

JOURNAL ARTICLE

Gridlines approach for dynamic analysis in speech ultrasound data: A multimodal app

Simon Gonzalez

The Australian National University, AU
simon.gonzalez@anu.edu.au

Having access to efficient technologies is essential for the accurate description and analysis of articulatory speech patterns. In the area of tongue ultrasound studies, the visualization/analysis processes generally require a solid knowledge of programming languages as well as a deep understanding of articulatory phenomena. This demands the use of a variety of programs for an efficient use of the data collected. In this paper I introduce a multimodal app for visualizing and analyzing tongue contours: UVA—Ultrasound Visualization and Analysis. This app combines the computational power of R and the interactivity of Shiny web apps to allow users to manipulate and explore tongue ultrasound data using cutting-edge methods. One of the greatest strengths of the app is that it has the capability of being modified to adapt to the users' needs. This has potential as an innovative tool for diverse academic and industry audiences.

Keywords: Tongue ultrasound; gridlines; shiny app; dynamic analysis; distance; displacement; velocity; acceleration

1. Introduction

Speech analysis technologies have played a fundamental role in the understanding of how human language uses all available articulators for communication. Within the literature, there has been a strong emphasis on studying the tongue as one of the most important organs in speech production. There is a range of techniques which can be used for this purpose, including X-ray (Bressmann, Koch, Ratner, Seigel, & Binkofski, 2015; Verma, Tandon, Agrawal, & Prabhat, 2012), electropalatography (Barberena et al., 2017; Gibbon, Lee, & Yuen, 2010; N. R. Miller, Reyes-Aldasoro, & Verhoeven, 2019; Verhoeven, Miller, Daems, & Reyes-Aldasoro, 2019), magnetic resonance imaging (Hewer, Wuhler, Steiner, & Richmond, 2018; Lim et al., 2019; Maekawa, 2019; Proctor, Lo, & Narayanan, 2015), electromagnetic articulography (Katz, Mehta, & Wood, 2017; Kocharov & Evdokimova, 2019; N. R. Miller et al., 2019; Shadle, Proctor, & Iskarous, 2008; Zeroual, Hoole, & Gafos, 2019), and ultrasound (Diskin et al., 2019; Mielke, Carignan, & Thomas, 2017; Zharkova, Gibbon, & Lee, 2017). Among these, ultrasound is used to examine correlates between tongue articulations and acoustic or phonological phenomena. Gick, Campbell, and Oh (2001) note that phonological contrast may not always be observable in the acoustic signal but may be observable in other levels, like articulatory gestures. Ultrasound offers the opportunity to observe and analyze these gestures and also allows for the observation and analysis of tongue articulations in detail. An advantage of ultrasound over other tongue imaging techniques (e.g., electropalatography, electromagnetic midsagittal articulography, and X-ray microbeam) is that a great length of the tongue contour can be imaged (Davidson, 2006). It is widely used to measure an extensive mid-sagittal

section of the tongue, allowing measurements of the tongue surface from most anterior to most posterior sections by examining upward and downward movements. The only limitation of ultrasound imaging is mainly on the most anterior portions of the tongue, such as maximum constriction on dental segments and in some cases, advanced alveolar realizations (Gonzalez, 2015). Mid-sagittal sections allow measuring three main parts of the tongue, namely, the front part of the tongue, the body, and the dorsal section.

Ultrasound tongue imaging has traditionally been analyzed using either static or dynamic approaches. One of the main techniques in static approaches is the comparison of tongue contours at specific articulatory landmarks (Alwabari, 2019; N. R. Miller et al., 2019; Oakley, 2019; Roon & Whalen, 2019). An advantage of this approach is that it allows for the comparison of gestural differences at a specific time of the speech process previously established in the study, for example, differences between vowels (Decker & Nycz, 2012; Mielke et al., 2017), consonants (Ahn, 2015, 2018; Recasens & Rodríguez, 2019), or phonological contexts (Davidson, 2006; Diskin et al., 2019). On the other hand, depending on the research questions to be addressed, dynamic approaches can capture time-series characteristics which can be crucial for identifying distinctions between segments that in a static approach may not be observed. Dynamic approaches can therefore be used to study the kinetics of speech sounds (Kochetov, Faytak, & Nara, 2019; S. R. Li et al., 2019; A. Miller & Finch, 2011). One main advantage of this approach is that it can measure articulatory data in an articulatory continuum. Data for this approach generally require more preparation than in static approaches, which makes it computationally more expensive and time consuming, depending on the workflow chosen.

1.1. The general workflow of ultrasound studies

The workflow within an ultrasound study varies depending on many factors, mainly of the research questions and the resources available for the study. However, the process of ultrasound analysis can generally be broken down into five main stages: data collection, contour extraction,¹ data wrangling, visualization, and analysis.² At every stage, there are important challenges for any researcher, and here I comment on key challenges that the present study aims to address. It is important to note that these stages are not necessarily sequential, since in some cases they can happen simultaneously. During data collection, one important question is the quality of the data and the frame rate at which contours are imaged. A low frame rate poses the challenge of missing key articulatory landmarks, depending on the phonological phenomena analyzed. For example, for trills, a high frame rate is required to capture more accurate gestural timing, as in Proctor (2009). However, for other phonological phenomena, such as vowels, a lower frame rate can be used. The challenge of high frame rates is that they can be computationally too expensive, and not knowing the optimal rate in advance can result in frame rates that are higher than necessary (meaning that the resource cost exceeds the benefit of having a higher frame rate). The second stage is contour extraction. Researchers make use of many available computer programs such as EdgeTrack (M. Li, Kambhamettu, & Stone, 2005) and the Articulate Assistant Advanced (AAA) software (Wrench, 2012), which are used for semi-automatic extraction of tongue contours. For an expanded review on different methods and studies and a new approach for fully automated extractions, see Karimi, Menard, and Laporte (2019). These programs are not 100% accurate and a manual correction stage is part of the process. The third stage is data wrangling, when the data is prepared for analysis. Among the programming languages used for wrangling, R is widely used (cf. Davidson,

¹ In some studies, tongue contours are not extracted but rather the whole images are analyzed, as in Mielke et al. (2017).

² For a detailed description of specific aspects in ultrasound studies see Stone (2005).

2006; Decker & Nycz, 2012; Lin, Beddor, & Coetzee, 2014; Pini, Spreafico, Vantini, & Vietti, 2019; Recasens & Rodríguez, 2019). Developing tools using this programming language therefore can be beneficial to the research community.

The fourth stage is visualization. The importance of this stage is that it is strongly connected to the analysis to be carried out in further stages. This means that the visualization is crucial not only to identify patterns in tongue kinematics, but also to establish the most appropriate analysis approach and methods. Three key aspects are strongly considered in the visualization stage.

1.1.1. Field of View

Field of View is an important parameter for ultrasound analysis. It defines the extent of the comparable contours observed on the graphs window at any given moment. It is measured as a horizontal angle (angle aperture), which corresponds with the virtual image origin of the ultrasound probe. The field of view depends on the articulatory phenomena examined. For example, in the case of coronal segments, it is preferable that the field capture activity towards the mid and front sections of the tongue. In the case of velar segments, it is preferable that the field be more retracted to capture more activity towards the back of the tongue. Establishing the field of view is crucial because if it is not properly established, there may be key articulations missed in the range chosen for observation. It is important to highlight the fact that the field of view must be considered before purchasing a machine and transducer since this cannot always be changed.

In this paper, I have implemented a functionality for users to specify the range or subsection of the original Field of View. To distinguish the internal capability of the app from the field of view, I refer to this as the Analysis Fan View (hereafter AFV). This refers to the sections from the original tongue contours that the user chooses to focus on. For this purpose, the angle origin in the AFV is created based on the tongue contours uploaded in the app, which does not directly correspond with the angle origin from the surface of the probe used in the data collection. This is developed in more detail in Section 3.3.2.

1.1.2. Landmarks definition

Articulatory landmarks are relevant because they can be used to define the areas in the tongue to be analyzed. Tongue ultrasound imaging poses challenges in relation to defining articulatory landmarks. Since the tongue moves as a whole unit and there are no hard-defined sections, as there are in passive articulators (e.g., teeth, alveolar ridge, soft palate), landmark definitions are necessary to accurately interpret the dynamics of the tongue (c.f. Kier & Smith, 1985; Stone & Murano, 2007). In ultrasound research, landmarks can be acoustic and/or articulatory. In the case of acoustic landmarks (cf. Dawson, Tiede, & Whalen, 2016; Lawson & Stuart-Smith, 2019; Markó, Bartók, Csapó, Deme, & Grácsi, 2019; Mizoguchi, Tiede, & Whalen, 2019; Zharkova, 2013), acoustic cues are the bases for defining moments during the articulatory process, for example, the acoustic time of the mid-point of a vowel (Mielke et al., 2017) or the maximum constriction of a consonant located at the mid-point of its duration. On the other hand, articulatory landmarks are defined by the gestural behaviour of the segment. For example, the constriction location in low front vowels is located at the still frame showing the most retracted tongue contour on the tongue dorsum (as in Decker & Nycz, 2012), and the maximum constriction of a velar stop is located at the frame showing the most raised tongue body during stop closure (as in Davidson, 2006). These two approaches for landmark definition are not mutually exclusive. It is common practice in the field to use mixed approaches in which both acoustic and articulatory cues are considered.

1.1.3. Articulatory landmarks for analysis

The articulatory moment(s), or the time frame, is another important parameter and it refers to the dynamic windows for analysis, for example, analyzing transitions between a vowel and the maximum constriction of the following consonant. In static approaches, the time window is just one screenshot at a given moment, e.g., maximum constriction of a consonant or the onset of a vowel. The definition of these articulatory landmarks strongly depends on the phonological phenomena observed and the questions to be answered. In the case of dynamic studies, the purpose is to measure gestural patterns from point A to point B of a given sequence.

The last stage in the ultrasound workflow is the analysis stage. This includes the type of measurement and the statistical approaches for analysis—qualitative, quantitative, or a combination of both. For example, in static approaches, studies have used Principal Component Analysis (Stone, Goldstein, & Zhang, 1997) and Smoothing Splines to compare tongue contours and measure differences between segments (Davidson, 2006; Decker & Nycz, 2012), and dynamic studies have looked at tongue displacement (Gonzalez, 2015) and velocities (Strycharczuk & Scobbie, 2015).

2. Motivation and main purpose

As observed above, there are many factors to be considered for a solid analysis of ultrasound data in speech research. This requires strong skills both in linguistic phenomena and programming analytical tools. Data analysis is achieved by using a combination of different software and scripts built in various programming languages. It is important to note that there are standalone programs which can efficiently do ultrasound analysis and incorporate different stages simultaneously, for example the Articulate Assistant Advanced (AAA) software (Wrench, 2012), which is powerful and of great use. However, these programs generally require specialized equipment and they do not tend to be available for code expansion/modification based on users' needs. Two relevant stand-alone R libraries have been developed to facilitate data processing from AAA. These are *rticulate*: Ultrasound Tongue Imaging in R (Coretta, 2020) and *ultRa* (Beare, 2018), which offer a range of functions to import and manipulate ultrasound data. In this paper, I introduce *UVA*: Ultrasound Visualization and Analysis, which can be accessed and downloaded from <https://github.com/simongonzalez/uva>. The app implements a wide range of visualization parameters and carries out analysis of speech ultrasound data for both static and dynamic approaches. The aim is to present users with full control of visualization and analysis parameters to carry out efficient analysis of their data. It is a standalone environment which can be freely accessible and released as open source. This allows for any further fine-tuning of the code as well as implementation of new methods by modifying the code. This app is targeted at phoneticians analyzing static and dynamic data of tongue ultrasound images as well as speech practitioners interested in measuring tongue images for clinical purposes. In relation to the different stages of the workflow explained above, the app aims to enhance and facilitate mainly stages four (Visualization) and five (Analysis) with a tagging functionality for stage three (Data Wrangling). Since it does not offer collection and contour extraction, users are assumed to have tongue contours extracted in *xy* Cartesian coordinates, either in pixels and/or millimetres. In Sections 3 and 4, I describe in detail the app architecture and present a small sample study.

3. App architecture

The app implements both static and dynamic approaches from Gonzalez (2015) in a single program. The code used in the app was developed using open-source tools, combining the analytical capacities of the R environment (R Core Team, 2018) and the web-app capabilities of the Shiny library (Chang, Cheng, Allaire, Xie, & McPherson, 2019), which

is used for the creation of java-based web applications. The motivation is that R is a programming language widely used for speech analysis and Shiny allows the creation of apps that can have both the cutting-edge interactive interfaces and the power of the programming language analysis. The strength of the Shiny application framework is that it is intrinsically reactive in its programming. This means that it links input and output data and updates to the outputs (figures and tables) without refreshing the program or uploading new data, hence *reacting* to users' actions on the interface. All the programming actions in the background are translated as visual and graphic outputs at the front end, such as click-on buttons, sliders, drop-down menus. This allows users to explore data in more efficient and sophisticated ways without requiring ample knowledge of R programming. Users are only required to have basic knowledge of R such as opening and running apps.

For the proper working of the app, the input data must meet specific requirements. First of all, the number of individual points for each contour must start with 1 and end in 100. This is relevant specially for AAA users in which tongue contours do not always start with 1 but can start in higher numbers depending on the gridline capturing that point. Secondly, the app has been optimized to work with cartesian data. For future stages, I aim to implement compatibility with polar coordinates (cf. Heyne & Derrick, 2015; Mielke, 2015).

In terms of the output data, the app automatically exports wrangled tongue contours with corresponding gridlines in a working folder labelled `workingFiles`, located within the main directory. All plots can be downloaded from the app in six formats: bmp, jpeg, pdf, png, svg, and tiff. For these images, users can change the width and height of the files.

The app requires a specific structure of the data, which can be structured in four levels as shown in **Figure 1**.

First, we have the Speaker level. It is a requirement to have at least one speaker in the data, this is, the data cannot have frames or sequences without being assigned to a speaker. In this sense, the speaker level is the overarching class in the data. The second level is the Segment. The data needs at least two segments to compare. It can be either two consonants or two vowels, or one segment in different conditions, for example, consonant /t/ in final and non-final position. The app requires each of these segments to have at least two *repetitions* to be analyzed, which is the third level. Segments may or may not have the same number of repetitions. For example, segment A has three repetitions and segment B has four repetitions. The app accounts for this difference using intrinsic and extrinsic criteria (See **Figure 2**). If it is intrinsic, only the same number of repetitions are taken into account. As in the example, only three repetitions per segment are analyzed, and the fourth one for segment B is ignored. On the other hand, if the extrinsic is selected, then all repetitions are analyzed irrespective of their uneven number of tokens.

The fourth level is the number of frames. This refers to the number of frames per repetition for each segment. At least one frame per repetition is needed for the program to work properly. Similar to the number of repetitions, frames also follow an intrinsic and extrinsic selection. For each frame, the app reads four columns from the input data (See **Table 1**). The first one is the number of the frame ("frame" column). The second one is the Cartesian Coordinate to which the measurement value in column "mm" is assigned. The "coord" column specifies whether it is the x or y coordinate. The third is the "point" column and it specifies the single point in the tongue contour. In the case of contours



Figure 1: Structure levels for the input data.

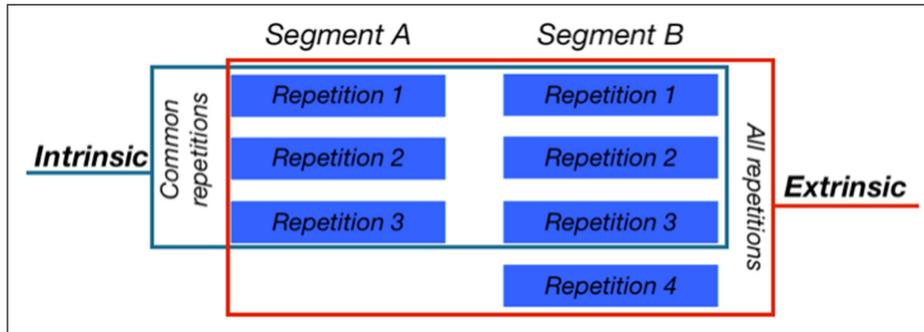


Figure 2: Intrinsic and Extrinsic criteria for repetitions.

Table 1: Sample data format for the input data showing the first five observations.

speaker	segment	repetition	frame	coord	point	mm
1	s	1	1	x	1	48.15356
1	s	1	1	x	2	48.68272
1	s	1	1	x	3	49.47646
1	s	1	1	x	4	50.00562
1	s	1	1	x	5	50.53478

extracted from EdgeTrak, each trace had 100 points in the data tested. The fourth column read to create the frame is the “mm” column. This column stores the x or y value of the point in the contour. For columns repetition, frame, and point, all counts must start with 1 without any skipping. For instance, the program will not work properly if one segment has repetitions 1 and 3 (repetition 2 is missing), or repetitions 2 and 3 (it does not start with repetition 1).

3.1. R libraries used

The app uses a range of R libraries for its full functionality. These can be organized in terms of their use within the app in six groups. The first group is the libraries used for its creation as a *java-based application* and for the main layout. Two libraries are used for this purpose: shiny (Chang et al., 2019) and shinydashboard (Chang & Ribeiro, 2018). The second group is used for the *data wrangling* and this includes pryr (Wickham, 2018) and tidyr (Wickham & Henry, 2019). The third one is used for *data managing and visualization of tables*: DT (Xie, Cheng, & Tan, 2019), data.table (Dowle & Srinivasan, 2019) and rhandsontable (Owen, 2018). The fourth group of libraries is used for *spatial calculations of lines and contour intersections*: raster (Hijmans, 2019), sp (Pebesma & Bivand, 2005), gss (Gu, 2020), and rgdal (Bivand, Keitt, & Rowlingson, 2019). The fifth group of libraries is used for the main *visualization* functionality, which includes ggplot2 (Wickham, 2016), ggrepel (Slowikowski, 2019), plotly (Sievert, 2018), highcharter (Kunst, 2019), and rAmCharts (Thieurmél, Marcelionis, Petit, Salette, & Robert, 2019). The last group of libraries is used for the *widgets layouts and display colors*: shinyjs (Attali, 2018), shinyjqui (Tang, 2019), colourpicker (Attali, 2017), shinyBS (Bailey, 2015), RColorBrewer (Neuwirth, 2014), and wesanderson (Ram & Wickham, 2018).

3.2. App structure

This section presents the general structure of the app. The app is a single web page organized by a navigation bar with all tabs available in one window. This enables going back and forward and alternating between different stages in the analysis. The motivation

is to make the analysis more holistic. Traditionally, wrangling, visualization, and analysis are done at separate and distinctive stages. But with this approach, the layout allows for switching between the different stages as needed. The app has 12 main sections and they can be divided into six subsections: data and overview, visualization, graphics, gridlines, analysis, and SSANOVA. For ease of visualization, screenshots of these sections are found in the Appendix.

3.2.1. Data and overview

There are five tabs in this section (See **Figure 3**). First is the *Home* tab, which shows a static logo and general information. The second is the *Documentation* tab and it offers general information on the different parts of the app. The third *Load* tab is where the user loads the data for analysis. The app only reads *csv* and *tab-separated* files. The data has to be previously wrangled from the specific source software so it can be read in. The fourth one is the *Data* tab. It shows the data imported in a table giving information on the number of speakers, segments, repetitions, and frames. The fifth tab is *Manage Data*. Different from the *Data* tab, in the *Manage Data* tab users can edit the data within the app. Here, the user has three options: delete, modify, and compare. In the *delete* option, users can delete one or more frames, repetitions, and speakers. The second option is the modify option, which can be used to relabel speaker names, segments, or repetition labels. The compare option makes comparisons of tongue contours of the selected rows. This is relevant when some data is being considered for deletion; having the option to visualize it beforehand is very important.

3.2.2. Visualization

The visualization section has three parts: *Data inspection*, *Landmarks*, and *In context*. The first one is the *Data inspection* tab. Here users can explore all the data imported by allowing extensive interactions with all contours using the widgets available. This section as well as all tabs that visualize data have the structure as shown in **Figure 3**. The main controls on the plotted data are located on the left pane. This includes speakers, repetitions, and frames, as well as an option to save the current plot. The main plot is located on the top-

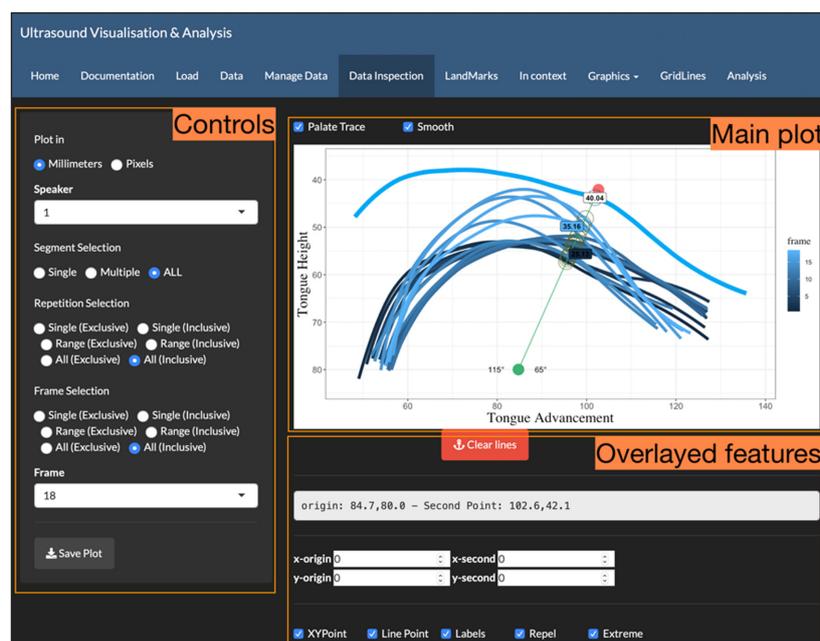


Figure 3: Structure of the main visualization tab.

right section. All the changes applied on the left controls are automatically updated on this main plot. Here, the visualization is done in individual speakers, that is, no speakers can be compared in the same graphics. One important feature in this section is that the app enables users to manually set anchor points in the visualization. Anchor points are fixed points in the figure which become the origin points from which polar-like lines are drawn. This allows for exploring areas that show more articulatory behaviour, which is the basis of the visualization and analysis approach of the app. After setting the anchor point, a second point is created by the user to create a line. The algorithm then automatically calculates distances based on the intersections between tongue contours and the line created. The app gives xy coordinates for each intersection. It also gives the angles to the left and to the right, which can be used to inspect articulatory advancement or retraction of specific segments based on angle differences (Proctor, 2009). The overlaid features are located on the bottom right. These are extra options to inspect the location of the point of origin and the option to manipulate the behaviour of other features such as lines, points, and labels.

The next visualization tab is *Landmarks*. This section allows for creating labels for articulatory landmarks to be analyzed in the data. These landmarks are not automatically created but instead they are manually established by the users. It can range from one single landmark to multiple landmarks. Users also have the option to modify the labels of the landmarks once these have been created. Landmark definition is not required for the visualization functionality of the app. However, they are required, as well as their assignation, for the dynamic analysis functionality.

The next tab is the *In Context* tab. This section assigns contours to the pre-established articulatory landmarks. This is a tagging capability of the app. Similar to the *Data inspection* tab, users can create lines with anchor points. One difference is that only one token with its corresponding repetition can be visualized, this is, no multiple repetitions are allowed in the same plot. The purpose is to establish the context in which each contour is located, allowing the user to see the previous and following contour(s). In the case of consonants, this allows for specification of the frame that contains the *maximum constriction*, which is defined as the frame where the constriction is held and then returns to a lower/resting position. Users can define the number of frames in context to be visualized.

3.2.3. Graphics

The *Graphics* tab gives extensive manipulation options to users. Users can edit the graphics of tongue contours, the palate trace, and the text on the images. For tongue contours and the palate trace, options include line colour, line thickness, alpha value, the smoothness level, and the line type. The last option is to modify the text in the plots. It includes font type, font size, and colour of the axis labels, axis ticks, and legend. The settings modified in this section automatically apply to plots in the tabs *Data Inspection*, *Landmarks*, and *In Context*.

In relation to the smoothness parameter, this is only used within the app for visualization purposes. The smoothness is therefore used for the front end to smooth tongue contours in the plots. This does not affect the calculations of the intersections as explained in Section 3.3. For the visualization, users can change the smoothness method of contours in the *Graphics* section. The methods implemented are gam and loess, as they are available in the ggplot2 (Wickham, 2016) R package.

3.2.4. Gridlines

The *Gridlines* tab is a foundational section in the app (this section is expanded in Section 3.3). This section is used to establish all gridlines in the data. This can be done on one or multiple segments at a time. The first step is to define the location of the origin point.

The first option is the manual option where the user defines where the best location is by clicking on the image. The other two options are for automatic location of the origin point, which is described in more detail in Section 3.3.1. The first option is the Wide setting. It defines the origin point considering the most anterior initial point and the most posterior last point of contours. On the other hand, the Narrow option considers the most posterior initial point and the most anterior last point. The following step after establishing the origin point is the definition of the analysis fan view, which establishes the range of comparison across contours. There are four options. The first three are the same as in the origin point, Manual, Narrow, and Wide. The extra option is the Angle option. It allows the user to define the analysis fan view based on angle degrees for the angle aperture. Once the field-view is established, the user defines the number of gridlines of the fan. The two options are by number and by angle. If chosen by numbers, the user selects the number of gridlines. If by angle, the user decides to choose the location of the grid lines by angle increments.

3.2.5. Analysis

The last tab is *Analysis*. In this section, the user has the opportunity to examine the data established in the previous sections. It is based on the intersections calculated in the *Gridlines* tab and with the landmarks created in the *Landmarks* tab and tagged in the *In Context* tab. Following the same layout structure as in the previous sections, the main controls are on the left and the main plot is at the top-right. The main difference is that there is an extra visualization at the bottom right that shows the dynamic differences between the contours selected. The four options are displacement, distance, velocity, and acceleration, which are explained in Section 3.3.4.

3.3. Analysis baseline

The analysis baseline is centred on a gridlines approach developed in Gonzalez (2015), which is similar to the one used in AAA (Wrench, 2012) and other studies (cf. Liker, Zorić, Zharkova, & Gibbon, 2019; Strycharczuk & Scobbie, 2015). These gridlines are a composition of multiple lines with the same origin point and projected in different angles to create a grid-like analysis fan view. The location of the gridlines can be determined purely on data-internal events, this is, not based on anatomical parameters but on contour behaviour. The gridlines origin is fixed for each participant. The use of gridlines

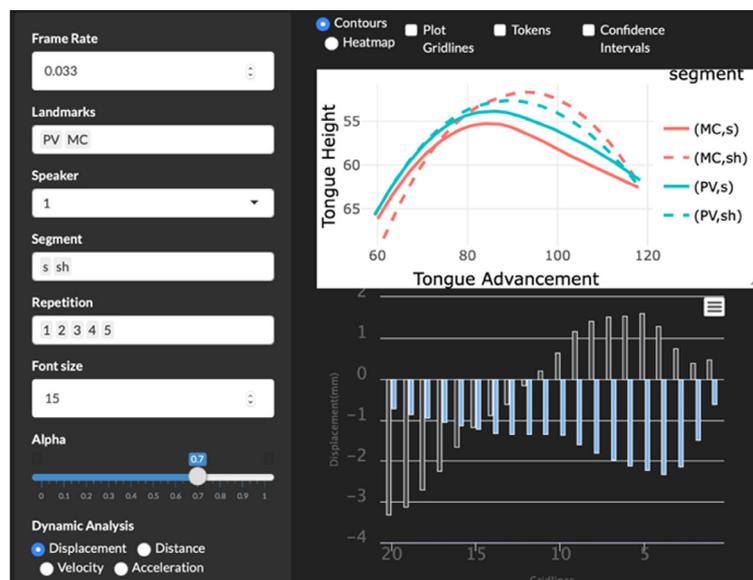


Figure 4: Layout of the Analysis tab.

is implemented to capture articulatory activity at different locations in tongue ultrasound images as well as to measure gestures in specified articulatory locations. The measurement values are defined from the intersection points between gridlines and tongue contours. Since the gridlines are fixed for all realizations of a given speaker, all the differences can therefore be interpreted as pertaining to differences in the articulatory patterns shown in the tongue contours.

This type of approach allows analysis of data in two dimensions. The first is the kinaesthetic dimension, which examines the data from static (e.g., comparing the maximum constriction points in two different segments, as observed in **Figure 4**) and dynamic perspectives (e.g., comparing how two contours change across time). The second dimension is the location of the articulatory activity, this is, analyzing contours either at a general level (e.g., comparing the articulatory activity between two segments along the full length of their surfaces), or at a specific level (e.g., comparing articulatory activity between two segments only in the tongue body section). This gridline approach can be used to carry out three types of analysis in ultrasound studies: temporal, spatial, and spatial-temporal. Temporal analyses measure time differences in the articulation between contours, for instance, which segment takes less time to reach its maximum constriction point. For the spatial analysis, the program can carry out spatial differences between contours, for example, the tongue movement from a lower position to a higher position in the vocal tract as done by displacement analyses. And the last one, spatial-temporal, carries out velocity and acceleration measurements to compare segments of interest in either whole full trajectories or isolated areas of the tongue across time. The following sections describe the three most important components of the gridlines analysis: the origin point, the analysis fan view, and the intersections.

3.3.1. Gridlines origin location

The definition of the gridlines origin location can be done manually or automatically. The first option allows the user to locate the origin point manually by clicking on the desired location within the plot. The second is the automatic option. In this case there are two further options, selection of either a narrow window or a wide window. In the Wide option, the x value of the midpoint is located between the extreme points: the most advanced point and the most retracted point of tongue contours, as shown in **Figure 5**. For the Narrow option, the x value is calculated between the most retracted first point and the most advanced last point of tongue contours.

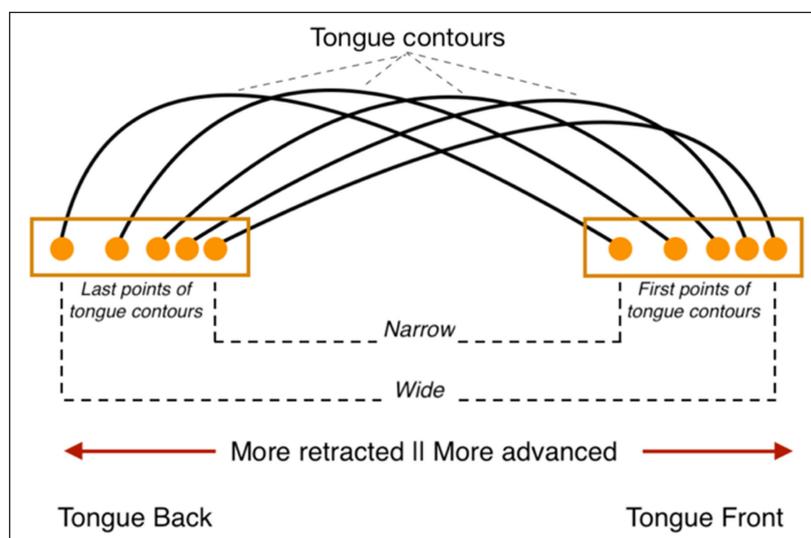


Figure 5: Definition of the x value of the Origin Point.

The y value is calculated by first extracting the highest y value of all contours and the lowest y value of all contours, as shown as the highest and lowest points in **Figure 6**. Then the final y value is calculated by subtracting the highest y value from the lowest y value.

3.3.2. Analysis Fan View

After the gridlines origin has been created, the next step is to define the Analysis Fan View (AFV), which establishes how much of each tongue contour is included in the analysis for comparison. A wider AFV can capture more articulatory activity at different sections of the tongue. However, if it is too wide, it risks having sections where not all contours are comparable due to tokens which are missing intersections in specific areas. On the other hand, a narrow AFV can be used to isolate areas of interest and analysis. The risk is that if looking at a very narrow AFV, the analysis may lose important articulatory activity that is outside the range. The trade-off between the two therefore has to be kept in mind by the researcher. The other sections of the app, especially *Data Inspection* and *In Context*, allow users to inspect areas of interest before deciding the AFV for analysis.

Similar to the gridlines origin, the AFV can be established manually or automatically. For the manual option, the user can choose the anterior and posterior lines manually by clicking on the plot or specifying the aperture angle for each of them. In the case of automatic options, it can be narrow or wide. These two depend on the angle aperture (See **Figure 7** for reference). In the case of the wide option, the left line is located at the intersection point with the widest angle. The right line is located at the intersection point with the narrowest angle. The main strength of this option is that it captures all contours across their length.

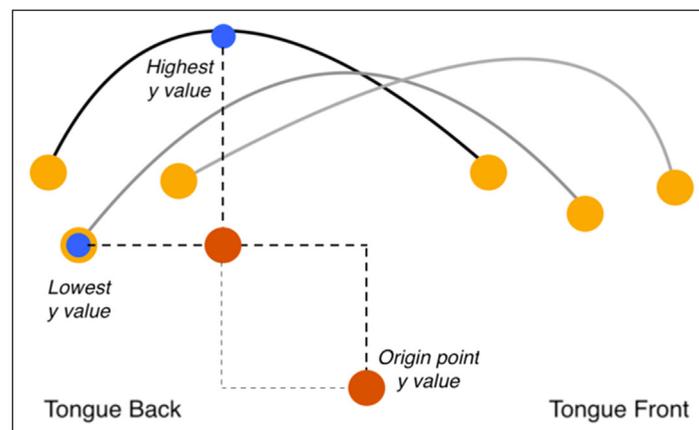


Figure 6: Definition of the y value of the Origin Point.

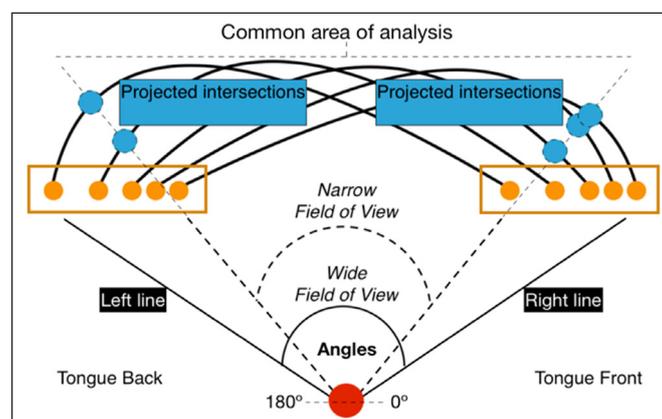


Figure 7: Creating the Analysis Fan View.

The narrow option creates an AFV that captures only the sections that are within the common area. There are two stages to calculate the second point for each line. For the left line, the algorithm calculates the angle between the origin point and the last point of each contour. It first starts with the widest angle and for each iteration it checks whether the projected line intersects with all contours. If the projected line does not intersect with all contours, the following angle is selected. The process stops when the line intersects with all contours in the data. The same process is applied to find the right line, but in this case, it starts from the narrowest angle and iterates until finding a line that intersects with all tongue contours. The result is then an angle aperture which captures a section that is common in all the data. As shown in the triangle in **Figure 7**, there is a common area which captures all contours that stay within the AFV. The areas at the left and the right outside the common area are not considered in the analysis.

3.3.3. Gridlines definition

When the AFV is established, the next step is the number of gridlines, which is decided based on either a selected number of gridlines or angle increments. In the first option, the user defines a specific number of gridlines and the algorithm divides the angle aperture of the AFV by the number of lines specified. For the angle option, the user defines the angle step increment. Then the lines are added by the angle step increment, starting from the right line and adding new lines until reaching the maximum angle within the field view aperture. In both cases, the result is a fan-like view of gridlines superimposed on all tongue contours. This allows extraction of tongue contours based on intersections between traces and gridlines. **Figure 8** shows the resulting gridlines with their corresponding angles in a sample data.

3.3.4. Intersection calculations

The following step is the intersection points. After all gridlines have been established, all intersections are calculated for all contours. The result is a new data frame with the same number of tongue contours but narrowed down to the intersection points within the AFV. This new data is the baseline for the analysis within the app. The app can analyze dynamic patterns in four ways: displacement, distance, velocity, and acceleration, which are key when examining tongue motion. In this context, tongue motion is defined as the change of a tongue section within the mouth in respect to time (how fast the tongue is moving). One parameter is displacement, which is the length of the path traveled by the tongue section from one landmark to another, based on a specific gridline. It is important to point out that this displacement is a relative measure from one tongue contour to another. Different from EMA techniques, which track tongue flesh points, the displacement calculated here can only measure the relative movement from point A to point B in a given gridline. It is represented in **Figure 9** and calculated using Formula 1 where d_{LMB} is the distance from the origin point to the intersection with a tongue contour in the second articulatory

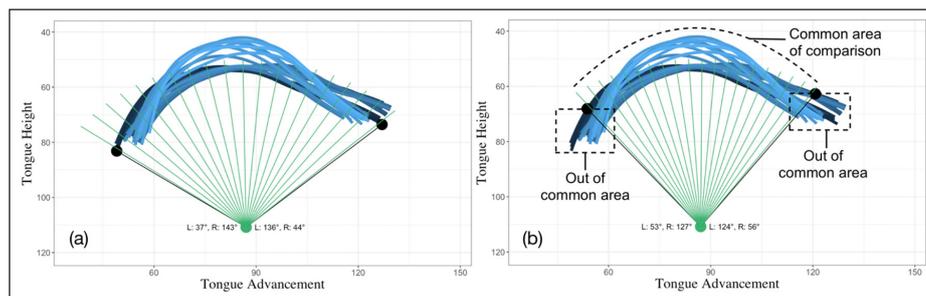


Figure 8: Final Gridlines in the data, Wide option at the left and Narrow option at the right.

landmark and d_{LMA} is the distance from the origin point to the intersection with a tongue contour in the first articulatory landmark.

Formula 1: Displacement formula.

$$d = d_{LMB} - d_{LMA}$$

It first calculates the distances from the origin point to the intersections of the first landmark contours at a given gridline. For example, the displacement of the tongue from the midpoint of a vowel to the maximum constriction of a following consonant on the 5th gridline. The displacement can therefore be positive, negative, or zero. Displacement in this framework is defined as the movement of the tongue in specific locations, which are determined by the gridlines. For each line, the displacement in mm or pixels is the distance from the first tongue contour of comparison (Intersection A in **Figure 9**) to the next articulatory moment (Intersection B in **Figure 9**) to capture more fined-tuned articulatory patterns.

The second dynamic analysis is distance. It calculates all the space the tongue has covered throughout its trajectory from the first landmark to the second landmark. **Figure 10** depicts this difference. On the left, there are three tongue intersections and they ascend in the gridline from the first intersection to the third, both landmarks. On the right, there are four tongue intersections. One difference is that from the first to the second, there is a downward movement, then it ascends to the third and finally to the fourth. As shown here, if we consider only landmark displacement, they have the same value. However, the distances for each token differ, since the example at the right has a transition contour intersection that adds more distance to the trajectory. This difference between displacement and distance is important when considering multiple contours in the trajectory between articulatory landmarks. This is also implemented in the app.

Velocity, the third dynamic analysis, measures the speed of a section of the tongue in a specified gridline (how fast the displacement is in relation to time). In Strycharczuk and Scobbie (2015), tongue contour velocity is measured based on upward or downward movements along lines placed on a fan-like shape. This measurement is also implemented

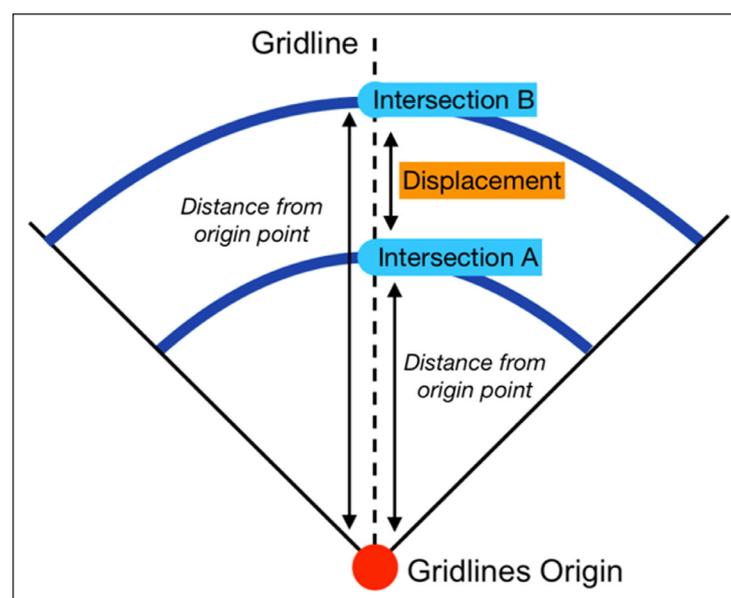


Figure 9: Displacement calculation between landmarks.

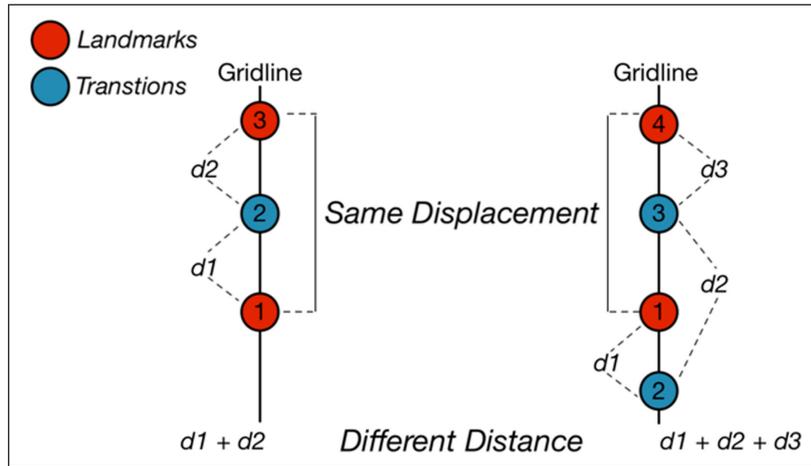


Figure 10: Difference between Displacement and Distance.

here and is calculated following Formula 2, where t represents time calculated by multiplying the number of frames between the first and second landmark by the frame rate of the ultrasound images (e.g., 30 fps are 0.033 seconds per frame-to-frame transition).

Formula 2: Velocity formula.

$$v = \frac{d_{LMB} - d_{LMA}}{t}$$

The last metric, acceleration, measures changes in the velocity of that section, with respect to time, using Formula 3.

Formula 3: Acceleration formula.

$$a = \frac{d_{LMB} - d_{LMA}}{t^2}$$

Both velocity and acceleration are based on tongue displacement: Velocity measures the rate of change of tongue displacement and acceleration measures the rate of change of the velocity. One important aspect of interpreting positive and negative values is that it depends on the measurement analyzed. In the case of displacement, distance, and velocity, negative values represent negative motion in relation to the baseline of comparison, which in this case is the first articulatory landmark. This reflects the directionality in the specified gridline. In the case of acceleration, it depends on the velocity. Since acceleration measures changes in velocity, negative acceleration takes place when there is slowing motion from one frame to another. In these terms, the first initial acceleration, from the first landmark contour to the next frame, the acceleration is positive. Then the following values can be positive or negative, depending on whether velocity slows down in frame sequences. This allows measuring a very important aspect of tongue articulations not observed at the spatial level but on a spatial-temporal dimension.

3.4. Comparing multiple segments with dynamic metrics

The tongue displacement/distance analyses examine spatial trajectories along the whole analysis fan view and the velocity/acceleration analyses look at this displacement across time. The purpose is to capture patterns of tongue gestures along extended or isolated sections of the tongue. Multiple time windows are required for this analysis, with at least

two landmarks for comparison. In the example below, I present displacement between three articulatory landmarks, namely, Previous Vowel (PV), Maximum Constriction (MC) of an obstruent consonant, and Following Vowel (FV), in the case of a VCV sequence. There are therefore two transitions, one from PV to MC (PV-MC), and another from MC to FV (MC-FV).

The analysis takes the distance intersections of the first moment as the baseline for comparison. In the case of the PV-MC displacements, the distance between the origin point and the PV intersections are the baseline. The baseline for the MC-FV displacement is the distance intersections for MC. These baseline distance intersections are compared to the distance intersections of the second moment. **Figure 11a** and **Figure 11b** show both moments. Point 1.1 in *a* shows the baseline intersection, which is the intersection in PV. The first distance calculated is the distance from the origin to the 1.1 intersection. The second distance calculated is from the origin to the 1.2 intersection. The origin-PV distance is compared to the origin-MC distance. Then the difference is calculated, which is the displacement distance in the given gridline.

In addition to the displacement distance, the analysis also examines the orientation of the displacement. If the distance in a gridline is longer for PV and shorter for MC, then it is classified as a negative displacement (as seen on the difference between point 40.1 and 40.2 on *a*). On the contrary, if the distance from the origin to the MC intersection is longer than the PV intersection, then it is classified as positive displacement (as seen on the difference between point 1.1 and 1.2 on *a*). This type of analysis allows measuring the directionality of sections of the tongue for achieving its articulatory target. After the displacement is calculated for each contour, the result is a displacement graph across all gridlines, as represented in **Figure 12**.

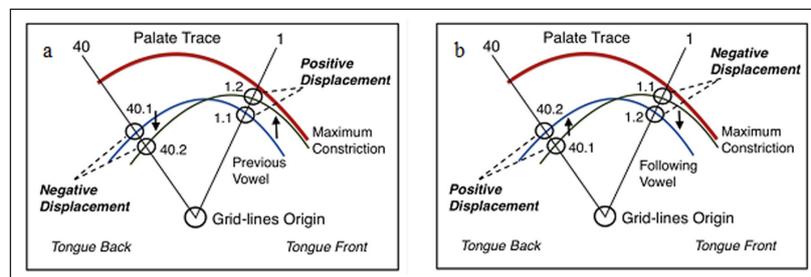


Figure 11: Positive and Negative Displacements representations.³

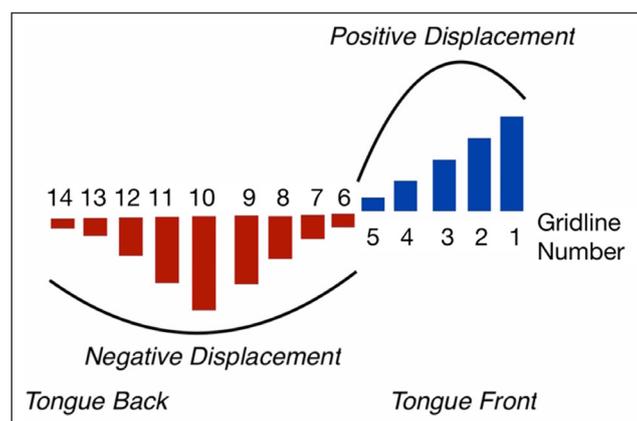


Figure 12: Representation of a displacement graph.

³ Figure 11 presents the palate trace contour. This version of the app does not have the capability to measure distances from tongue contours to palate traces. However, this is a feature that I am planning to implement in future versions.

The comparison between segments can then be achieved with the displacement graph approach. **Figure 13** represents the process of comparison. In *a* and *b*, displacements are calculated for each segment. As an example, Segment A displacement shows strong positive activity at the body and back sections of the tongue, whereas there is negative activity in the front section, but less than the positive displacement. On the other hand, Segment B does not display as much activity at the back of the tongue compared to Segment A and with little positive displacement at the first gridlines. When these are compared on the superimposed figure in *c*, the displacement shows two main distinctive differences. At the front section of the tongue, Segment B has positive displacement whereas Segment A shows negative movement. At the back section, even though both show positive displacement, Segment A shows more movement than Segment B. These differences can be assessed qualitatively as well as quantitatively.

3.5. Smoothing splines analysis of variance

The app offers a functionality to analyze tongue contours using the Smoothing Splines Analysis of Variance (henceforth SSANOVA) approach. This section is implemented following the procedure developed in Davidson (2006) and Mielke (2015). In this section, the first step is to fit the data using smoothing splines by fitting a polynomial function connecting the tongue contour points. This type of approach allows the user to measure multiple whole contour trajectories and analyze how repetitions of different sound segments compare to each other. When using SSANOVAs, the common practice when comparing tongue contours is to examine the Bayesian confidence intervals that are constructed from the multiple repetitions. If the confidence intervals of the two compared groups overlap, then this section is described as not having significant differences. On the other hand, when there is no overlap in the confidence intervals, then this section shows significant differences between the groups.

The advantage of this approach is that it allows users to examine tongue contours not as a whole, but to focus only on the specific areas that display differences between groups. This type of approach is in line with the nature of the tongue, which can display articulatory similarities between segments in one section but not in another. Therefore, by implementing SSANOVAs, the app offers another analysis tool to examine significant differences both visually and statistically sound.

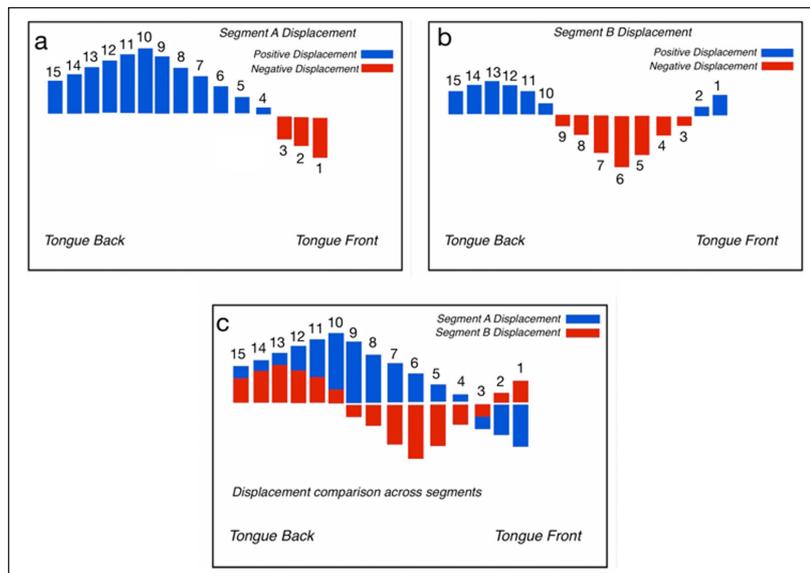


Figure 13: Representation of displacement comparisons between two segments.

In this section of the app, see **Figure 14**, the user selects the speaker and the two groups to compare. In **Figure 14**, the user compares all the repetitions for /s/ with all the repetitions for /ʃ/. There are three outputs, namely, the individual contours, the overall comparisons, and the confidence intervals. All of these, along with the numeric data, can be downloaded using the corresponding graphical user interfaces.

For the individual contours, as observed in **Figure 15**, all raw tongue contours are plotted. This helps to inspect the input data which is then used in the analysis.

For the overall comparison, the splines from the analysis are displayed (see **Figure 16**). These show the best fit for each group as well as the confidence intervals across the length of the trajectories.

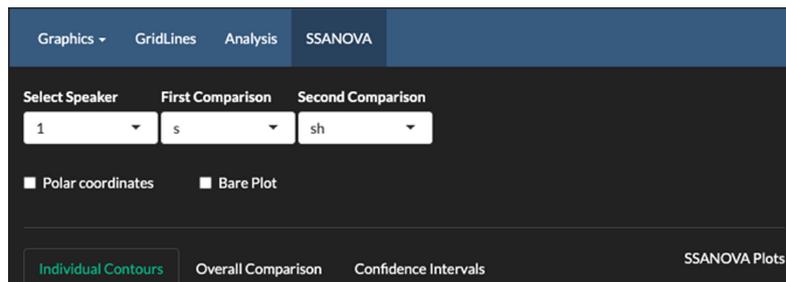


Figure 14: SSANOVA tab options.

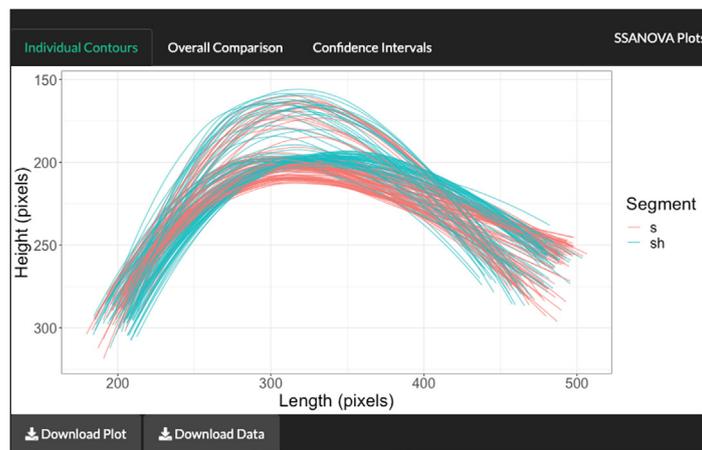


Figure 15: First visual output in the SSANOVA analysis. This shows the individual contours in each group.



Figure 16: Second visual output displaying the smoothed splines for each group.

The last tab shows the Bayesian confidence intervals for each group. The horizontal dashed line at 0 represents the baseline to identify whether areas along the trajectories are significantly different (see **Figure 17**). The areas where trajectories touch the 0 line represent sections that are not significantly different, for example, the cross-sections right at the beginning and end of both panels and around 20% in from left to right. The other non-overlapping sections are significantly different between the groups. This shows that the area around 60% into the trajectories is the area with most significant differences. This section corresponds to the tongue body differences between /s/ and /ʃ/. The fact that /ʃ/ is lower than /s/ is an artefact of the pixel measurement, which is inverted: Lower values correspond to higher positions of the tongue.

These three visual aids, along with the numeric output, give users a powerful analysis tool that can be used for effective tongue contour analysis. In this way, the widely used SSANOVA approach can also be used in the app.

4. Demonstration study

In this section, I present a sample test from a subset of the data obtained in Gonzalez (2015). For demonstration purposes, I only illustrate two segments (/s/ and /ʃ/ in English) from one speaker. Each segment has five repetitions and they appear in the carrier sentence, *Please utter X publicly*, where X is the carrier word, *sack* (for /s/) and *shack* (for /ʃ/). Each token appears between two vowels, /ə/ and /æ/. **Figure 18** shows all the tokens and the gridlines established with the Narrow option to only focus on the common area of all contours. Two articulatory landmarks were established, Previous Vowel and Maximum Constriction with 20 gridlines. All results shown below are based on the intersections from the gridlines.



Figure 17: Third visual output showing the Bayesian Confidence Intervals for each group.

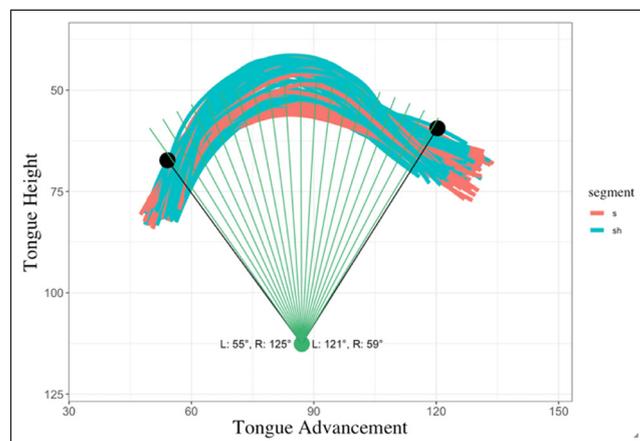


Figure 18: Sample data tokens and gridlines.

One hypothesis tested is that for PV-MC transitions the palato-alveolar segment /ʃ/ shows more positive displacement of the middle section of the tongue when compared to alveolar /s/, which is hypothesized to show positive displacement at the most advanced sections of the tongue to achieve the MC at the alveolar region. The analysis does not only focus on positive displacement patterns but also on negative displacements. **Figure 19** shows the raw plots of both segments at the MC point, /s/ shown with the solid line and /ʃ/ shown with the dashed line. The first observation is that there is more raising of /ʃ/ at the tongue body. This shows that the main difference between /s/ and /ʃ/ is that /ʃ/ has more articulatory activity at the body apex. There are no strongly distinctive differences at the front section of the tongue.

I added another type of visualization which uses heatmaps to observe articulatory activity. This is available in the *Analysis* tab. The MC heatmaps of /s/ and /ʃ/ are shown in **Figure 20**. The darker the area, the more activity there is at that specific location. As observed in the comparison, both have strong activity at the tongue body apex. The difference is that /ʃ/ shows more localized activity at that section, whereas /s/ also has more activity spreading mainly at the front section of the tongue. The heatmap visualization then adds another perspective for analyzing articulatory activity.

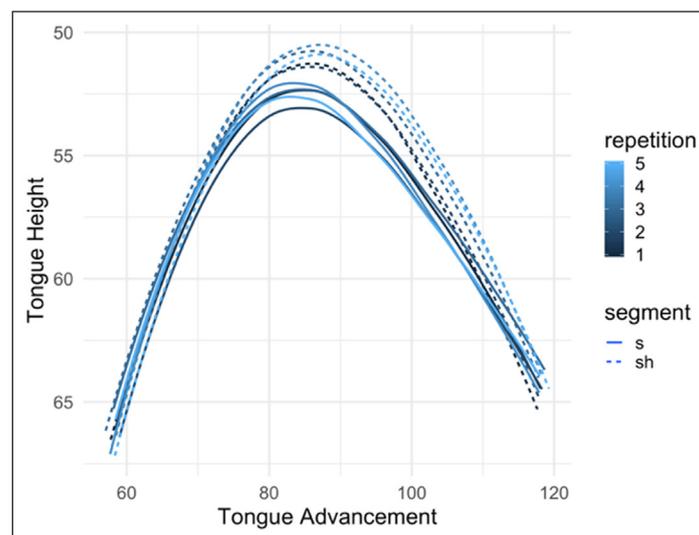


Figure 19: Raw tongue contours from the gridlines intersections.

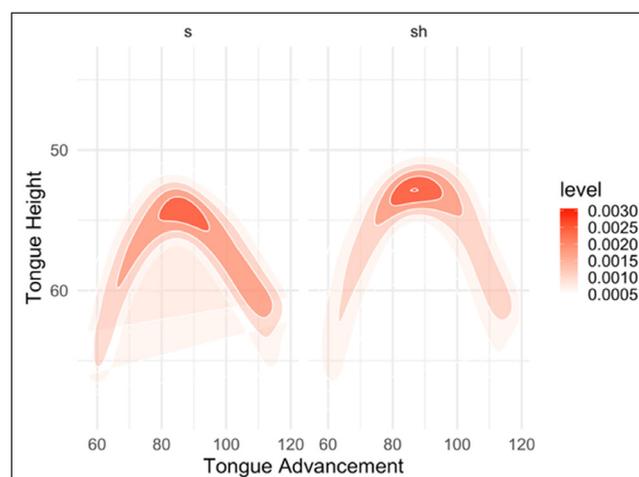


Figure 20: Heatmaps during MC across all five repetitions.

The next step is to inspect each segment individually to observe displacement differences. First, I inspect /ʃ/ in **Figure 21**. In the left I see all the five tokens at two landmarks; the solid lines are the MC contours and the dashed lines are the PV contours. The hue of the contours represents the repetition number: Darker hues are earlier repetitions. On the right, all repetitions per landmark have been grouped, which shows the same pattern as the individual tokens: MC tokens are raised and more advanced at the tongue body apex than the PV contours.

The patterns for /s/ are different (See **Figure 22**), with the individual tokens at the left and the grouped ones at the right. The patterns observed in /s/ differ in that there is lowering at the body apex from PV to MC. There is also lowering at the front section of the tongue from PV to MC. Finally, another difference is that the lowest section of the tongue at the tongue back shows less movement for /s/ than for /ʃ/.

Figure 23 presents the heatmap transitions from PV to MC, which show more local differences. As observed in the contour plots, activity at the body apex is distinctive, with /ʃ/ having more raised and advanced articulations. In the case of /s/, there is more significant gestural movement at the front section of the tongue than for /ʃ/. This reflects the expected behaviour for /s/: Since its MC is achieved at the alveolar ridge, it shows more activity at the very front sections of the contours. In the case of /ʃ/, more activity is expected at the body apex, which is the section of the tongue to achieve the MC by approaching the palato-alveolar region.

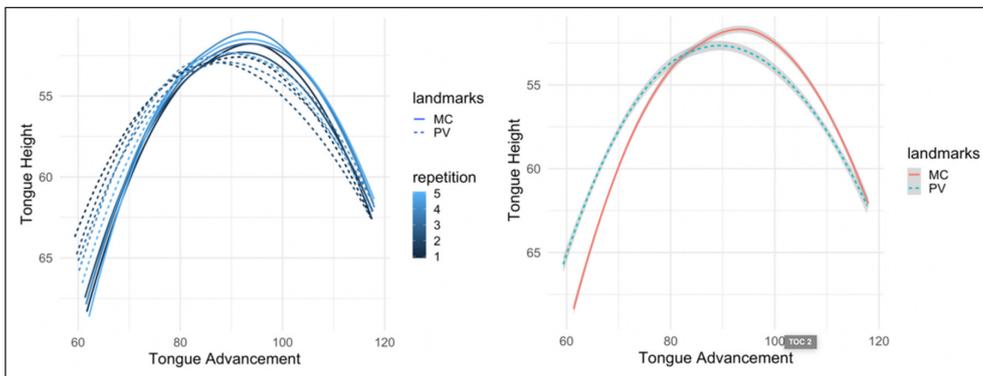


Figure 21: Contour differences from PV to MC for /ʃ/. Shaded areas in the right plot represent confidence intervals.

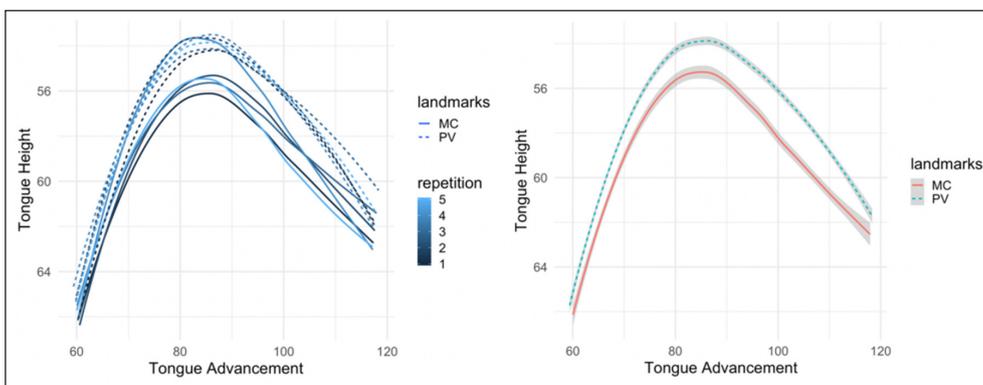


Figure 22: Contour differences from PV to MC for /s/.

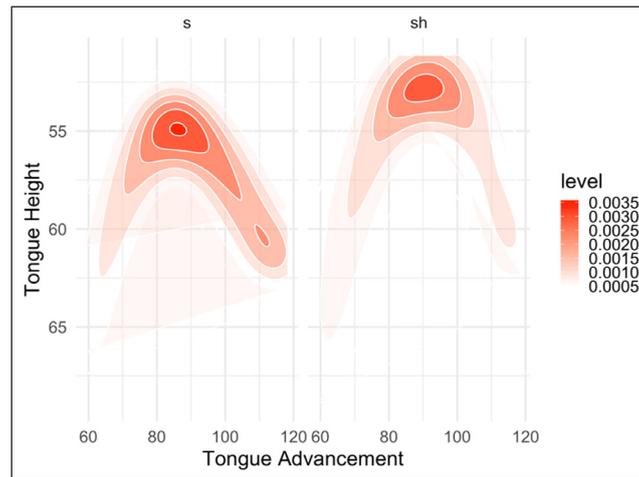


Figure 23: Heatmaps from PV to MC across all repetitions.

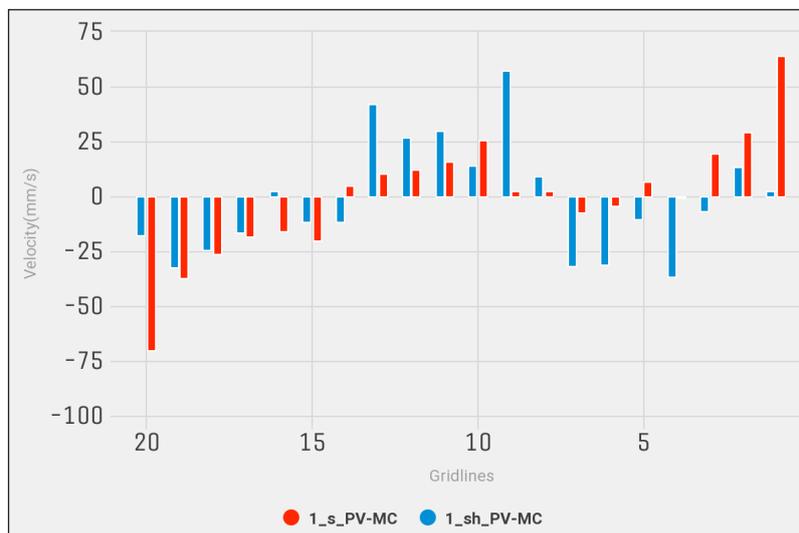


Figure 24: Velocity patterns from PV to MC.

The last type of analyses can be used to observe dynamic patterns in a deeper way. First, we analyze the velocity patterns of /s/ in red and /ʃ/ in highlighted blue in **Figure 24**. Velocity values are shown across all 20 gridlines and grouped per segment.

The comparison shows four main areas of activity. At the front section, from gridlines 1 to 3, there are positive velocities for /s/, whereas there are negative velocities for /ʃ/ between gridlines 4 and 7. The positive velocities for /ʃ/ that are comparable to /s/ are from gridlines 9 and 13. Gridlines 15 to 20 show negative velocities for both. Since velocities are closely related to the distances, we can interpret contour movement related to the time it takes to achieve the articulatory targets. First, regardless of their positive or negative values, larger velocities correspond to those areas where the movement of the tongue moves faster from PV to the MC. One key finding here is that both segments show the most prominent positive velocities exactly in those areas where the main constriction is targeted in the vocal cavity. In the case of /s/, it is at the first gridlines that would correspond to the alveolar region, where the MC is expected to take place coming from the previous vowel. In the case of /ʃ/, the gridlines with more prominent positive velocities correspond to the postalveolar region, where its MC takes place.

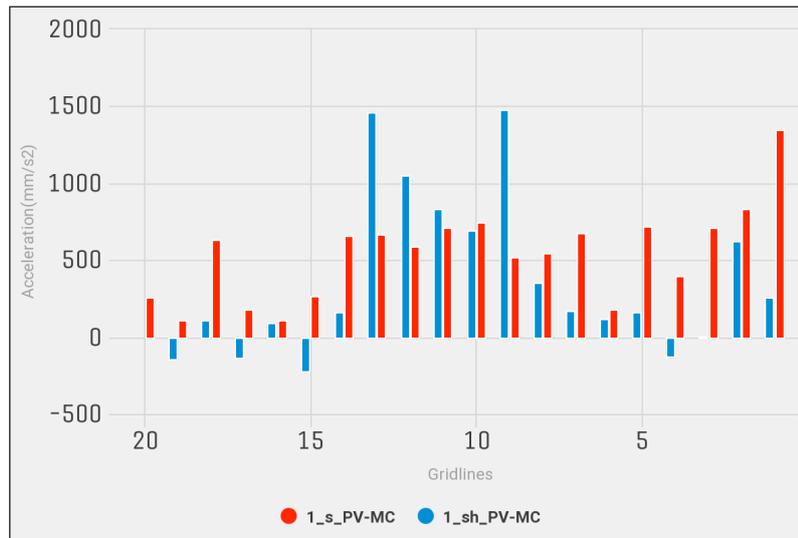


Figure 25: Acceleration patterns from PV to MC.

Acceleration results also show important patterns in **Figure 25**. As noted above, negative acceleration values correspond to transitions where there is slowing down in the articulation. Again, acceleration values are grouped across segments throughout the 20 gridlines with /s/ in red and /ʃ/ in blue.

Similar to velocity results, there is more prominent acceleration at the front section for /s/ and in the middle section for /ʃ/. It shows that there is strong extended acceleration also at the tongue body, where /ʃ/ achieves its MC. The pattern for /ʃ/ does not show strong acceleration at the front section of the tongue, except for gridline 2. Also, unlike velocity measurements, acceleration here shows that there is not much activity happening at the back section of the tongue, especially for /ʃ/. Finally, this analysis is in line with the heatmap plots in **Figure 23**. Both show that the articulatory activity of /s/ is more spread across the vocal cavity, whereas it is more localized for /ʃ/, at the middle section. Since the articulation of the palato-alveolar segment requires significant movement of the tongue body, energy shown in the acceleration suggests that most of the muscle effort is focused on moving the tongue body, leaving the back and front areas of the tongue less dynamic. On the other hand, since to achieve its MC /s/ mainly requires movement at the front section of the tongue, which has less mass than the tongue body, the muscle effort can be spread in other sections of the tongue, that is, not putting all the effort into movement of the front section, as it is done in /ʃ/ to move the tongue body.

5. Conclusions

Articulatory analysis of speech phenomena has helped us have a better understanding of how articulation and gestures function. The use of technologies like ultrasound has been of undeniable importance in the field. By developing technologies like the one described in this paper, we can expand our study of human articulation, not only in normal speech but also in cases of speech impediment and speech development. It is my aim then that by implementing this tool, many of the hurdles for efficient tongue ultrasound analysis can be overcome so we can tackle new challenges of research. Being an open source tool, UVA opens the door to broadening the scope of ultrasound studies by allowing users to create new analysis algorithms and then implement them as needed within the app from the source code.

Additional File

The additional file for this article can be found as follows:

- **Appendix.** PDF file containing the User Interface for all the main components of the app. DOI: <https://doi.org/10.16995/labphon.6463.s1>

Acknowledgements

This paper and the research behind it would not have been possible without the exceptional guidance and support of my Ph.D. thesis supervisors from which the main theoretical background of this app comes: Mark Harvey, Michael Proctor, Katherine Demuth, Susan Lin, and Alan Libert. I would also like to thank Rosey Bollington for her comments and suggestions in one of the reviews of the manuscript. The app functionality has also been used in another workshop from which very insightful comments were given by Chloé Diskin-Holdaway, Deborah Loakes, Rosey Billington, Hywel Stoakes, and Sam Kirkham. Finally, I want to thank Patrycja Strycharczuk, Stefano Coretta, the anonymous reviewers of this manuscript, and the editors, for their invaluable comments and insights in the shape and content of the final manuscript. The generosity and expertise of everyone have improved this paper in innumerable ways and saved me from many errors. Those that inevitably remain are entirely my own responsibility.

Competing Interests

The author has no competing interests to declare.

References

- Ahn, S. (2015). Utterance-initial voiced stops in American English: An ultrasound study. *The Journal of the Acoustical Society of America*, 138(1777). DOI: <https://doi.org/10.1121/1.4933625>
- Ahn, S. (2018). The role of tongue position in laryngeal contrasts: An ultrasound study of English and Brazilian Portuguese. *Journal of Phonetics*, 71, 451–467. DOI: <https://doi.org/10.1016/j.wocn.2018.10.003>
- Alwabari, S. (2019). An Ultrasound Study on Gradient Coarticulatory Pharyngealization and Its Interaction with Arabic Phonemic Contrast. In S. Calhoun, P. Escudero, M. Tabain & P. Warren (Eds.), *Proceedings of the 19th International Congress of Phonetic Sciences* (pp. 1729–1733). Melbourne, Australia: Australasian Speech Science and Technology Association Inc.
- Attali, D. (2017). colourpicker: A Colour Picker Tool for Shiny and for Selecting Colours in Plots (Version 1.0) [R package]. Retrieved from <https://CRAN.R-project.org/package=colourpicker>
- Attali, D. (2018). shinyjs: Easily Improve the User Experience of Your Shiny Apps in Seconds (Version 1.0) [R package]. Retrieved from <https://CRAN.R-project.org/package=shinyjs>
- Bailey, E. (2015). shinyBS: Twitter Bootstrap Components for Shiny (Version 0.61) [R package]. Retrieved from <https://CRAN.R-project.org/package=shinyBS>
- Barberena, L. D., Portalete, C. R., Simoni, S. N., Prates, A. C., Keske-Soares, M., & Mancopes, R. (2017). Electropalatography and its correlation to tongue movement ultrasonography in speech analysis. *Codas*, 29(2), e20160106. DOI: <https://doi.org/10.1590/2317-1782/20172016106>
- Beare, R. (2018). Processing of speech ultrasound data (Version 0.0.0.9000) [R Package].
- Bivand, R., Keitt, T., & Rowlingson, B. (2019). rgdal: Bindings for the ‘Geospatial’ Data Abstraction Library (Version 1.4-4) [R package]. Retrieved from <https://CRAN.R-project.org/package=rgdal>

- Bressmann, T., Koch, S., Ratner, A., Seigel, J., & Binkofski, F. (2015). An ultrasound investigation of tongue shape in stroke patients with lingual hemiparalysis. *Journal of Stroke and Cerebrovascular Diseases*, 24(4), 834–839. DOI: <https://doi.org/10.1016/j.jstrokecerebrovasdis.2014.11.027>
- Chang, W., Cheng, J., Allaire, J., Xie, Y., & McPherson, J. (2019). shiny: Web Application Framework for R (Version 1.3.2) [R package]. Retrieved from <https://CRAN.R-project.org/package=shiny>
- Chang, W., & Ribeiro, B. B. (2018). shinydashboard: Create Dashboards with ‘Shiny’ (Version 0.7.1) [R package]. Retrieved from <https://CRAN.R-project.org/package=shinydashboard>
- Coretta, S. (2020). Ultrasound Tongue Imaging in R (Version 1.6.0.9000) [R Package].
- Davidson, L. (2006). Comparing tongue shapes from ultrasound imaging using smoothing spline analysis of variance. *The Journal of the Acoustical Society of America*, 120(1), 407–415. DOI: <https://doi.org/10.1121/1.2205133>
- Dawson, K. M., Tiede, M. K., & Whalen, D. H. (2016). Methods for quantifying tongue shape and complexity using ultrasound imaging. *Clinical Linguistics & Phonetics*, 30(3–5), 328–344. DOI: <https://doi.org/10.3109/02699206.2015.1099164>
- Decker, P. M. D., & Nycz, J. R. (2012). Are tense [æ]s really tense? The mapping between articulation and acoustics. *Lingua*, 122, 810–821. DOI: <https://doi.org/10.1016/j.lingua.2012.01.003>
- Diskin, C., Loakes, D., Billington, R., Stoakes, H., Gonzalez, S., & Kirkham, S. (2019). The /el-/æ/ merger in Australian English: Acoustic and articulatory insights. In S. Calhoun, P. Escudero, M. Tabain & P. Warren (Eds.), *Proceedings of the 19th International Congress of Phonetic Sciences* (pp. 1764–1768). Melbourne, Australia: Australasian Speech Science and Technology Association Inc.
- Dowle, M., & Srinivasan, A. (2019). data.table: Extension of ‘data.frame’ (Version 1.12.2) [R package]. Retrieved from <https://CRAN.R-project.org/package=data.table>
- Gibbon, F. E., Lee, A., & Yuen, I. (2010). Tongue-palate contact during selected vowels in normal speech. *The Cleft Palate-Craniofacial Journal*, 47(4), 405–412. DOI: <https://doi.org/10.1597/09-067.1>
- Gick, B., Campbell, F., & Oh, S. (2001). A cross-linguistic study of articulatory timing in liquids. *The Journal of the Acoustical Society of America*, 110(5). DOI: <https://doi.org/10.1121/1.4777043>
- Gonzalez, S. (2015). *Place oppositions in English coronal obstruents: An ultrasound study*. (Doctoral dissertation). The University of Newcastle.
- Gu, C. (2020). General Smoothing Splines (Version 2.1-12) [R Package].
- Hewer, A., Wuhler, S., Steiner, I., & Richmond, K. (2018). A Multilinear Tongue Model Derived from Speech Related MRI Data of the Human Vocal Tract. *Computer Speech and Language*, 51, 24. DOI: <https://doi.org/10.1016/j.csl.2018.02.001>
- Heyne, M., & Derrick, D. (2015). Using a radial ultrasound probe’s virtual origin to compute midsagittal smoothing splines in polar coordinates. *The Journal of the Acoustical Society of America*, 138(6), EL509–EL514. DOI: <https://doi.org/10.1121/1.4937168>
- Hijmans, R. J. (2019). raster: Geographic Data Analysis and Modeling (Version 2.9-23) [R package]. Retrieved from <https://CRAN.R-project.org/package=raster>
- Karimi, E., Menard, L., & Laporte, C. (2019). Fully-automated tongue detection in ultrasound images. *Computers in Biology and Medicine*, 111, 103335. DOI: <https://doi.org/10.1016/j.compbiomed.2019.103335>

- Katz, W. F., Mehta, S., & Wood, M. (2017). Using electromagnetic articulography with a tongue lateral sensor to discriminate manner of articulation. *The Journal of the Acoustical Society of America*, 141(1), 7. DOI: <https://doi.org/10.1121/1.4973907>
- Kier, W. M., & Smith, K. K. (1985). Tongues, tentacles and trunks: The biomechanics of movement in muscular-hydrostats. *Zoological Journal of the Linnean Society*, 83(4). DOI: <https://doi.org/10.1111/j.1096-3642.1985.tb01178.x>
- Kocharov, D., & Evdokimova, V. (2019). Within-Word Articulatory Effect of Vowel Rounding. In S. Calhoun, P. Escudero, M. Tabain & P. Warren (Eds.), *Proceedings of the 19th International Congress of Phonetic Sciences* (pp. 552–556). Melbourne, Australia: Australasian Speech Science and Technology Association Inc.
- Kochetov, A., Faytak, M., & Nara, K. (2019). Manner Differences in The Punjabi Dental-Retroflex Contrast: An Ultrasound Study of Time-Series Data. In S. Calhoun, P. Escudero, M. Tabain & P. Warren (Eds.), *Proceedings of the 19th International Congress of Phonetic Sciences* (pp. 2002–2006). Melbourne, Australia: Australasian Speech Science and Technology Association Inc.
- Kunst, J. (2019). highcharter: A Wrapper for the ‘Highcharts’ Library (Version 0.7.0) [R package]. Retrieved from <https://CRAN.R-project.org/package=highcharter>
- Lawson, E., & Stuart-Smith, J. (2019). The effects of syllable and sentential position on the timing of lingual gestures in /l/ and /r/. In S. Calhoun, P. Escudero, M. Tabain & P. Warren (Eds.), *Proceedings of the 19th International Congress of Phonetic Sciences* (pp. 547–551). Melbourne, Australia: Australasian Speech Science and Technology Association Inc.
- Li, M., Kambhamettu, C., & Stone, M. (2005). Automatic contour tracking in ultrasound images. *Clinical Linguistics & Phonetics*, 19(6–7), 545–554. DOI: <https://doi.org/10.1080/02699200500113616>
- Li, S. R., Woeste, H. M., Dugan, S., Mast, T. D., Riley, M. A., Colin Annand, ... Boyce, S. (2019). Differentiating Normal Vs Misarticulated Tongue Trajectories from Ultrasound for Fast Automatic Articulatory Biofeedback. In S. Calhoun, P. Escudero, M. Tabain & P. Warren (Eds.), *Proceedings of the 19th International Congress of Phonetic Sciences* (pp. 1074–1078). Melbourne, Australia: Australasian Speech Science and Technology Association Inc.
- Liker, M., Zorić, A. V., Zharkova, N., & Gibbon, F. E. (2019). Ultrasound Analysis of Postalveolar and Palatal Affricates in Croatian: A Case of Neutralisation. In S. Calhoun, P. Escudero, M. Tabain & P. Warren (Eds.), *Proceedings of the 19th International Congress of Phonetic Sciences* (pp. 3666–3670). Melbourne, Australia: Australasian Speech Science and Technology Association Inc.
- Lim, Y., Zhu, Y., Lingala, S. G., Byrd, D., Narayanan, S., & Nayak, K. S. (2019). 3D dynamic MRI of the vocal tract during natural speech. *Magn Reson Med*, 81(3), 1511–1520. DOI: <https://doi.org/10.1002/mrm.27570>
- Lin, S., Beddor, P. S., & Coetzee, A. W. (2014). Gestural reduction, lexical frequency, and sound change: A study of post-vocalic /l/. *Laboratory Phonology*, 5(1), 9–36. DOI: <https://doi.org/10.1515/lp-2014-0002>
- Maekawa, K. (2019). A Real-Time MRI Study of Japanese Moraic Nasal in Utterance-Final Position. In S. Calhoun, P. Escudero, M. Tabain & P. Warren (Eds.), *Proceedings of the 19th International Congress of Phonetic Sciences* (pp. 1987–1991). Melbourne, Australia: Australasian Speech Science and Technology Association Inc.

- Markó, A., Bartók, M., Csapó, T. G., Deme, A., & Grácsi, T. E. (2019). The Effect of Focal Accent on Vowels in Hungarian: Articulatory and Acoustic Data. In S. Calhoun, P. Escudero, M. Tabain, & P. Warren (Eds.), *Proceedings of the 19th International Congress of Phonetic Sciences* (pp. 2715–2719). Melbourne, Australia: Australasian Speech Science and Technology Association Inc.
- Mielke, J. (2015). An ultrasound study of Canadian French rhotic vowels with polar smoothing spline comparisons. *The Journal of the Acoustical Society of America*, 137(5). DOI: <https://doi.org/10.1121/1.4919346>
- Mielke, J., Carignan, C., & Thomas, E. R. (2017). The articulatory dynamics of pre-velar and pre-nasal /æ/-raising in English: An ultrasound study. *The Journal of the Acoustical Society of America*, 142(332), 332–349. DOI: <https://doi.org/10.1121/1.4991348>
- Miller, A., & Finch, K. (2011). Corrected High-Frame Rate Anchored Ultrasound With Software Alignment. *Journal of Speech, Language and Hearing Research*, 54, 16. DOI: [https://doi.org/10.1044/1092-4388\(2010/09-0103\)](https://doi.org/10.1044/1092-4388(2010/09-0103))
- Miller, N. R., Reyes-Aldasoro, C. C., & Verhoeven, J. (2019). Asymmetries in Tongue-Palate Contact During Speech. In S. Calhoun, P. Escudero, M. Tabain & P. Warren (Eds.), *Proceedings of the 19th International Congress of Phonetic Sciences* (pp. 1734–1738). Melbourne, Australia: Australasian Speech Science and Technology Association Inc.
- Mizoguchi, A., Tiede, M. K., & Whalen, D. H. (2019). Production of The Japanese Moraic Nasal /N/ by Speakers of English: An Ultrasound Study. In S. Calhoun, P. Escudero, M. Tabain & P. Warren (Eds.), *Proceedings of the 19th International Congress of Phonetic Sciences* (pp. 3493–3497). Melbourne, Australia.
- Neuwirth, E. (2014). RColorBrewer: ColorBrewer Palettes (Version 1.1-2) [R package]. Retrieved from <https://CRAN.R-project.org/package=RColorBrewer>
- Oakley, M. (2019). Articulation of L2 French Mid and High Vowels. In S. Calhoun, P. Escudero, M. Tabain & P. Warren (Eds.), *Proceedings of the 19th International Congress of Phonetic Sciences* (pp. 1724–1728). Melbourne, Australia: Australasian Speech Science and Technology Association Inc.
- Owen, J. (2018). rhandsontable: Interface to the ‘Handsontable.js’ Library (Version 0.3.7) [R package]. Retrieved from <https://CRAN.R-project.org/package=rhandsontable>
- Pebesma, E. J., & Bivand, R. S. (2005). Classes and methods for spatial data in R [R package]. Retrieved from <https://cran.r-project.org/doc/Rnews/>
- Pini, A., Spreafico, L., Vantini, S., & Vietti, A. (2019). Multi-aspect local inference for functional data: Analysis of ultrasound tongue profiles. *Journal of Multivariate Analysis*, 170, 13. DOI: <https://doi.org/10.1016/j.jmva.2018.11.006>
- Proctor, M. (2009). *Gestural Characterization of a Phonological Class: The Liquids*. (Doctoral dissertation). Yale University.
- Proctor, M., Lo, C. Y., & Narayanan, S. (2015). *Articulation of English vowels in running speech: A real-time MRI study*. Paper presented at the 18th International Congress of Phonetic Sciences, London.
- R Core Team. (2018). R: A Language and Environment for Statistical Computing: R Foundation for Statistical Computing. Retrieved from <https://www.R-project.org/>
- Ram, K., & Wickham, H. (2018). wesanderson: A Wes Anderson Palette Generator (Version 0.3.6.9000) [R package]. Retrieved from <https://github.com/karthik/wesanderson>
- Recasens, D., & Rodríguez, C. (2019). The Effect of Voicing on Tongue Configuration for Unaspirated Stop Sequences. In S. Calhoun, P. Escudero, M. Tabain & P. Warren (Eds.), *Proceedings of the 19th International Congress of Phonetic Sciences* (pp. 418–421). Melbourne, Australia: Australasian Speech Science and Technology Association Inc.

- Roon, K. D., & Whalen, D. H. (2019). Velarization of Russian labial consonants. In S. Calhoun, P. Escudero, M. Tabain & P. Warren (Eds.), *Proceedings of the 19th International Congress of Phonetic Sciences* (pp. 3488–3492). Melbourne, Australia: Australasian Speech Science and Technology Association Inc.
- Shadle, C., Proctor, M. I., & Iskarous, K. (2008). *An MRI Study of the Effect of Vowel Context on English Fricatives*. Paper presented at the Joint Meeting of the Acoustical Society of America and European Acoustics Association, Paris, France. DOI: <https://doi.org/10.1121/1.2935246>
- Sievert, C. (2018). plotly for R. Retrieved from <https://plotly-r.com>
- Slowikowski, K. (2019). ggrepel: Automatically Position Non-Overlapping Text Labels with 'ggplot2' (Version 0.8.1) [R package]. Retrieved from <https://CRAN.R-project.org/package=ggrepel>
- Stone, M. (2005). A guide to analysing tongue motion from ultrasound images. *Clinical Linguistics & Phonetics*, 19(6–7), 455–501. DOI: <https://doi.org/10.1080/02699200500113558>
- Stone, M., Goldstein, M., & Zhang, Y. (1997). Principal component analysis of cross sections of tongue shapes in vowel production. *Speech Communication*, 22(2–3), 11. DOI: [https://doi.org/10.1016/S0167-6393\(97\)00027-7](https://doi.org/10.1016/S0167-6393(97)00027-7)
- Stone, M., & Murano, E. Z. (2007). *Speech patterns in a muscular hydrostat: Lip, tongue and glossectomy movement*. Paper presented at the Proceedings of the Third B-J-K Symposium on Biomechanics, Healthcare and Information Science, Kanazawa, Japan.
- Strycharczuk, P., & Scobbie, J. M. (2015). *Velocity measures in ultrasound data. Gestural timing of post-vocalic /l/ in English*. Paper presented at the Proceedings of the 18th International Congress on Phonetic Sciences.
- Tang, Y. (2019). shinyjqui: 'jQuery UI' Interactions and Effects for Shiny (Version 0.3.2) [R package]. Retrieved from <https://github.com/yang-tang/shinyjqui>
- Thieurmél, B., Marcelionis, A., Petit, J., Salette, E., & Robert, T. (2019). rAmCharts: JavaScript Charts Tool (Version 2.1.10) [R package]. Retrieved from <https://CRAN.R-project.org/package=rAmCharts>
- Verhoeven, J., Miller, N. R., Daems, L., & Reyes-Aldasoro, C. C. (2019). Visualisation and Analysis of Speech Production with Electropalatography. *Journal of Imaging*, 5(40), 16. DOI: <https://doi.org/10.3390/jimaging5030040>
- Verma, S. K., Tandon, P., Agrawal, D. K., & Prabhat, K. C. (2012). A cephalometric evaluation of tongue from the rest position to centric occlusion in the subjects with class II division 1 malocclusion and class I normal occlusion. *Journal of Orthodontic Science*, 1(2), 34–39. DOI: <https://doi.org/10.4103/2278-0203.99758>
- Wickham, H. (2016). *ggplot2: Elegant Graphics for Data Analysis*. New York: Springer-Verlag. DOI: <https://doi.org/10.1007/978-3-319-24277-4>
- Wickham, H. (2018). pryr: Tools for Computing on the Language (Version 0.1.4) [R package]. Retrieved from <https://CRAN.R-project.org/package=pryr>
- Wickham, H., & Henry, L. (2019). tidyr: Easily Tidy Data with 'spread()' and 'gather()' Functions (Version 0.8.3) [R package]. Retrieved from <https://CRAN.R-project.org/package=tidyr>
- Wrench, A. (2012). *Articulate Assistant Advanced User Guide*. Edinburgh: Articulate Instruments Ltd.
- Xie, Y., Cheng, J., & Tan, X. (2019). DT: A Wrapper of the JavaScript Library 'DataTables' (Version 0.8) [R package]. Retrieved from <https://CRAN.R-project.org/package=DT>
- Zeroual, C., Hoole, P., & Gafos, A. (2019). Vowel-To-Consonant Coarticulation In Moroccan Arabic. In S. Calhoun, P. Escudero, M. Tabain & P. Warren (Eds.), *Proceedings of the*

19th International Congress of Phonetic Sciences (pp. 537–541). Melbourne, Australia: Australasian Speech Science and Technology Association Inc.

Zharkova, N. (2013). Using ultrasound to quantify tongue shape and movement characteristics. *The Cleft Palate-Craniofacial Journal*, *50*(1), 76–81. DOI: <https://doi.org/10.1597/11-196>

Zharkova, N., Gibbon, F. E., & Lee, A. (2017). Using ultrasound tongue imaging to identify covert contrasts in children's speech. *Clinical Linguistics & Phonetics*, *31*(1), 21–34. DOI: <https://doi.org/10.1080/02699206.2016.1180713>

How to cite this article: Gonzalez, S. 2021 Gridlines approach for dynamic analysis in speech ultrasound data: A multimodal app. *Laboratory Phonology: Journal of the Association for Laboratory Phonology* *12*(1): 16, pp. 1–28. DOI: <https://doi.org/10.16995/labphon.6463>

Submitted: 30 September 2019 **Accepted:** 03 June 2021 **Published:** 19 November 2021

Copyright: © 2021 The Author(s). This is an open-access article distributed under the terms of the Creative Commons Attribution 4.0 International License (CC-BY 4.0), which permits unrestricted use, distribution, and reproduction in any medium, provided the original author and source are credited. See <http://creativecommons.org/licenses/by/4.0/>.



Laboratory Phonology: Journal of the Association for Laboratory Phonology is a peer-reviewed open access journal published by Open Library of Humanities

OPEN ACCESS The Open Access icon, a stylized 'O' with a person inside, representing accessibility.