A simplified version of the OpS algorithm for pitch stylization

Antonio Origlia, Francesco Cutugno

LUSI-Lab, Dept. of Electrical Engineering and Information Technology
University of Naples "Federico II" - Italy

antonio.origlia@unina.it, cutugno@unina.it

Abstract

In this work we present a new version of our previously published Optimal Stylization (OpS) algorithm for pitch stylization. Here we give a better perceptual representation of the pitch curve for linguistics research. While the OpS algorithm produced good stylizations for naive listeners, when deployed in a prosodic analysis tool, we observed that, under specific conditions, important details were missed in the stylized curve to an expert's ear. Changes introduced in the dynamic tonal perception model to solve these problems resulted in a simpler and more robust model. We show how the new version of the OpS algorithm is able to recover these situations while not significantly altering the original OpS curves.

Index Terms: pitch stylization, tonal perception, prosody

1. Introduction

Prosodic research focuses on messages transmitted through the use of intonational strategies. While the $F0$ curve is indeed the main correlate of intonation, it does not represent what it is actually *heard* by the human ear. In [1, p. 25], it was stated that *No matter how systematically a phenomenon may be found to occur through a visual inspection of F0 curves, if it cannot be heard, it cannot play a part in communication*. This led to the definition of *stylization* as an approximation of the $F0$ curve by means of linear segments. In [1, p.42], this was defined as a sequence of segments that *[...] should eventually be auditorily indistinguishable from the resynthesized original and it must contain the smallest possible number of straight-line segments with which the desired perceptual equality can be achieved*.

Among the attempts to produce an account of the intonational account, the MOMEL algorithm [2] has been widely used in the literature. This algorithm does not produce a *proper* stylization as its goal is to produce a model of the macroprosodic component, which can be used together with the microprosodic component the algorithm produces to rebuild the original pitch curve. In this sense, a stylization should be intended as a lossy filter for microprosody while the output of the MOMEL algorithm does not discard the microprosodic component. Nevertheless, the macroprosodic profile obtained with MOMEL is usually considered as reference for stylization algorithms.

In [3], the concept of dynamic tones, or glissandos, was used in order to produce a stylization of the pitch curve. The Prosogram, a perceptually motivated representation of the pitch curve [4] is based on this algorithm. This representation includes a segmentation of the considered utterance into syllables to represent the pitch curve in terms of glissandos and static tones. In [5, 6], the concept of syllables was used again to position the linear segments used in the stylization. In [7], the pitch stylization problem was treated as an optimization problem for the first time by using a Dynamic Programming algorithm designed to optimize the position of a predefined number of segments estimated on the basis of the findings presented in [8]. As a quality measure, this algorithm used the statistical closeness between the stylized curve and the original one.

In [9], we presented the OpS framework along with an investigation of the possibility of using prominence information to reduce the number of points used to stylize non-prominent areas. The OpS algorithm uses a *divide et impera* strategy to balance a cost measure, based on the number of points used by the stylized curve, and a quality measure. In [9], we showed that statistical closeness does not necessarily reflect the results of the listening test so, in [10], we presented an updated version of the algorithm using a tonal perception model to compute the quality measure. While this model has the same basis of [3], it is dynamic in the sense that it uses the findings of [4] and the indications coming from the experiments with the OpS version using prominence annotation to avoid using rigid thresholds. Also, by retaining the generic OpS framework, it explicitly takes into account the cost of the curve during computation, closely following the definition of stylization.

In this paper, we summarize the parts of the OpS algorithm that have been modified to obtain the new version, we highlight the problems that the algorithm had in retaining certain classes of details that are important for linguistics research and we present the changes we introduced. Qualitative and quantitative tests performed on the same corpus we used for the objective tests in our previous works show that these changes do not alter the OpS curve on a large scale. By means of a case study, we show that the details we were interested in recovering are correctly represented by the new version of the algorithm.

2. The OpS algorithm

In [10], we presented a new version of the OpS algorithm substituting the original quality measure $q(S, \bar{S})$ with a new measure based on a tonal perception model, following the approach of [3]. This tonal perception model was dynamic in the sense that it did not use rigid thresholds to model the human ear's capabilities of perceiving dynamic tones (glissandos). This was achieved by considering the effect energy movements have on tonal perception (i.e. [11]) by taking as reference the Spectral Constraint Hypothesis (SCH) [12] and by relying on a continuous value to describe the *glissando likelihood* Γ_g of a pitch movement based on the findings of [4]. The reader is referred to [10] for details regarding the computation of the Γ_g value. For reasons of space, in this paper we summarize only the parts that we modified to obtain the new version of the algorithm.

First of all, we describe the segmentation strategy adopted during the first phase of the *divide et impera approach*. Given a generic pitch curve, the algorithm splitted this curve into two

10.21437/SpeechProsody.2014-187

subcurves sharing and an endpoint by choosing the first local maximum, if present, and choosing the midpoint otherwise. This is because, by considering them later during the backtracking phase, tonal peaks were implicitly considered more important than other points, as their removal influenced large portions of the final curve.

During the backtracking step of the *divide et impera* schema, the removal of the point shared by two adjacent subcurves $A = [a_1, \ldots, a_n]$ and $B = [b_1, \ldots, b_m]$, with $a_n = b_1$ was evaluated. The two possible mergings of the two subcurves were either the curve where the shared point was kept $S = [a_1, \ldots, a_n, b_2, \ldots, b_m]$ or the curve where it was not $\bar{S} = [a_1, \ldots, a_{n-1}, b_2, \ldots, b_m]$. The overall quality function $F(S)$, computed as the balance between perceptual equality and cost, was compared with $F(\bar{S})$. If $F(\bar{S})$ had a higher value than $F(S)$, the $\bar{S}$ curve was passed to the next backtracking step. For the sake of simplicity, in the following formulas we assume that the two curves being compared have the same number of points. Of course, when the removed point s_i is considered in the $\bar{S}$ curve, $\bar{s}_i$ corresponds to the value obtained by linearly interpolating s_{i-1} and s_{i+1} in t_{s_i}.

To compute the quality of the stylized curves, the algorithm considered the accumulated difference between the original curve and the proposed one in terms of glissando likelihood for each of the segments in the curve, evaluating the risk of introducing a glissando where a static tone was found and vice versa. The distance D between a generic segment $[s_i, s_{i+1}]$ and its stylized counterpart $[\bar{s}_i, \bar{s}_{i+1}]$ was computed as:

$$\begin{aligned} &D([s_i, s_{i+1}], [\bar{s}_i, \bar{s}_{i+1}]) = \\ &D_{acc}([s_i, s_{i+1}]) + \Gamma_g([s_i, s_{i+1}]) - \Gamma_g([\bar{s}_i, \bar{s}_{i+1}]) \end{aligned} \qquad (1)$$

The quality of the $\bar{S}$ curve with respect to the S curve was computed as the weighted mean glissando likelihood accumulated distance over the segments of the $\bar{S}$ curve. The weights were the time portions of the complete curve represented by each segment, thus obtaining:

$$\begin{aligned} &q_g(S, \bar{S}) = \\ &\sum_{i=1}^{n-1} \left((1 - |D([s_i, s_{i+1}], [\bar{s}_i, \bar{s}_{i+1}])|) \frac{t_{s_{i+1}} - t_{s_i}}{t_{s_n} - t_{s_1}} \right) \end{aligned} \qquad (2)$$

The model presented in [10] also takes into account a $q_d(S, \bar{S})$ measure related to differential glissando perception checking that a glissando, if present, is kept as it was. In this paper we concentrate on $q_g(S, \bar{S})$ as the changes we made to this function are simply replicated in $q_d(S, \bar{S})$.

Concerning the cost measure, in [9] it was computed as the ratio between the number of points used by the stylized curve and the number of points found in the original curve transformed with a sigmoid function. After introducing the dynamic tonal perception model, in [10] we reported a problem causing the quality measure to rapidly dominate the cost measure in long, continuous pitch curves and causing the insertion of more target points than necessary. In [10] we counterbalanced this effect by introducing an empirically determined α modifier to augment the weight of the cost factor depending on the length of the curve in the cost function. The cost function used by the OpS algorithm is:

$$c(S, \bar{S}) = 1 - \left(\frac{1}{1 + exp(\frac{-(x^{\alpha} - 0.5)}{0.1})} \right) \qquad (3)$$

where x is the ratio between the number of points used by the stylized curve and the number of points used by the original one.

3. Observed problems

The results of the perceptual tests reported in [9, 10], in which naive listeners were recruited, indicated that the stylization proposals of the OpS algorithm performed, in terms of quality, in a similar way with respect to other approaches. The OpS algorithm had the advantage of being parameter independent and it was able to use less points by explicitly taking into account a cost measure during computation. In [13], we included the OpS algorithm in the Prosomarker tool: an instrument designed to give a perceptual account of the pitch curves and to describe the synchronization of the pitch targets with automatically detected segmental events (syllable boundaries and nuclei). While using this tool to describe simple intonation phenomena, we were able to trace a number of recurring situations in which the OpS algorithm was not able to capture specific classes of details from the curve that appeared to be critical to an expert linguist's ear.

1) In [10] we found that giving priority to local minima if no local maxima can be found in the curve during the splitting phase did not seem to introduce improvements. Not having this rule introduced the possibility that a local minimum was evaluated very early during backtracking. As we have seen, this implicitly assigns less importance to the point because the impact of its removal is evaluated on a limited portion of the curve. This was systematically noticed by the human experts evaluating the quality of the OpS curves while testing Prosomarker, as they were able to detect small discrepancies both in timing and in tonal level of lowering targets in the resynthesis with respect to the original utterance.

2) The quality measure dominating the cost one in long pitch segments was not completely addressed by the introduction of the α parameter. Continuous pitch segments, longer than the ones we tuned α on, were found in other corpora: in these segments the effect was strong enough to make the α weighting useless. The presence of the α parameter is also less motivated from a theoretical point of view than the rest of the model, thus making the framework less reliable than we intended.

3) When local maxima split the curve in two subcurves that are very unbalanced in length, the algorithm was unable to adequately protect the smaller part of the curve. The quality of the longer subcurve was considered more important than the quality of the shorter subcurve that, subsequently, was often overstylized. This was caused by the weighting of each segment dependently of the fraction of time it stylized.

4. The SOpS algorithm

We now present the updates to the OpS algorithm we introduced in order to address the problems we highlighted in the previous section. The final model we obtain is simplified with respect to the preceding version. For this reason, we will refer to the updated version of the OpS algorithm as the Simplified Optimal Stylization (SOpS) algorithm.

To address problem 1, we reintroduced the splitting rule giving priority to local minima if no local maxima can be found. By evaluating these points later during the backtracking phase, the SOpS algorithm is able to protect low targets better than the OpS algorithm. Problems 2 and 3 were both related to the measure we used to evaluate shared endpoints removal during backtracking. Specifically, having the whole subcurves influence the quality measure introduced the problems related to differences in the curves' length. However, the removal of the shared endpoint, while generically influencing the quality of the two curves' mergings, is more specifically related to the quality of the two neighboring *segments*. Back to the preceding example, given the A and B curves, the removal of the shared point $a_n = b_1$ only influences the quality of the $[a_{n-1}, a_n]$ and $[b_1, b_2]$ segments. Therefore, having the quality evaluation of the curves $[a_1, a_{n-1}]$ and $[b_2, b_m]$ contributing to the evaluation introduces an identical factor on both sides of the comparison operator. Eliminating this factor makes the algorithm take into account only the neighboring segments quality. By weighting equally these two segments, we also remove the effect of longer movements being considered more important than shorter ones. Equation 2 is reformulated as

$$\begin{gathered} q_g(S, \bar{S}) = \\ \{2 - |D([s_{i-1}, s_i], [\bar{s}_{i-1}, \bar{s}_i])| - \qquad (4) \\ -|D([s_i, s_{i+1}], [\bar{s}_i, \bar{s}_{i+1}])|\}/2 \end{gathered}$$

Also, by considering local minima earlier in the splitting phase of the *divide et impera* schema, the *midpoint split* rule is applied to segments that are either quasi-linear or parabolic. In the first case, small differences are introduced by removing points while, in the second case, the *midpoint split* rule rapidly produces quasi-linear segments. This way, early evaluated points are more concerned with small details mainly depending on energy and pitch interactions, while lately evaluated points are more related with the description of larger prosodic events. Because of this distinction, it is not necessary to retain the fine details produced by the early backtracking steps up to the points controlling medium/long range pitch movements. Since the changes introduced by removing these points become very evident by delaying their evaluation to the latest steps of the backtracking process, the influence of the fine details on the decision process is not relevant. We therefore modified Equation 1 so that it does not keep track anymore of the preceding stylization steps obtaining the new formulation

$$\begin{gathered} D([s_i, s_{i+1}], [\bar{s}_i, \bar{s}_{i+1}]) = \\ \Gamma_g([s_i, s_{i+1}]) - \Gamma_g([\bar{s}_i, \bar{s}_{i+1}]) \end{gathered} \qquad (5)$$

Concerning the cost measure, in [9] we used the sigmoid transformation of the ratio between the number of points used by the stylization and the number of points used by the original curve so that [9, p. 1994] values of the cost measure at one end of the scale would not have been very different. As the impact revealed itself to be negative with respect to the evaluation of the quality/cost balance, we now consider the untransformed ratio represented as x in Equation 3 as cost measure.

5. Test material

For the presented evaluations we employed the 382 files of the prominence annotated TIMIT subset used in [14] to test automatic methods for prominence detection. This dataset was chosen for the curves cost evaluation we presented in [9] because we needed a large set of prominence annotated speech samples to evaluate the impact of using prominence information in a pitch stylization task. The same dataset was used for the cost related tests in [10]. In this work, we use the same dataset for qualitative, other than cost, evaluation because our goal is to check that the new approach does not introduce detectable changes on a large scale as we are interested in recovering only the details the OpS algorithm was missing. The dataset consists of 382 files containing 20 minutes of read speech extracted from the TIMIT corpus.

6. Results

From the quantitative point of view, we considered the number of points used by the SOpS algorithm with respect to OpS. The SOpS algorithm, on the considered dataset, uses 3.46 points per second (Pps) while the OpS algorithm uses 3.59 Pps. Table 1 shows a summary of the cost test between OpS, SOpS and an older version of the OpS algorithm employing manual prominence annotation called OpSProm [9].

Table 1: Cost test results.

	OpS	SOpS	OpSProm
Points per second	3.59	3.46	3.47
Total points	4118	4007	4009

A paired t-test indicated that the difference in Pps between OpS and SOpS is not statistically significant ($p > 0.01$). However, close inspection of the pitch curves where the OpS algorithm introduced more points than necessary showed that the SOpS algorithm does not suffer from this problem. The amount of reduction observed (0.13 Pps) and the actual p-value (0.012) are coherent with the goal we had of reducing the number of points used only in specific areas. The performance of the SOpS algorithm in terms of Pps is much more similar to the one we obtained with the OpSProm algorithm. A paired t-test between the Pps measures obtained by SOpS and OpSProm confirms this ($p > 0.9$) with greater certainty with respect to the result we presented in [10], where we stated (p. 205) that the difference between OpS and OpSProm, while not significant ($p > 0.01$), was to be taken carefully as the actual p-value was 0.0142.

From the qualitative point of view, a Wilcoxon test on the differences between curves generated by the two algorithms showed that the location shift is not significant ($p > 0.4$). The size of the considered dataset makes it safe to assume that no significant differences can be found between the curves proposed by the two algorithms on a large scale. This result confirms that the modifications introduced by the SOpS algorithm do not alter the stylized curve up to a statistically detectable degree. Close inspection of the cases on which the new model is intended to perform better, however, show that the details the OpS algorithm was not able to retain are correctly modeled by the SOpS algorithm.

7. Case study

In Figure 1, we show the detail of a pitch contour, the stylization proposed by the OpS algorithm (dashed line) and and the alternative proposed by the SOpS algorithm (dotted line) along with the energy profile. While the two algorithms perform identically on the first movement, the final rise/fall sequence is described differently. Since the curve's portion after the peak is much shorter than the rest of the curve, protecting the final lowering movement was considered not valuable enough by the OpS algorithm. This decision is encouraged by the tonal perception model as the rising movement preceding the final fall is synchronized with a rising energy profile, thus lowering the modeled glissando perception capability. The influence of sections that do not depend by the point being evaluated also plays

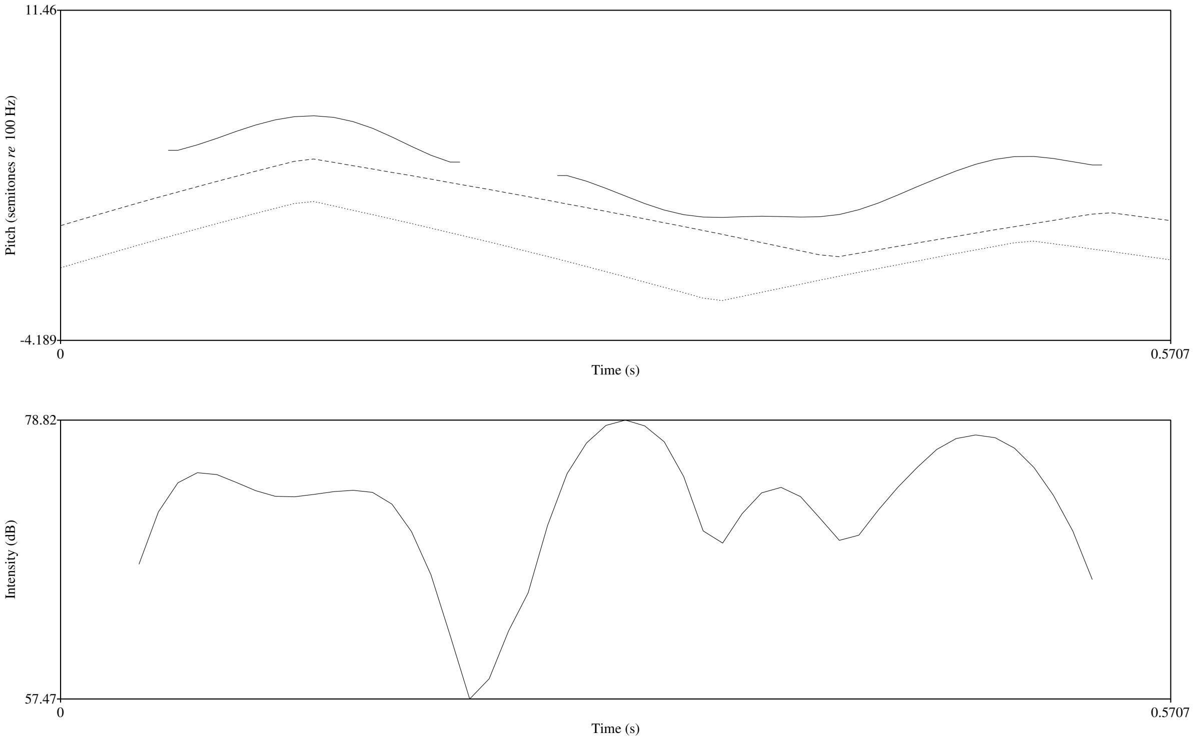

Figure 1: A pitch contour (solid line) along with the OpS stylization (dashed line) and the SOpS one (dotted line). The stylized curves are shifted by 2 Semitones each with respect to the original one for visualization purposes. Along with the pitch curve, the energy profile of the considered speech fragment is shown.

a role, as discussed before. The SOpS algorithm, by considering only the neighboring subcurves and by weighting them equally, is able to protect the final movement when evaluating the peak point, as expected because of the synchronized falling energy contour. The turning point before the rise is shifted 60ms earlier because of the segmentation strategy giving more importance to local minima. This improves the representation of the subcurve synchronized with the falling energy movement. The following pitch rise, synchronized with a rising energy contour, is more stylized than before, so no points are added. From perceptual inspection, this choice appears to improve the overall quality of the curve used in the example. The audio files of the original utterance from which the provided example is extracted are attached to this paper together with the resynthesis obtained with the OpS and SOpS curves. The magnitude of the changes the SOpS algorithm introduces with respect to the OpS curves are, in general, similar to the ones shown in the example. This explains why the similarity test based on statistical closeness is not able to detect a significant difference between the two algorithms. Being these changes important for an expert listener, however, we are in line with out observation that statistical closeness measures are not good estimators of a stylized curve's quality [9].

8. Conclusions and future work

We have presented the SOpS algorithm, an evolution of the OpS algorithm achieving better precision in stylizing specific details of the pitch curve that are important for an expert's ear. The SOpS algorithm is based on a simplified version of the dynamic tonal perception model used by the OpS algorithm. While the curves produced by the SOpS algorithms do not differ in a statistically relevant way from the original OpS curves in a qualitative and quantitative sense, we have shown that close inspection of the details we were interested in recovering are correctly represented by the SOpS stylization, thus obtaining a better representation of the *perceived* intonational profile that can be used in prosodic research. Concerning the number of control points used, we have shown that the new algorithm obtains results more similar to the ones we reported by using manual prominence annotations without altering the produced curves in a significant way. Therefore, the perceptual model behind the SOpS algorithm produces representations of the pitch curve that are both more precise and essential than the ones produced by its predecessor on fine intonational details. The simplified version of the dynamic tonal perception model will make it easier, in the future, to introduce the full range of changes indicated by the SCH, as we are currently considering energy only. The SOpS algorithm is implemented as a Python module of the Prosomarker tool, which is freely available for research purposes.

9. Acknowledgements

This work was supported by the European Community, within the FP7 SHERPA IP #600958 project. We would like to thank all the people who used the Prosomarker tool and gave us the necessary feedback to improve the OpS algorithm.

10. References

[1] J. t'Hart, R. Collier, and A. Cohen, *A Perceptual Study of Intonation: An Experimental-Phonetic Approach*. Cambridge: Cambridge University Press, 1990.

[2] D. Hirst and R. Espesser, "Automatic modelling of fundamental frequency using a quadratic spline function." *Travaux de l'Institut de Phonetique d'Aix-en-Provence*, vol. 15, pp. 75–85, 1993.

[3] C. D'Alessandro and P. Mertens, "Automatic pitch contour stylization using a model of tonal perception," *Computer Speech and Language*, vol. 9, no. 3, pp. 257–288, 1995.

[4] P. Mertens, "The Prosogram: Semi-Automatic Transcription of Prosody Based on a Tonal Perception Model," in *Proc. of Speech Prosody*, 2004.

[5] M. Wypych, "Automatic pitch stylization enhanced by top-down processing," in *Proc. of Speech Prosody [Online]*, 2006.

[6] S. Ravuri and D. Ellis, "Stylization of pitch with syllable-based linear segments," in *Proc. of ICASSP*, 2008, pp. 3985–3988.

[7] P. K. Ghosh and S. Narayanan, "Pitch Contour Stylization Using an Optimal Piecewise Polynomial Approximation," *IEEE Signal Processing Letters*, vol. 16, no. 9, pp. 810–813, 2009.

[8] D. Wang and S. Narayanan, "Piecewise linear stylization of pitch via wavelet analysis," in *Proc. of the European Conference on Speech Communication and Technology*, 2005, pp. 1–4.

[9] A. Origlia, G. Abete, C. Cutugno, I. Alfano, R. Savy, and B. Ludusan, "A divide et impera algorithm for optimal pitch stylization," in *Proc. of Interspeech*, 2011, pp. 1993–1996.

[10] A. Origlia, G. Abete, and F. Cutugno, "A dynamic tonal perception model for optimal pitch stylization," *Computer Speech and Language*, vol. 27, pp. 190–208, 2013.

[11] M. Rossi, "Interactions of intensity glides and frequency glissandos," *Language and Speech*, vol. 21, pp. 384–394, 1972.

[12] D. House, "Differential perception of tonal contours through the syllable," in *Proc. of ICSLP*, 1996, pp. 2048–2051.

[13] A. Origlia and I. Alfano, "Prosomarker: a prosodic analysis tool based on optimal pitch stylization and automatic syllabification," in *Proc. of LREC-2012*, 2012, pp. 997–1002.

[14] F. Tamburini and P. Wagner, "On automatic prominence detection for german," in *Proc. of Interspeech*, 2007, pp. 1809–1812.