obtain each participant's tongue and lip fleshpoint positions, and an image of the tongue surface was estimated using cubit spline interpolation. Participants compared their head-corrected, near real-time data with an/æ/target obtained from a multiple linear regression model based on previous X-ray microbeam data and additional EMA data.

The project that most directly relates to ours is a visual feedback framework (Kristy et al. [19]) based on Blender, a free and open-source software for 3D

development. This framework is able to visualise and record data from a EMA system or from a file. It uses NDI WAVE hardware and performs data-processing steps (including head correction, smoothing, and transformation to local coordinate system) before generating the visual feedback. A Python program is used to fetch and process data using the NDI Wave API.

Although the EMA-based game systems mentioned above do provide the user with visual feedback of their tongue movements, they have a near-static environment with a minimal feature set. As such, these systems lack several important features, including an ability to add interactive targets, easily control the visual elements on the screen, vary the sensor placement, and conduct visuomotor tracking experiments.

3 Opti-Speech-VMT

Visuomotor tracking (VMT) tasks involve a participant following a rhythmic external signal with a limb or speech articulator (typically the lip/jaw). It is common to designate targets of different speeds and levels of predictability to assess the role of feedforward/feedback processing in motor control. In addition, the direction of movement may be of interest [5,6].

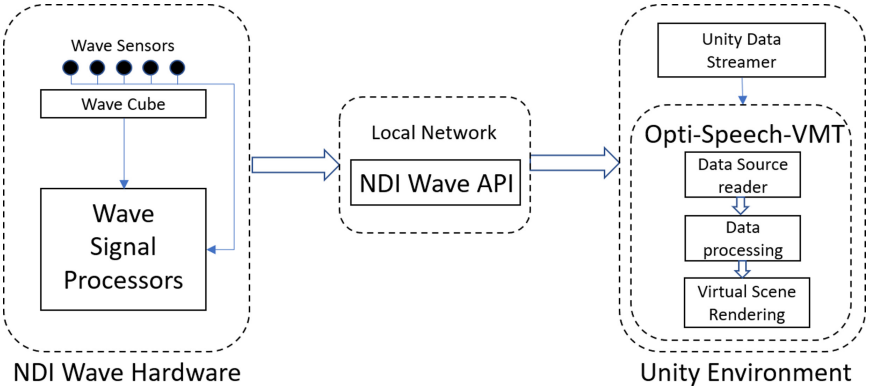

Fig. 2. A flowchart describing the workflow of the Opti-Speech-VMT system used with NDI Wave hardware.

3.1 General Features

Opti-Speech-VMT is built using the open source software Unity (version 2020.1.12f1) in Windows 10, and is compatible with the latest version of Unity and Windows (as of this date). Due to the application being built in Unity, which is a cross-platform tool, it can be migrated to other Unity supported platforms (such as Linux, MacOS, etc.) with minimal effort. Figure 3 shows the GUI of

Opti-Speech-VMT. All the menus are collapsible, to allow a clean user interface. Some menus have been expanded in the figure to show the available options for the user.

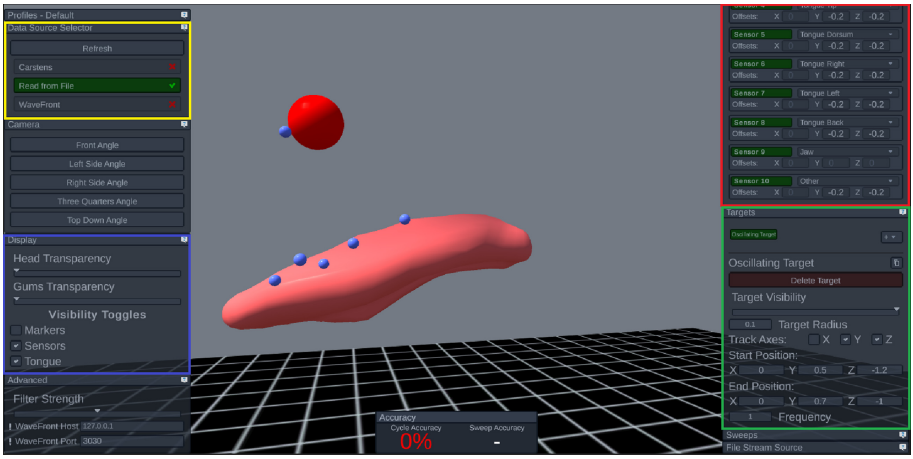

Fig. 3. A screenshot of the Opti-Speech-VMT GUI showing the tongue avatar with fully transparent skull and jaw models.

Various features offered by Opti-Speech-VMT are:

1. **Data source selector**: Opti-Speech-VMT supports multiple data sources simultaneously. The Data source selector menu (highlighted in a yellow box in Fig. 3) allows the user to select from multiple data sources that could be connected to the system. Currently supported sources are:
   (a) File reader: Allows the user to play back a recorded sweep from a file
   (b) WaveFront: Allows user to stream data from NDI WAVE hardware.
   (c) Carstens (under development): Allows user to stream data from Carstens AG500 series articulography systems.
   Additional data sources can be added to the application by following the developer manual available with the project.
2. **Display menu**: The display menu (highlighted in a blue box in Fig. 3) allows the user to hide, show, and control the transparency of various elements on the screen, such as the tongue model, skull, jaws, markers, and sensors. Figure 4 shows an example with transparency for the skull and jaws set to around 50%. Users can set the values according to the participant's preferences or the experiment's requirements.
3. **Sensor List**: The Sensors List menu (highlighted in a red box in Fig. 3) allows the user to map the software sensor markers to physical sensors connected to the system. This mapping can be changed in real-time, which saves the user time and effort by not needing to ensure that the physical sensors are placed in a particular order. The menu also allows the user to set sensor-specific

offsets in case adjustments are needed to the positions of the virtual sensors (without requiring the user to adjust the physical sensors).

4. **Targets**: The targets menu (highlighted in a green box in Fig. 3) allows the user to add multiple targets to the scene simultaneously. The targets can be of different types, each having their own sub-menus containing modifiable parameters specific to the target type. Additional targets with custom trajectories can also be added to the application using the guide provided in the Developers manual.

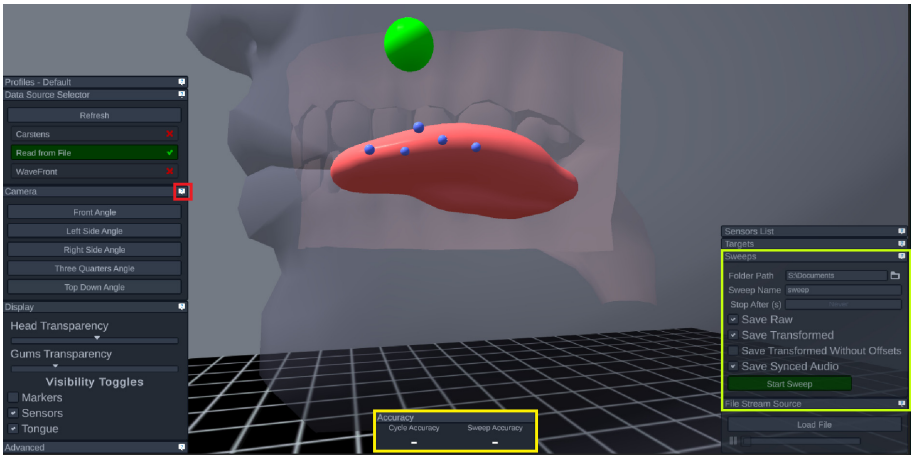

Fig. 4. A screenshot of the Opti-Speech-VMT GUI showing the tongue avatar with translucent skull and jaw models.

5. **Sweeps**: This menu (highlighted in a lime green box in Fig. 4) allows the user to record sweeps of an experiment. The sweeps record the data of all the objects present in the scene per timestamp. This includes status and positions of all the sensors, and the position and parameters of the targets and markers and other data needed by the application to replay the sweep (using the File Reader Data Source in Data Selector menu). Since the timestamp has a resolution of 1 millisecond, the data recorded are high resolution and allow for a precise replay of the sweep. The data are saved in a .tsv (tab-separated values) format, which is easy to read with any text editor or Microsoft Excel. Audio data synchronized with the timestamps can also be saved using options available in the menu.
6. **Tongue model**: The tongue model used in Opti-Speech-VMT is not rigid. This allows the shape and size of the tongue model to automatically change based on the positions of the physical sensors on the participant's tongue. While the model is not intended to be biomechanically/anatomically realistic for medical (e.g., surgical reconstruction planning) purposes, it closely represents tongue surface dimensions and movements sufficient for real-time applications.

7. **Accuracy Window**: The accuracy window (highlighted in a yellow box in Fig. 4) shows the cycle and sweep accuracies of the tongue movement with respect to a given target. This can be used as a visible metric for the participant to understand their performance on a per-cycle and per-sweep level.
8. **Other menus**: Apart from the above-mentioned menus, other menus are mainly designed for convenience. This includes the **Profiles menu** (above the Data Source Selector menu), which allows the users to save experiment settings in a profile and quickly load them to effortlessly repeat an experiment, the **Camera menu**, which has preset camera angles that help the user change the camera view to any of the available ones with just one click, and the **Advanced menu**, which allows a user to specify network settings used to communicate with connected EMA hardware and also specify Filter Strength to smooth out the raw sensor data incoming from the API. This can help smooth out the 'jittering' of virtual sensors that can result from the hardware being highly sensitive.
9. **Documentation**: Detailed documentation of the application is also available for user reference and further development of the application. The documentation is split into the "Researcher manual", meant for users of the application, and the "Developer manual", meant for developers who wish to modify or extend the functionalities of the application. The white speech bubbles (highlighted in a red box in Fig. 4) at the top right corner of each of the menus directly takes the user to the section of the documentation describing the usage of that specific menu. An example of the documentation for the Data Source selector menu is shown in Fig. 5.

3.2 Target Features

Static/Moving Targets: As in the original Opti-Speech prototype, Opti-Speech-VMT provides a static target for tongue tracking by allowing the operator to designate a virtual sphere in the oral cavity that a participant "hits" using a selected sensor on their tongue avatar. When the target is hit, it changes color, providing the participant with visual feedback of accuracy. Opti-Speech-VMT includes a "moving target" option that programs a single target to oscillate between two positions in the oral cavity. The direction, distance traversed, speed, and predictability of the oscillating target motion are all set by the operator during a session. The speed is controlled by a "frequency" option available in the menu. An option to add "randomness" to the target speed, which makes the speed variable and unpredictable between oscillations, is also available in the menu.

Trajectories: Target trajectories can be linear, curved, or "custom". This allows the experimenter to set tongue targets in linear oscillating patterns that are traditionally reported in VMT studies, as well as curved patterns that are more speech-like, as previous studies have reported curved (arc-like) patterns taken by the tongue in reaching spatial endpoints [20]. The custom trajectory

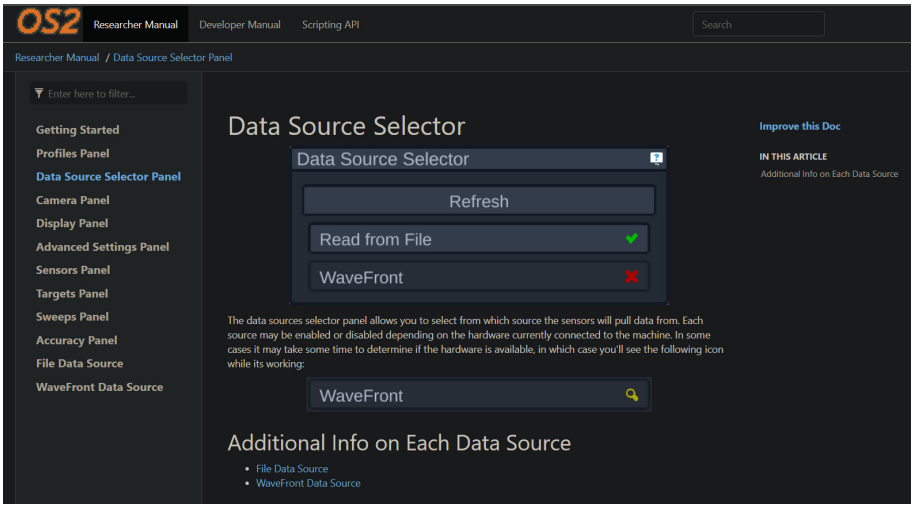

Fig. 5. An example of the documentation provided with Opti-Speech-VMT.

option can be used to create a variety of movement patterns, such as may be required in different studies of motor control.

Linear trajectories are based on a sinusoidal function, as described in Eq. 1 and Eq. 2. We use Eq. 1 with a Linear Interpolation (Lerp) function (Eq. 2) and user-provided start and end positions to get the position of the target at any given time (currTime). Employing a sinusoidal function instead of a standard linear function for linear oscillating motion allows us to generate a trajectory that slows down the target near the start and end positions, instead of abruptly reversing the direction of motion. This makes the trajectory closer to natural tongue motion.

$$f(freq, time) = \frac{Cos(2 * \pi * freq * time/1000) + 1}{2} \tag{1}$$

$$current\ position = Lerp(startPosition, endPosition, f(freq, time)) \tag{2}$$

Where,

- freq = frequency of oscillation (oscillations/second).
- nRebounds = number of times the target has changed directions till now.
- startPosition = User defined start position of the target.
- currPosition = current position of the target at the given time.
- pauseTime = User defined time to pause after reaching start/end position (in ms).
- time = Current timestamp from the application

Curved trajectories are implemented using an ellipse equation, as shown in (Eq. 3). The trajectory is currently restricted to the YZ plane only. Apart from the frequency of oscillation, this equation also needs the major and minor axis of the ellipse as user inputs. To emulate a more natural tongue movement, we add a small user-defined pause towards the end and start positions of the trajectory. The adding of a pause time makes the algorithm significantly more complicated.

$$currPosition = (c.x, c.y + r.y * Sin(angle), c.z + r.z * Cos(angle)) \quad (3)$$

$$if\ tempAngle >= 90,\ angle = tempAngle,\ else\ angle = 180 - tempAngle \quad (4)$$

$$tempAngle = (freq * 180 * localTime/1000)\ mod\ 180 \quad (5)$$

$$localTime = time - nRebounds * pauseTime \quad (6)$$

$$c = startPosition - r \quad (7)$$

Where,

- c = 3D coordinates of the center of the ellipse.
- r = User defined 3D vector with the radius of the ellipse along each axis.
- freq = frequency of oscillation (oscillations/second).
- nRebounds = number of time the target has changed directions till now.
- startPosition = User defined start position of the target.
- currPosition = current position of the target at the given time.
- pauseTime = User defined time to pause after reaching start/end position (in ms).
- time = Current timestamp from the application

The curved trajectories are currently limited to the YZ plane (Y axis being vertical and Z axis being horizontal with respect to the oral cavity) for simplicity and since most common 2D tongue motions are restricted to the YZ (mid-sagittal) plane.

Custom motion trajectories replicate a prerecorded tongue movement. Tongue movement can be recorded using the 'Record Sweep' functionality offered by the application. The menu allows the Target to replicate the movements of any of the sensors recorded in the sweep. This can be used to have the patient emulate pronunciation of a particular syllable/word. All the trajectories that cannot be described by a simple mathematical equation can be recorded by the user in a sweep and replicated by this function for experiments and/or training.

4 Measurement of Latency

For a visual feedback system, latency is an important metric. A significant lag between the tongue's motion and the visual feedback on the screen can have significant effects on the subject's experience and response as well as on the experimental results [21–23].

For instance, studies of the effects of visual and/or auditory feedback on pointing and steering tasks (Friston et al. [22]), sequence reproduction on a keyboard (Kulpa et al. [21]), and sentence repetition (Chesters et al. [23]) report a broad range at which latencies affect motor behavior, ranging from 16–400 ms. Friston et al. [22] also note that measurement of latencies below 50 ms, in tasks that involve indirect physical interaction, has only recently become possible due to advancements in technology. Regarding tongue visual feedback, Suemitsu et al. [18] describe "no perceptually apparent latency between sensor motion and its visualization" for their system based on a Carstens AG500 EMA system. While perceptual benchmarks are important, it would also be useful to have measurements of the latency of visual feedback frameworks to help further studies in this domain. We therefore designed an experiment to measure the latency of the Opti-Speech framework at different levels.

4.1 Visual Feedback Task

The experiment makes use of a camera to monitor the real sensor and the computer screen (60 Hz refresh rate) showing the API and Opti-Speech-VMT sensors. The camera records a video of the sensor moving in an oscillating trajectory in the plane of observation of the camera. The perspectives of the camera in the rendering software (API and Opti-Speech-VMT) are adjusted to match that of the camera recording the experiment, such that they have the same plane of observation. A tripod-mounted phone camera (Samsung Galaxy S20FE) was used in 1920×1080 60 FPS video recording mode, with all other video settings set to auto. Video was recorded in a well-lit environment to facilitate color-based tracking.

We directly measured two types of latency periods: the latency at the API level (API Latency), which is the time taken by the EMA hardware to process raw signals from the sensors into sensor positions, and latency at the framework level (total latency), which is the total time taken for a change in real sensor position to reflect in Opti-Speech-VMT's visual tongue avatar. We use these latency values to compute Opti-Speech-VMT latency (Eq. 8), which is the time taken by Opti-Speech-VMT to get the sensor positions from the API and render the scene with the tongue avatar.

$$OptiSpeechVMT\ latency = total\ latency - API\ latency \tag{8}$$

As we cannot directly compare the raw displacement values between the sensor, the API, and Opti-Speech-VMT, since the sensors are not calibrated, we use an oscillating trajectory, normalize the amplitudes of the trajectories, and plot them (Fig. 6). Next, we measure the latency at points in the trajectory where there is a change in direction (peaks and valleys of the oscillating trajectory). Since the oscillating movement of the real sensor is done by hand, the trajectories aren't perfectly sinusoidal. We process the video recording in Python frame-by-frame and use OpenCV [24] to track the trajectories of the sensors based on color.

Since the monitor displaying the API and Opti-Speech-VMT sensors and the camera have a refresh rate 60 Hz, the lowest latency we can measure is

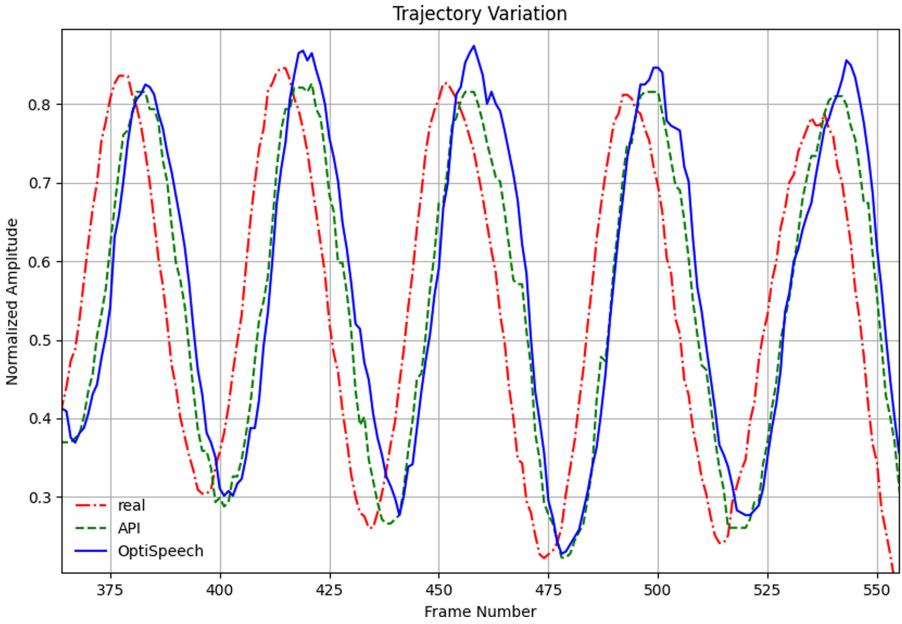

Fig. 6. Normalized amplitude of the trajectory of the Real sensor, API sensor and Opti-Speech-VMT sensor plotted against the frame number. The graph has been magnified to focus on a few periods of the oscillation to better show the temporal differences between the trajectories. (Color figure online)

1/60 s, which is 16.67 ms. The latency is directly measured in frames, and then converted to milliseconds based on the frame time (16.67 ms) (Eq. 9). We average the latency measurements from 42 samples taken at the peaks and valleys of the trajectories.

$$latency\ in\ ms = number\ of\ frames * 16.67 \tag{9}$$

4.2 Results

The results in Table 1 show a total latency of 5.238 frames (87.318 ms), out of which the API latency is responsible for 4.19 frames (69.855 ms) while Opti-Speech-VMT accounts for 1.048 frames (17.464 ms).

Table 1. Latency test results.

	Avg. total latency	Avg. API latency	Avg. Opti-Speech-VMT latency
Latency (Frames)	5.238	4.19	1.048
Latency (ms)	87.319	69.855	17.464

These data are shown graphically in Fig. 6, where one can observe that the major lag between the sensor signal (red dot-dash line) and the Opti-Speech-VMT signal (blue solid line) is due to the API signal (green dotted line). The measurements also suggest the possibility of Opti-Speech-VMT latency being less than 1 frame time, since we are limited to a resolution of 1 frame with our measurements.

4.3 Conclusion

With these results, we conclude that a major portion of the total latency is due to the signal processing hardware of the EMA system being used, in this case NDI Wavefront. Opti-Speech-VMT has minimal latency implications on the system, contributing only 17.46 ms out of 87.31 ms of total latency, barely measurable with our experimental setup. Although the refresh rate of the camera and monitor limit the resolution of latency we can measure, it is sufficient for us to arrive at this conclusion.

5 Future Work

Studies on the effects of latency on visual-feedback systems suggest that the minimum latency that can adversely effect human performance can greatly differ based on the nature of the task [21–23]. Although studies such as the one by Chesters et al. [23] investigate the effects of delayed visual feedback on speech experiments, the values of delays tested in such research are significantly higher than ours. Thus, a more comprehensive study into the effect of latency in speech visual- feedback systems in specific scenarios/speech experiments would be needed to evaluate the potential impact of Opti-Speech latencies across a variety of experimental settings.

Our findings suggest the total latency can be improved based on the EMA hardware being used or signal processing optimizations by the hardware manufacturers. At the time of this writing, the NDI WAVE system is no longer in production, with its support ending soon. Opti-Speech-VMT, while capable of receiving input from NDI WAVE, is being optimized for input from Carstens AG500 series articulography systems. These devices are more accurate, with measured dynamic accuracy of 0.3 mm during speech recording [25,26], and providing sampling rates as high as 1250 samples/sec. Such instrumentation, along with a faster API, should greatly reduce potential accuracy problems resulting from system latency lags.

A concomitant improvement we are working on is to devise a means of calibration to map distances between the real world and the virtual scene, as distances for the oscillating targets are currently estimated through approximation. We have had some success with this in a recent study [27] by scaling talker's vocal tract size as a function of maximum tongue displacement. We plan to expand these efforts to provide more effective target placement in Opti-Speech-VMT.

References

1. Katz, W., et al.: Opti-Speech: a real-time, 3D visual feedback system for speech training. In: INTERSPEECH, pp. 1174–1178 (2014)
2. Wang, J., Green, J.R., Samal, A.: Individual articulator's contribution to phoneme production. In: IEEE International Conference on Acoustics, Speech and Signal Proceedings, pp. 7785–7789, May 2013
3. Wang, J., Samal, A., Rong, P., Green, J.R.: An optimal set of flesh points on tongue and lips for speech-movement classification. J. Speech Lang. Hear. Res. **59**(1), 15–26 (2016)
4. Katz, W.F., Mehta, S.: Visual feedback of tongue movement for novel speech sound learning. Front. Hum. Neurosci. **9**, 612 (2015)
5. Watkins, C.H.: Sensor driven real-time animation for feedback during physical therapy, (Masters Thesis), The University of Texas at Dallas (2015)
6. Mental, R.L.: Using Realistic Visual Biofeedback for the Treatment of Residual Speech Sound Errors, (Doctoral Dissertation), Case Western Reserve University (2018)
7. Fazel, V., Katz, W.F.: Visuomotor pursuit tracking accuracy for intraoral tongue movement. J. Acoust. Soc. Am. **140**(4), 3224 (2016)
8. Fazel, V.: Lingual speech motor control assessed by a novel visuomotor tracking paradigm, (Doctoral Dissertation), The University of Texas at Dallas (2021)
9. Bernhardt, M.B., et al.: Ultrasound as visual feedback in speech habilitation: exploring consultative use in rural British Columbia. Canada. Clin. Linguist. Phonetics **22**(2), 149–162 (2008)
10. Preston, J.L., Leece, M.C., Maas, E.: Intensive treatment with ultrasound visual feedback for speech sound errors in childhood apraxia. Front. Hum. Neurosci. **10**(2016), 440 (2016)
11. Preston, J.L., et al.: Ultrasound visual feedback treatment and practice variability for residual speech sound errors. J. Speech Lang. Hear. Res. **57**(6), 2102–2115 (2014)
12. Haldin, C., et al.: Speech recovery and language plasticity can be facilitated by sensori-motor fusion training in chronic non-fluent aphasia. A case report study. Clin. Linguist. Phonetics **32**(7), 595–621 (2018)
13. Laporte, C., Ménard, L.: Multi-hypothesis tracking of the tongue surface in ultrasound video recordings of normal and impaired speech. Med. Image Anal. **44**, 98–114 (2018)
14. Karimi, E., Menard, L., Laporte, C.: Fully-automated tongue detection in ultrasound images. Comput. Biol. Med. **111**, 103335 (2019)
15. Mozaffari, M.H., Lee, W.-S.: Encoder-decoder CNN models for automatic tracking of tongue contours in real-time ultrasound data. Methods **179**, 26–36 (2020)
16. Shtern, M., Haworth, M.B., Yunusova, Y., Baljko, M., Faloutsos, P.: A game system for speech rehabilitation. In: Kallmann, M., Bekris, K. (eds.) MIG 2012. LNCS, vol. 7660, pp. 43–54. Springer, Heidelberg (2012). https://doi.org/10.1007/978-3-642-34710-8_5
17. Tilsen, S., Das, D., McKee, B.: Real-time articulatory biofeedback with electromagnetic articulography. Linguist. Vanguard **1**(1), 39–55 (2015)
18. Suemitsu, A., Dang, J., Ito, T., Tiede, M.: A real-time articulatory visual feedback approach with target presentation for second language pronunciation learning. J. Acoust. Soc. Am. **138**(4), EL382-7 (2015). PMID: 26520348, PMCID: PMC4608962. https://doi.org/10.1121/1.4931827

19. James, K., et al.: Watch your Tongue: A point-tracking visualisation system in Blender
20. Katz, W.F., Bharadwaj, S.V., Carstens, B.: Electromagnetic articulography treatment for an adult with Broca's aphasia and apraxia of speech. J. Speech Lang. Hear. Res. **42**(6), 1355–1366 (1999)
21. Kulpa, J.D., Pfordresher, P.Q.: Effects of delayed auditory and visual feedback on sequence production. Exp. Brain Res. **224**(1), 69–77 (2013)
22. Friston, S., Karlstrum, P., Steed, A.: The effects of low latency on pointing and steering tasks. IEEE Trans. Vis. Comput. Graph. **22**(5), 1605–1615 (2016)
23. Chesters, J., Baghai-Ravary, K., Mottonen, R.: The effects of delayed auditory and visual feedback on speech production. J. Acoust. Soc. Am. **137**(2), 873–883 (2015). https://doi.org/10.1121/1.4906266
24. Bradski, B.: The OpenCV Library. Dr. Dobb's J, Software Tools (2000)
25. Berry, J.: Accuracy of the NDI wave speech research system. J. Speech Lang. Hear. Res **54**, 1295–1301 (2011)
26. Sigona, F., Stella, M., Stella, A.P., Bernardini, P., Fivela, B.G., Grimaldi, M.: Assessing the position tracking reliability of Carstens' AG500 and AG501 electromagnetic articulographs during constrained movements and speech tasks. Speech Commun. **104**, 73–88 (2018)
27. Glotfelty, A., Katz, W.F.: The role of visibility in silent speech tongue movements: a kinematic study of consonants. J. Speech Lang. Hear. Res **2021**, 1–8 (2021)