SLAM 3: An Updated Stylization Model for Speech Melody

Emmett Strickland, Marc Evrard, Anne Lacheret-Dujour

SLAM 3: AN UPDATED STYLIZATION MODEL FOR SPEECH MELODY

Emmett Strickland[1], Marc Evrard[2], Anne Lacheret-Dujour[1]

[1]MoDyCo, Paris Nanterre University, France
[2]LISN, Paris-Saclay University, France
emmett.strickland@parisnanterre.fr, marc.evrard@lisn.upsaclay.fr, anne.lacheret@parisnanterre.fr

ABSTRACT

This paper presents the newest version of the annotation software Stylization and Labelling of Speech Melody (SLAM), a language-independent prosodic model that automatically annotates pitch contours in linguistic units of arbitrary length. We review the core principles of SLAM before describing several shortcomings and the innovations introduced in SLAM 3 to address them. These notably include methods implemented to minimize the influence of F0 microvariations and alignment errors and to better model the perception of short-duration pitch changes. Secondly, we present additional functionality allowing speech segments to be annotated relative to the mean pitch of their nearest neighbors, reducing the influence of downdrift on annotations. Finally, we demonstrate the utility of these changes by comparing SLAM 3 against its predecessor in terms of measured distances between their stylized outputs and the natural pitch contours used as input.

Keywords: pitch contour, stylization, prosody, automatic labelling

1. FUNDAMENTALS OF THE SLAM MODEL

Stylization and **La**belling of Speech **M**elody (SLAM[1]) is a program for automatically labeling the pitch contours of any linguistic unit fixed by the user. SLAM was developed in line with other perception-based approaches to intonation [1, 2, 3], aiming to produce simplified descriptions of natural F0 contours that represent only perceptually salient variations. SLAM describes pitch contours' relative height and shape using a finite set of discrete labels. The most distinctive benefit of this model is that it can be applied to various linguistic units of any size (e.g., syllables, words or syntactic phrases).

Users are prompted to define two separate *TextGrid* tiers. The *target tier* provides a segmentation based on the linguistic unit the user wishes to study. If the target tier segments the sound file into syllables, labels will be produced for every syllable. Meanwhile, a *support tier* provides a higher-level segmentation providing a reference pitch on which these contours are computed. Users may, for example, select the syllable as the target tier, and the utterance as the support tier. In this case, syllables would be annotated according to their pitch relative to the overall F0 of the wider utterance. Annotations therefore take into account the broader context of linguistic units.

The SLAM model represents each stylized pitch contour as a textual label containing three key pieces of information:

- The initial pitch value of the segment.
- The final pitch value of the segment
- The pitch value and relative position of the most prominent internal saliency, if applicable.

At each of these three points, pitch values are represented as one of five symbolic tones ranging from *H* (extreme-high) to *L* (extreme-low). Each of these tones traditionally covers a pitch span of 4 semitones, with the medium tone being centered on the mean F0 of the support unit.

A high flat contour would therefore be described with the label *hh*, while a falling contour beginning with a medium pitch and ending with a low pitch would be assigned the label *ml*. If an extreme internal F0 value is detected, the contour is described with an additional symbolic tone, while its approximate position is specified using a numeric value ranging from 1 (beginning) to 3 (end). The contour mmh2 describes a contour beginning and ending with a medium pitch and a high pitch maximum in the middle position. These examples are illustrated in Figure 1. These contours are calculated from the raw pitch contours used as input, after an automatic preprocessing phase using the LOWESS smoothing algorithm [4].

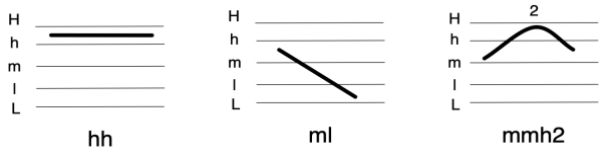

Figure 1: Examples of contour labels

SLAM+, the second major iteration of SLAM, introduced the ability to differentiate between local and global intonational registers, thus producing two

[1] https://github.com/vieenrose/SLAMplus

contours for each target. A *global contour* was based on the mean pitch of the unit located in the support tier. A *local contour* used the target segment as the support, effectively calculating the contour using the mean pitch of the target itself.

For a more detailed overview of the SLAM model, and of the core differences between the prior iterations, see [5] and [6].

2. IMPROVEMENTS IN SLAM 3

The following section presents several shortcomings of the previous iterations of SLAM, as well as the solutions introduced in SLAM 3.

2.1. Alignment errors and F0 microvariations

All stylized labels produced by SLAM include the initial and final pitch values of each alignment. The model is therefore sensitive to minor alignment errors and F0 microvariations at the edges of units. Articulatory constraints can have imperceptible impacts on F0 [7, 8]. If these occur at the edges of a target, the edge pitch values may be unrepresentative of the contour as it is perceived. To address this problem, SLAM 3 performs a linear regression, which allows a better prediction of the pitch values observed in the original contour. The endpoints of this regression are then used to determine the symbolic tones used in the stylization, as shown in Figure 2.

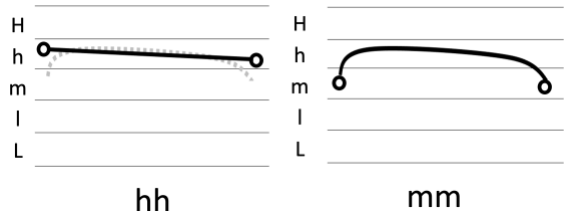

Figure 2: Improved labels enabled by linear regression (hh instead of mm)

Minor alignment errors could also cause significant inaccuracies in previous iterations of SLAM. If a target segment's alignment boundaries were slightly overstepped by a neighboring segment's F0 contour, that inaccurate information would be used in computing the target's stylized contour. This issue is typical in automatically aligned corpora. Figure 3 provides an example of a highly inaccurate contour label introduced by small misalignments at the left and right boundaries of the target.

To reduce these errors, SLAM 3 disregards continuous F0 measurements of 30 milliseconds or less, provided that they occur at the very edge of a target segment and are isolated from other voiced segments by a gap of 50 milliseconds or greater. We find this heuristic sufficient to resolve most observed alignment errors in our corpora without excluding relevant pitch information elsewhere.

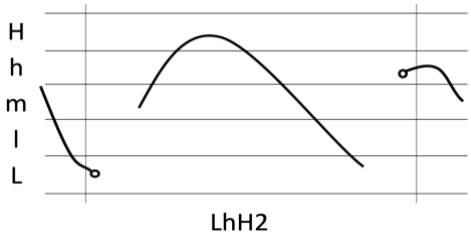

Figure 3: Annotation error caused by misalignment

2.2. Duration and the perception of pitch change

The SLAM model was designed to model pitch contours as they are perceived in linguistic units of any length. Though prior versions of SLAM could theoretically be applied to syllables and other short segments, they neglected the influence of duration in the perception of pitch changes at these levels.

Whether contours are perceived as flat tones or as glissandos is influenced not only by the pitch range traversed from start to finish, but also by the duration of the segments in question. A small pitch rise may be inaudible in a short-duration segment but easily perceived when spread over a longer timespan [9]. The *glissando threshold* refers to the rate of pitch change above which glissandos are perceived, and below which flat tones are heard. Hart et al. [1] estimated this using Formula 1, where G_t represents the threshold in semitones per second, and T the duration of the segment.

$$G_t = \frac{0.16}{T^2} \quad (1)$$

In SLAM 3, this formula is applied several times to ensure that annotations represent only pitch changes that are perceptible within a given duration. For a contour to be annotated as a rise or fall, it must now meet two conditions. As in previous versions of SLAM, its start and end points must fall within separate pitch windows associated with differing symbolic tones. However, the pitch difference between the endpoints must also exceed the glissando threshold calculated according to the timespan between those points. Otherwise, it is annotated as a flat tone whose pitch equals the mean of its endpoints.

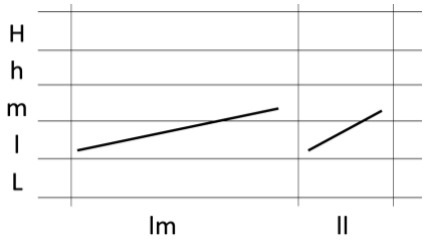

Figure 4: Differentiated annotations based on length

The same principle is used in the detection of F0 saliences, which are only included in the final annotation if the pitch changes on each side exceed the threshold, as shown in Figure 5.

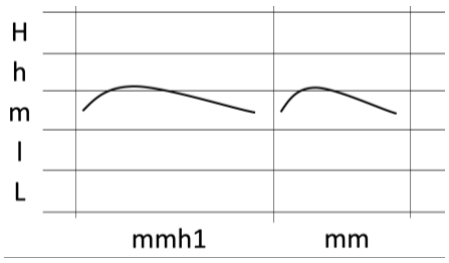

Figure 5: Differentiated annotation of pitch saliencies

In addition to annotations that better represent how short-duration F0 variations are perceived, integrating the glissando threshold also resolves a longstanding problem with the SLAM model. Associating each of the five symbolic tones with a rigid pitch range can result in flawed and inconsistent annotations in some cases. If a nearly flat F0 contour nevertheless crossed the boundary between two symbolic tones, previous versions would automatically annotate it as a glissando, even in cases where the pitch movement was imperceptibly small. This problem is thoroughly described in [10], from which Figure 6 is adapted.

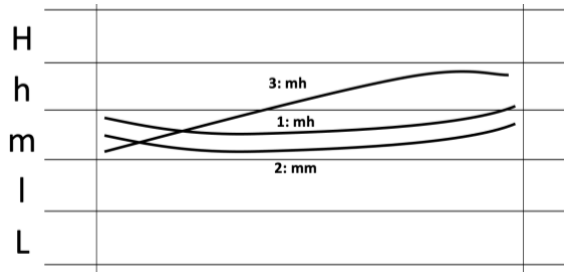

Figure 6: Inconsistent annotations from rigid categories

In this example, an imperceptible difference in height causes contour 1 and contour 2 to receive different annotations (*mh* and *mm* respectively) despite being perceptively indistinguishable. Meanwhile, contour 3 is given the same annotation as 1, despite being highly distinct from it in both shape and pitch range.

The glissando threshold ensures that contours 1 and 2 would be treated as flat tones. Meanwhile, contour 3 would be correctly annotated as a rise (*mh*), assuming its duration is long enough to permit an audible pitch change. In this specific example, linear regressions also provide additional protection against inconsistent annotations. Since contour 1 falls within the medium window for most of its duration, the associated linear regression would fall entirely within the medium window. Consequently, contours 1 and 2 would both be annotated as *mm*.

2.3. Local contours and the F0 of target neighbors

As discussed in section 1, the previous version of SLAM provided two different registers for generating stylized contours: a global register based on the mean F0 of a larger linguistic unit used as the support, and a local register that used the target as the support. Under this system, local contours accurately depicted rises, falls, and internal pitch saliencies. However, flat contours were systematically labeled *mm* regardless of their pitch values because the *m* label was centered on the mean F0 of the segment. This meant that even contours at the extremely high or low ends of the pitch spectrum were labeled *mm*, signaling a high degree of information loss regarding height.

To increase the utility of the local register, SLAM 3 introduces a sliding local support window spanning N units to the left and right of the target. Its size can be directly set by the user. When producing a local contour label for a syllable, the *m* tone label is therefore centered on the mean F0 of a larger segment comprising the target and its neighbors. If it is assigned a size of 1, the window will span from the beginning of the target's left neighbor to the end of its direct neighbor to the right. A window of size 2 will include two neighbors on each side, while a window of size 0 will function like SLAM+ by considering only the mean F0 of the target. Note that segments are only included in this window if they fall within the same support segment. This sliding support window, visualized in Figure 7, also helps to mitigate downdrift's effects on the stylization, since the pitch labels only consider the target's most immediate context.

Utterance 1	*Utterance 2 (global support)*					
...	*Syl 1*	***Syl 2***	*Syl 3*	*Syl 4*	*Syl 5*	N = 0
...	*Syl 1*	***Syl 2***	*Syl 3*	*Syl 4*	*Syl 5*	N = 1
...	*Syl 1*	***Syl 2***	*Syl 3*	*Syl 4*	*Syl 5*	N = 2

Figure 7: Target (Syl 2) and local support of size N

2.4. Customizability and ease of use

This paper has so far presented various improvements introduced in SLAM 3. While the correction of misalignments, the use of linear regressions, the glissando threshold, and the sliding window for local contours are all valuable additions, we have also aimed to make this iteration of SLAM as customizable as possible. All these additions can therefore be activated or deactivated independently in the software parameters. If all of them are deactivated, the software simply functions according to the same principles as its predecessor, SLAM+.

Finally, SLAM 3 allows annotations to be automatically exported to a *tsv* file containing each segment's duration, textual content, and stylized contours in the target tier. This lets users directly analyze results using statistical analysis packages or spreadsheet software.

3. EVALUATING SLAM 3

3.1. Data and methodology

To compare the effectiveness of SLAM 3 to that of its immediate predecessor, we produced a series of stylizations using both versions of the software. Our dataset was comprised of two random samples of ten files selected from two corpora representing languages of differing prosodic typologies: the Rhapsodie [11] corpus of French (Corpus 1), and the NaijaSynCor [12] corpus of Nigerian Pidgin (Corpus 2). In all files, word-level alignments were used as the target, with the utterance used as the support. Contours were also generated using a variant of SLAM 3 that disregarded the glissando threshold (represented as S3-G in section 3.2).

The stylizations were then converted into numerical lists representing pitch contours, from which the mean absolute error (MAE) and mean squared error (MSE) were computed between the reconstructed contours and smoothed versions of the originals. Only the global register was considered in this study, since SLAM 3 bases its local register on a sliding support that was not featured in previous versions.

3.2. Results

	Corpus 1		**Corpus 2**	
	MAE	**MSE**	**MAE**	**MSE**
S+	1.106	1.467	1.036	1.347
S3	1.021	**1.369**	0.975	1.278
S3-G	**1.011**	1.381	**0.943**	**1.258**

Table 1: Results of evaluation (best scores in bold)

This test yielded similar results on both corpora. SLAM 3 consistently produced a lower MAE and MSE than SLAM+. This indicates a meaningful improvement over the previous version. Excluding the glissando threshold resulted in a more modest reduction in the mean error in most cases. In Corpus 2, this yielded an approximately 3% reduction in both MAE and MSE compared to SLAM 3. The impact of this approach on Corpus 1 is negligible, with an approximately 1% reduction in MAE coupled with an equivalent increase in MSE.

4. CONCLUSION AND DISCUSSION

We have presented several changes introduced to address shortcomings in the previous versions of the SLAM prosodic modeling software. These innovations were then evaluated by comparing the performances of the most recent version of SLAM and its predecessor in a task designed to measure similarity between their stylizations and the original contours used as input.

For both corpora used in this study, SLAM 3 performed significantly better than SLAM+ at modeling the smoothed pitch contours used as inputs. These results nevertheless contain an inconsistency that merits further exploration. In Corpus 1, excluding the glissando threshold made a negligible difference in the mean error between the input contours and those reconstructed from the stylized labels. However, doing so significantly improved the performance of SLAM 3 on Corpus 2.

These varying results can be explained by fundamental differences between the two corpora. Files in Corpus 2 are characterized by a higher speech rate and more spontaneous discourse. The corpus should therefore contain more short-duration segments labeled as flat tones when the threshold is considered, leading to less accurate labels when measured in terms of objective pitch movements. Corpus 1, with its slower speech rate, would contain fewer pitch movements labeled as flat tones. Prosodic differences between French and Nigerian Pidgin are also likely to have played a role. Given the presence of lexical tone in the latter, it is logical that Corpus 2 would contain more monosyllabic words with large pitch changes.

These results highlight the utility of the features introduced by SLAM 3. However, they also underline the need to supplement our evaluations with perceptual tests. While the glissando threshold contributed to inconsistent mean error rates, it should not be concluded that it weakens the model. Indeed, this addition is intended to better represent pitch changes as they are *perceived*. The test performed in this study measures how well the model represents any change in F0, including those that are not perceptible.

In future studies of SLAM, we will provide a more comprehensive evaluation approach that combines both objective metrics and perception tests involving multiple unbiased participants. Informal explorations of our data already reveal promising results in this regard, with labels based on the glissando threshold better representing contours as we perceive them. Given the highly customizable nature of this latest version, SLAM 3 also allows users to choose the modeling approach which best suits their needs.

5. ACKNOWLEDGEMENTS

We would like to thank Luigi Liu for his technical contributions to the development of SLAM 3.

6. REFERENCES

[1] J. T. Hart, R. Collier, and A. Cohen, A perceptual study of intonation: an experimental-phonetic approach to speech melody, Cambridge: Cambridge University Press, 1990.

[2] P. Mertens, “The prosogram: Semi-automatic transcription of prosody based on a tonal perception model.,” in *Speech Prosody*, Nara, 2004.

[3] P. Mertens, “Polytonia: a system for the automatic transcription of tonal aspects in speech corpora,” *Speech Sciences,* vol. 4, no. 2, pp. 17-57, 2014.

[4] W. Cleveland, “LOWESS: A program for smoothing scatterplots by robust locally weighted regression,” *American Statistician,* vol. 35, no. 1, p. 54, 1981.

[5] N. Obin, J. Beliao, C. Veaux and A. Lacheret, “SLAM: Automatic Stylization and Labelling of Speech Melody,” in *Speech Prosody 2014*, Dublin, 2014.

[6] L. Yu-Cheng, A. Lacheret-Dujour and N. Obin, “Automatic Modelling and Labelling of Speech Prosody: What's New with SLAM+ ?,” in *International Congress of Phonetic Sciences (ICPhS)*, Melbourne, 2019.

[7] J. Kirby and R. Ladd, “ Effects of obstruent voicing on vowel F0: Evidence from "true voicing" languages,” *The Journal of the Acoustical Society of America,* vol. 140, no. 4, pp. 2400-2411, 2016.

[8] H. Torres, M. Güemes, J. Gurlekian and D. Evin, “F0 Perturbation Due to Articulatory Movements: Filtering, Characterization and Applications,” *IEEE/ACM Transactions on Audio, Speech, and Language Processing 29,* vol. 29, pp. 1977-1986, 2021.

[9] C. d'Alessandro, S. Rosset and J.-P. Rossi, “The pitch of short-duration fundamental frequency glissandos,” *The Journal of the Acoustical Society of America,* vol. 104, no. 4, pp. 2339-2348., 1998.

[10] R. Dall and X. Gonzalvo, “JNDSLAM: A SLAM extension for Speech Synthesis,” in *Speech Prosody*, Boston, 2016.

[11] A. Lacheret-Dujour, S. Kahane, J. Beliao, A. Dister, K. Gerdes and e. al.., “Rhapsodie: un Treebank annoté pour l’étude de l’interface syntaxe-prosodie en français parlé,” in *4e Congrès Mondial de Linguistique Française*, Berlin, 2014.

[12] B. Bigi, B. Caron and A. Oyelere, “Developing Resources for Automated Speech Processing of the African Language Naija (Nigerian Pidgin),” in *8th Language & Technology Conference: Human Language Technologies as a Challenge for Computer Science and Linguistics*, Poznan, 2017.