THE INSTITUTE OF ELECTRICAL AND ELECTRONICS ENGINEERS INC.

SIGNAL PROCESSING
ALGORITHMS, ARCHITECTURES, ARRANGEMENTS, AND APPLICATIONS
SPA 2015
September 23-25th, 2015, Poznań, POLAND

Representing The Evolving Temporal Envelope of Musical Instruments Sounds Using Computer Vision Methods

Cong Yang and Marcin Grzegorzek
Research Group of Pattern of Recognition
University of Siegen
Hoelderlinstr. 3, D-57076 Siegen, Germany
Email: {cong.yang, marcin.grzegorzek}@uni-siegen.de

Ewa Łukasik
Institute of Computing Science
Poznan University of Technology
Piotrowo 2, 60-965 Poznan, Poland
Email: ewa.lukasik@cs.put.poznan.pl

***Abstract*—This paper proposes application of shape retrieval method developed and used in the domain of Computer Vision to describe and to match the long-term temporal envelope of musical instruments sounds that are to be compared. To effectively describe each envelope, we employ the skeleton descriptor, namely Audio Skeleton, to integrate both geometrical and topological envelope features. Based on skeletons, the audio envelope matching can be substituted by searching for the correspondences of skeleton endpoints. Finally, the similarity of audio envelopes is calculated based on their correlated skeleton matching. Our main contributions include (i) the introduction of a novel audio envelope descriptor with skeleton and (ii) the efficient and fast audio skeleton pruning and matching algorithms. Our method is validated through the skeleton matching and audio retrieval experiments on AMATI violin sound dataset.**

***Index Terms*—Violin Sound Analysis, Audio Envelope, Skeletonisation, Skeleton Graph Matching**

I. INTRODUCTION

Audio signal is represented in the time domain as a waveform, i.e. as a shape of the change of the air pressure as recorded from the microphone or generated synthetically. The waveform has a fine temporal structure (TFS) changing rapidly and a slower varying part called the envelope (E) [1]. The envelope is often regarded as a signal modulating the TFS component (carrier).

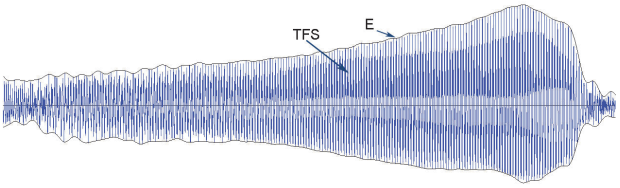


Fig. 1: Temporal fine structure (TFS) and Envelope (E).

The envelope of a signal is broadly defined as slow changes in time of the signal, whereas the temporal fine structure (TFS) is associated with its fast changes in time and may be treated as the carrier wave(s) of the signal [2]. The carrier signal reflects such parameters as spectral components and possible periodicity of a signal (Figure 1). It is known from Helmholtz time that the way the waveform sounds-its timbre-depends on its spectrotemporal characteristics, i.e. the timbre of a specific waveform depends on both-spectral components and the envelope [3]. In speech, envelope appeared to be most important for intelligibility (e.g. [4]). What is more timbre roles in the perception of speech, music and environment sounds are very similar [5]. Therefore the temporal envelope is similarly important for those three kinds of signals. A meaningful observation has been made during the posititron emission tomography experiment performed to examine the response of human auditory cortex to spectral and temporal variation of a sound [6]. Results indicated that responses to the temporal and spectral features are placed in very distinct places: temporal - towards the left, while spectral - towards the right hemispheres. This specialization in human brain responses to the sound spectrotemporal characteristics acknowledges the necessity of analyzing both - temporal and spectral structure of the sound in the context of their timbre.

To describe the evolution of the temporal envelope of individual sound, its segmentation into meaningful parts has been employed. The first to do that was Helmholtz, who proposed three elements: the attack, the steady state and the decay. Now the isolated sound is most commonly described by the ADSR model (attack, decay, sustain and release) introduced by Moog for his synthesizer [3] and much effort has been done to precisely describe the boundaries of those regions.

Many methods have been proposed to determine an envelope of a signal. Good reviews of existing methods have been given in [7] and in [8]. The problems of comparing those methods come from the fact, that there is neither an explicit mathematical definition, nor clear physical meaning of envelope [9]. The question is often posed how to describe the evolution of the temporal envelope of a musical tone leading further to its segmentation. One of the approaches is based on examining the relationship between the amplitude envelope and the spectral centroid (ACT) [3]. Peeters noted [10] that the ADSR envelope is hardly possible to describe the most of natural sounds (like in Figure 2 representing a violin vibrato sound) and proposed to segment the sounds into the region of attack and the rest. It might be the case for not-isolated sounds, e.g. in the series of sounds played. One of the interesting methods is called the *weakest effort* method and is based on an indirect measurement of the changes in the slope of the amplitude envelope. Glover has introduced a method of real time segmentation of individual musical tones [11]. Peeters et al. [10] collected several temporal audio descriptors related to envelope: attack, decay, release, log-attack time, attack slope, decrease slope, temporal centroid, effective duration, frequency of energy modulation, amplitude of energy modulation and RMS Energy Envelope. All above mentioned methods of sound envelope representation in fact concern the description of their shapes. Those shapes are useful for sound synthesis, sound morphing and also for sound retrieval.

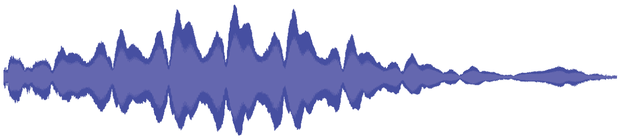


Fig. 2: Envelope of violin vibrato sound.

We propose a completely new approach to the problem of sound temporal envelope representation, using the method developed in Computer Vision for visual objects retrieval according to their shapes [12], [13]. We propose to use the skeleton based method to describe the two sided evolving envelope of a sound or a series of sounds in musical excerpts. This will enable to find similarities of the audio envelope shapes in the time domain.

One shape-based method for audio analysis has been proposed in [14] for heart signal. Meanwhile, the audio envelope in time domain is an important feature carrier for audio shapes and little attention has been paid to it. In this paper, we examine the relationship between the shape of audio envelope for violin sound.

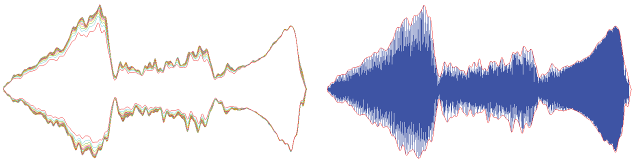
Posing as the general objective of the research to investigate the content based information on violin quality retrieval, we concentrate in this paper on calculating the similarity between the shape of amplitude envelopes over the violin music excerpts. In order to efficiently represent the shape of audio envelopes, we propose to use skeleton, namely audio skeleton, as the descriptor for computing the shape similarity, since skeleton integrates both geometrical and topological features of the shape. We also propose an efficient and fast matching algorithm based on the context features of audio skeleton endpoints. Our experiments for violin sound analysis are performed on AMATI [15] database.

II. Audio Representation

In this section, we first describe the method that generates the audio shape based on its envelopes. After that, an audio shape descriptor, namely audio skeleton, is introduced. Finally, we construct the audio descriptor with the skeleton endpoints.

A. Amplitude Envelope in the Time Domain

In this part, for a giving audio excerpt, we generate the audio envelope based on its waveform. A waveform is a time domain representation of a signal showing how its instantaneous amplitude varies over time. Amplitude envelope in the time domain is the line representing the evolution of the maximum amplitude of a waveform over time. It is a smooth curve outlining waveform extremes.



(a) Envelopes (different colours) (b) Waveform and selected envelope

Fig. 3: Iterative generation and selection of an audio envelope.

In this paper the method of creating an amplitude envelope in the time domain is a slight modification [16] of the approach, called the True Amplitude Envelope (TAE), proposed by Caetano and Rodet [8] for the rectified waveform. This method ensures that the curve representing the envelope is smooth during rather stable regions of the waveform and is able to react to sudden changes of amplitude.

Audio shapes analysed in this paper have to be represented by both positive and negative audio envelopes in time domain. They are calculated separately and then combined to get the full shape of dual side envelope. To make the signal more compact horizontally, a downscaling by a coefficient η (zooming out) is necessary (from η samples the maximum amplitude is retained). Then the low-pass filter is applied. A cut-off frequency controls the smoothness of the envelope. In order to outline the extremes of the waveform, the following iterative procedure is performed (presented in a pseudocode):

Algorithm 1 Envolop Calculation

```
1: Let $Y$ be input vector of signal $X$
2: Let $i = 0$
3: Run FFT on $Y$ to create spectrum
4: Leave only first $s$ values
5: Run inverse FFT to go back to time domain
6: Let $y[n]$ = Max$(y[n], x[n])$
7: Go back to 3, with $i = i+1$ and $s = s+1$ or end algorithm
   at desired $s$.
```

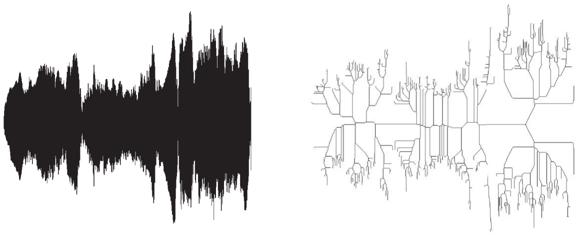
The smoothness and number of iterations are interdependent. The bigger number of iterations, the better the envelope matches the amplitude peaks of the signal. Figure 3(a) represents a set of audio amplitude envelopes of a fragment of the violin music calculated for 50 iterations at a smoothness s, while Figure 3(b) presents the signal and its envelope.

B. Audio Skeleton

A skeleton can be defined as a connected set of medial lines along the limbs of a shape [12]. For a given shape A, its skeleton $S(A)$ can be generated by the Maximum Disc method [13] that is continuous collection of centre points of maximal tangent disks that touch the object boundary in two or more locations. In order to regulate and standardise our next descriptions, based on the introduction in [17], we have the following definition:

Definition 1. *A skeleton point having only one adjacent point is an endpoint (the skeleton endpoint); a skeleton point having three or more adjacent points is a junction point. If a skeleton point is not an endpoint or a junction point, it is called a connection point. The sequence of connection points between two directly connected skeleton points is called a skeleton branch.*

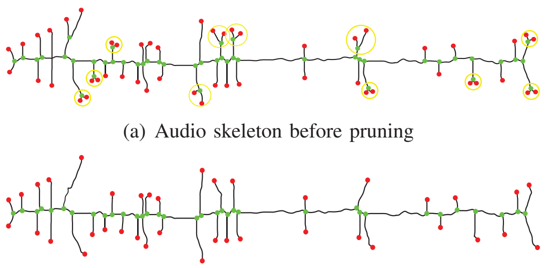
Due to skeleton's sensitivity to an object's boundary deformation, little noise or a variation of the boundary often generates redundant skeleton branches. These branches may seriously disturb the topology of the skeleton's graph. The most common approaches to overcome skeleton instability are based on skeleton pruning. Therefore, we introduce a set of restrictions for skeleton pruning. The basic idea is to remove the short branches in which their direct connected junction points are not in the middle area of the skeleton graph. More specifically, for a skeleton graph $S(A)$ and its junction points J and endpoints E, the horizontal H and vertical V borders of middle area are calculated by



(a) Original shape

(b) Original skeleton

Fig. 4: The skeleton in (b) has many redundant branches. This is mainly because of the boundary noise of shape (a).



(a) Audio skeleton before pruning

(b) Audio skeleton after pruning

Fig. 5: Skeleton pruning based on the locations of junction points.

$$\begin{aligned} h_l &= \min(J_h), & h_r &= \max(J_h) \\ v_t &= \text{mean}(J_v) - \delta, & v_b &= \text{mean}(J_v) + \delta \end{aligned} \tag{1}$$

where h_l and h_r are the left and right borders, $H = [h_l, h_r]$. v_t and v_b are the top and bottom borders, $V = [v_t, v_b]$. J_h and J_v denote the horizontal and vertical locations of all junction points, respectively. δ is the parameter for controlling the range of vertical borders. The δ is chosen if the ratio between the number of renewed junction points in the range of vertical borders and the increment of δ is maximum.

Based on the middle area, the junction points J can be divided into two groups: inside the middle region (J') and outside the middle region ($J^\star$). $J' \subseteq J$, $J^\star \subseteq J$ and $J' \cap J^\star = \varnothing$. For each junction point $j^\star$ in $J^\star$, all paths $(j^\star, e)$ are removed where e is the endpoint in E and there is another point e' in E with $length(j', e') > length(j', e)$. The pruned skeleton is shown in Figure 5(b). The non-pruned endpoints (red dots) and junction points (green dots) are denoted by E' and J', respectively. All branches that directly connect the endpoints and junction points are denoted by B. As illustrated in Figure 5(b), for each endpoint e_i, there is one branch b_i that directly connect to a junction point j_m. $e_i \in E'$, $b_i \in B$ $(i = 1, \cdots, N)$ and $j_m \in J'$, $(m = 1, \cdots, K)$.

Based on the pruned skeleton, we finally represent the audio features by its skeleton endpoints. More specifically, for each endpoints e_i in E', we describe its geometrical and topological features by a six-dimensional vector $\boldsymbol{e}_i$

$$\boldsymbol{e}_i = [L(b_i), l_i, L(e_i), L'(e_i), \Theta(e_i), \Theta'(e_i)] \tag{2}$$

in which $L(b_i)$ represents the length of an associated branch with e_i. l_i denotes the horizontal distance from e_i to the leftmost junction point. $L(e_i)$ and $L'(e_i)$ denote the mean distances from e_i to each endpoints and junction points, respectively. These two features are calculated by the mean Euclidean distance in the log space:

$$\begin{aligned} L(e_i) &= \frac{1}{N}\sum_{r=1}^{N} \log(1 + \frac{\|\overrightarrow{e_i} - \overrightarrow{e_r}\|^2}{f_{nor}}) \\ L'(e_i) &= \frac{1}{K}\sum_{m=1}^{K} \log(1 + \frac{\|\overrightarrow{e_i} - \overrightarrow{j_m}\|^2}{f_{nor}}) \end{aligned} \tag{3}$$

where f_{nor} indicates the normalisation factor to ensure our proposed features are scale invariant, $f_{nor} = \text{abs}(h_r - h_l)$ [1]. We also normalise the $L(b_i)$ and l_i by their ratios with the mean branch lengths. In order to avoid the situation where the input of log is zero, we add one to Euclidean distance. The rest two features $\Theta(e_i)$ and $\Theta'(e_i)$ present the mean pairwise orientations of vectors from e_i to each endpoints and junction points, respectively. These two features are calculated by

$$\begin{aligned} \Theta(e_i) &= \frac{1}{N}\sum_{r=1}^{N} \text{atan2}(\overrightarrow{e_i} - \overrightarrow{e_r}) \\ \Theta'(e_i) &= \frac{1}{K}\sum_{m=1}^{K} \text{atan2}(\overrightarrow{e_i} - \overrightarrow{j_m}) \end{aligned} \tag{4}$$

[1] In most experiments, $f_{nor} = 1$ since the scale of skeleton branches represent the audio intensity which is also considered for audio evaluation.

where atan2 stands for the four quadrant inverse tangent which can ensure $\Theta(e_i), \Theta'(e_i) \in [-\pi, \pi]$. [2] Eventually, based on our method, for a given audio, its shape A can be represented with the feature vectors of skeleton endpoints $\boldsymbol{e}_i$.

III. Audio Matching

Let E_1 and E_2 denote the set of skeleton endpoints from two shapes A_1 and A_2 respectively. $\boldsymbol{e}_i$ and $\boldsymbol{e}'_r$ denote a single endpoint in E_1 and E_2 respectively. Therefore, $\boldsymbol{e}_i = [L(b_i), l_i, L(e_i), L'(e_i), \Theta(e_i), \Theta'(e_i)]$ and $\boldsymbol{e}'_r = [L(b'_r), l'_r, L(e'_r), L'(e'_r), \Theta(e'_r), \Theta'(e'_r)]$. Let the numbers of endpoints in E_1 and E_2 be N and H, respectively, and $N \leqslant H$. Here we only use the length of associated branch and horizontal distance features for searching correspondence to fully use the properties of audio skeleton and reduce the computational complexity for matching.

More specifically, we first divide each endpoint set E_1 and E_2 into two groups, up group E_1^{up}, E_2^{up} and under group E_1^{under}, E_2^{under}, based on whether the vertical location of an endpoint is smaller or bigger than the mean vertical values of skeleton junction points $mean(J_v)$, respectively. Consequently, $E_1^{up} = \{\boldsymbol{e}_1, \cdots, \boldsymbol{e}_{N'}\}$, $E_1^{under} = E_1 - E_1^{up}$, $E_2^{up} = \{\boldsymbol{e}'_1, \cdots, \boldsymbol{e}'_{H'}\}$, $E_2^{under} = E_2 - E_2^{up}$. After that, we search for the correspondences of endpoints in two groups independently. As discussed above, skeleton branches and time series order should also be considered for search correspondences. Therefore, we use the associated branch length and horizontal distance features to calculate the dissimilarity $d(\boldsymbol{l}_i, \boldsymbol{l}'_r)$ between $\boldsymbol{l}_i$ and $\boldsymbol{l}'_r$ by their Euclidean distance. $\boldsymbol{l}_i = [L(b_i), l_i]$ and $\boldsymbol{l}'_r = [L(b'_r), l'_r]$. For a skeleton endpoint group E_1^{up} and E_2^{up} from two shapes A_1 and A_2, we compute all the dissimilarity between endpoints and obtain a matrix:

$$\boldsymbol{M}(E_1^{up}, E_2^{up}) = \begin{pmatrix} d(\boldsymbol{l}_1, \boldsymbol{l}'_1) & d(\boldsymbol{l}_1, \boldsymbol{l}'_2) & \dots & d(\boldsymbol{l}_1, \boldsymbol{l}'_{H'}) \\ d(\boldsymbol{l}_2, \boldsymbol{l}'_1) & d(\boldsymbol{l}_2, \boldsymbol{l}'_2) & \dots & d(\boldsymbol{l}_2, \boldsymbol{l}'_{H'}) \\ \vdots & \vdots & \vdots & \\ d(\boldsymbol{l}_{N'}, \boldsymbol{l}'_1) & d(\boldsymbol{l}_{N'}, \boldsymbol{l}'_2) & \dots & d(\boldsymbol{l}_{N'}, \boldsymbol{l}'_{H'}) \end{pmatrix} \tag{5}$$

Based on $\boldsymbol{M}(E_1^{up}, E_2^{up})$, we find the best matched points from E_1^{up} to E_2^{up} by Hungarian algorithm [18]. With the same method, for the group E_1^{under} and E_2^{under}, we also find the correspondences by applying the Hungarian algorithm on the dissimilarity matrix $\boldsymbol{M}(E_1^{under}, E_2^{under})$. The final correspondences between E_1 and E_2 is obtained by fusing the correspondences from up and under groups. These correspondences will be used for calculating the similarity between audio skeletons using their full features.

Next, we calculate the dissimilarity between E_1 and E_2 by all the distance and angle features of their corresponding skeleton endpoints above. More specifically, for a pair of corresponding endpoints $\boldsymbol{e}_i$ and $\boldsymbol{e}'_r$ ($\boldsymbol{e}_i \in E_1$ and $\boldsymbol{e}'_r \in E_2$), in order to ensure the numerical integration, we first split their original feature vectors into distance and angle sub-vectors: $\boldsymbol{e}_i$ is divided into $\boldsymbol{e}_{1i} = [L(b_i), l_i, L(e_i), L'(e_i)]$, $\boldsymbol{e}_{2i} = [\Theta(e_i), \Theta'(e_i)]$ and $\boldsymbol{e}'_r$ is divided into $\boldsymbol{e}'_{1r} = [L(b'_r), l'_r, L(e'_r), L'(e'_r)]$, $\boldsymbol{e}'_{2r} = [\Theta(e'_r), \Theta'(e'_r)]$. After that, we calculate the dissimilarity between $\boldsymbol{e}_i$ and $\boldsymbol{e}'_r$ by two parts:

$$d(\boldsymbol{e}_i, \boldsymbol{e}'_r) = d_1(\boldsymbol{e}_{1i}, \boldsymbol{e}'_{1r}) + d_2(\boldsymbol{e}_{2i}, \boldsymbol{e}'_{2r}) \tag{6}$$

Particularly, $d_1(\boldsymbol{e}_{1i}, \boldsymbol{e}'_{1r})$ is calculated by the correlation between distance features $\boldsymbol{e}_{1i}$ and $\boldsymbol{e}'_{1r}$:

$$d_1(\boldsymbol{e}_{1i}, \boldsymbol{e}'_{1r}) = \frac{\frac{1}{4}\sum_{z=1}^{4}(\boldsymbol{e}_{1i,z}\boldsymbol{e}'_{1r,z} - \mu_{\boldsymbol{e}_{1i}}\mu_{\boldsymbol{e}'_{1r}})^2}{\sigma_{\boldsymbol{e}_{1i}}\sigma_{\boldsymbol{e}'_{1r}}} \tag{7}$$

where $\mu_{\boldsymbol{e}_{1i}}$ and $\mu_{\boldsymbol{e}'_{1r}}$ are the means of vector $\boldsymbol{e}_{1i}$ and $\boldsymbol{e}'_{1r}$ respectively, and $\sigma_{\boldsymbol{e}_{1i}}$ and $\sigma_{\boldsymbol{e}'_{1r}}$ are the standard deviations of vector $\boldsymbol{e}_{1i}$ and $\boldsymbol{e}'_{1r}$. $d_2(\boldsymbol{e}_{2i}, \boldsymbol{e}'_{2r})$ is calculated by the mean difference between the angle features $\boldsymbol{e}_{2i}$ and $\boldsymbol{e}'_{2r}$

$$d_2(\boldsymbol{e}_{2i}, \boldsymbol{e}'_{2r}) = \frac{\frac{1}{2}\sum_{z=1}^{2}(\boldsymbol{e}_{2i,z} - \boldsymbol{e}'_{2r,z})^2}{\lambda^2} \tag{8}$$

where λ represents the tolerance to ensure the numerical integration with $d_1(\boldsymbol{e}_{1i}, \boldsymbol{e}'_{1r})$. In our experiment, $\lambda = 10$.

Finally, the resulting dissimilarity values of the matched endpoints can be denoted as $(d_1, d_2, \cdots, d_N)$ and the global dissimilarity $d(E_1, E_2)$ is calculated by the mean of dissimilarity values between the matched endpoints.

IV. Experiment

To evaluate the recognition performance of our method, we implement the audio shape retrieval on the AMATI dataset. AMATI dataset is composed of the recorded violin sounds during the 10th International Henryk Wieniawski Violinmaking Competition in Poznan (2001). It contains 52 violin sound played by the same violinist. The analysis has been performed on 5s excerpt of J.S. Bach Partita no 2 in D minor, Sarabande (BWV 1004). In order to compute the similarity between each excerpt, we first generate their audio envelope with the smoothness coefficient $s = 19$ and iteration $i = 30$. After that, the skeleton generation and matching processes are applied based on our proposed methods. Figure 6 illustrates some audio shapes in AMATI dataset.

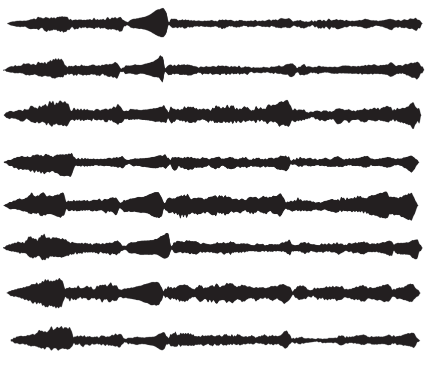


Fig. 6: Some of the audio shapes in the Violin-52 dataset.

During the experiment, we randomly select 10 shapes and use each of them as a query to perform the retrieval on the whole dataset. For each retrieval, the retrieved shapes are

[2] Since $\Theta(e_i)$ and $\Theta'(e_i)$ are angle values in $[-\pi, \pi]$, we do not have to normalise them.

ranked according to their similarity to the query and the top 10 shapes with the highest similarity are selected for analysis. Table I illustrates the retrieval results among ten queries. In order to evaluate the results by human perception, we employ two volunteers (one is familiar with the shape matching and another one is familiar with the violin sound evaluation) and ask them to check the retrieved results by considering the similarities of shapes. Based on their validations, the incorrect results are marked with bold fonts in Table I.

query	1	2	3	4	5	6	7	8	9	10	accuracy
10	84	58	**36**	20	40	118	113	**91**	43	**56**	70%
100	109	**93**	108	31	78	**15**	116	32	92	77	80%
20	**36**	**91**	84	58	**40**	113	118	10	35	79	70%
24	46	112	11	**85**	21	**104**	32	17	78	92	80%
32	78	92	**104**	21	46	**11**	112	116	24	**85**	70%
40	118	113	35	**36**	58	**91**	20	41	79	49	80%
74	**56**	**76**	**43**	**72**	**30**	**117**	**84**	**10**	**20**	**36**	0%
78	32	116	92	**104**	21	46	93	109	24	112	90%
85	46	112	24	**11**	17	21	**104**	92	32	80	80%
43	**56**	76	30	10	84	**74**	58	**36**	20	**91**	60%

TABLE I: Audio shape retrieval on AMATI dataset. Retrieval results are summarised as the ID of top 1-10 shapes which have the highest similarity to the query. The bold ID represents the incorrect result based on artificial validation.

As we can see all results seem to be incorrect for the query 74. A possible reason is that the query instrument 74 was considered to produce the outstanding sound by the jury during the violin competition. Mapping to its correlated audio shape, it is also difficult to find the similar shapes for the query 74. Moreover, all the similarity values of retrieved top 10 shapes to the query 74 are lower than 0.64, which is much smaller than the average value 0.8 among the rest queries. Therefore, it is reasonable to propose a threshold ψ to filter the retrieved results based on their similarity values to the query. If the similarity of a shape is below ψ, then it will not be considered for ranking. Another similar example for this situation is the query 43, which was considered to be one of the worst sound according to the jury's score. Consequently, as marked in Table I, it is challenging for the query 43 to find the similar shapes. If we set the threshold $\psi = 0.65$ then the average of accuracy for ten queries is 77.5%.

Additional experiment has been carried to address the stability of the skeletons while changing the time-scale of audio representation. The comparison of resultant skeletons led to the following conclusions. Audio Skeleton structure depends on the smoothness of the envelope. If the shape boundary is smoothed enough then the majority of small branches of the skeleton are removed and the main skeleton structure remains the same. For less smoothed envelope shapes the pruned skeleton remains stable with changing time-scale as the main skeleton branches are growing faster than the small branches (and vice versa).

V. Conclusion and Future Work

In this paper, a novel audio representation and matching method in the time domain is introduced. We represent a shape of audio temporal two-sided envelope with the audio skeleton which can preserve the time and amplitude equivalence by its junction points and endpoints. Consequently, the matching of audio envelopes is substituted by searching for the correspondences of skeleton endpoints. Experiment of matching envelopes of violin audio exerpts has been conducted. It illustrates the usability of audio skeleton for audio envelope matching. In the future, we will carry out more experiments to seek for the specific relations between the envelope shape and the violin sound qualities. Moreover, we will generate the more meaningful features for fully using the property of pairwise orientations.

Acknowledgment

Research activities leading to this work have been supported by the China Scholarship Council (CSC) and the German Research Foundation (DFG) within the Research Training Group 1564 (GRK 1564). We greatly thank Mateusz Kurek, Jakub Kedzierski, Aleksander Kedzierski, Magdalena Sroczan, Marek Sroczan, Eugeniusz M. Sroczan and Ioseb Otskheli from Poznan University of Technology for their significant help in our experiments.

References

[1] B. C. J. Moore, "The role of temporal fine structure processing in pitch perception, masking, and speech perception for normal-hearing and hearing impaired people," *J Assoc Res Otolaryngol.*, vol. 9, no. 4, pp. 399–406, 2008.

[2] R. D. P. Sondergaard and T. Dau, "On the relationship between multichannel envelope and temporal fine structure," in *3rd Int. Symp. on Auditory and Audiological Research*, 2012.

[3] J. B. M.Caetano and X. Rodet, "Automatic segmentation of the temporal evolution of isolated acoustic musical instrument sounds using spectrotemporal cues," in *DAFx-10*, 2010.

[4] R. Drullman, "Temporal envelope and fine structure cues for speech intelligibility," *J. Acoust. Soc. Am.*, vol. 97, no. 1, pp. 585–592, 1995.

[5] A. Houtsma, "Pitch and timbre: Definition, meaning and use," *J. New Music Research*, vol. 26, no. 2, pp. 104–115, 1997.

[6] R. Zattore and P. Belin, "Spectral and temporal processing in human auditory cortex," *Cerebral Cortex*, vol. 11, no. 10, pp. 946–953, 2001.

[7] A. M. A. Dabrowski and M. Portalski, "Comparison of methods for obtaining the envelope of acoustic signals," in *Proceedings of 7the Symposium New Trends in Audio*, 2000.

[8] M. Caetano and X. Rodet, "Improved estimation of the amplitude envelope of time-domain signals using true envelope cepstral smoothing," in *ICASSP*, 2011, pp. 4244–4247.

[9] Y. Z. F. H. M. Qinglin, Y. Meng, "An empirical envelope estimation algorithm," in *CISP 2013*, 2013.

[10] P. S. N. M. G. Peeters, B.L. Giordano and S. McAdams, "The timbre toolbox: Extracting audio descriptors from musical signals," *J. Acoust. Soc. Am.*, vol. 130, no. 5, pp. 2902–16, 2011.

[11] V. L. J. Glover and J. Timoney, "Real-time segmentation of the temporal evolution of musical sounds," in *Acoustics*, 2012.

[12] E. R. Davies, *Machine Vision: Theory, Algorithms, Practicalities*. Morgan Kaufmann Publishers Inc., 2004.

[13] R. Gonzalez and R. Woods, *Digital Image Processing*, 2nd ed. Boston, MA, USA: Addison-Wesley Longman Publishing Co., Inc., 2001.

[14] T. Syeda-Mahmood and F. Wang, "Shape-based retrieval of heart sounds for disease similarity detection," in *ECCV*, 2008, pp. 568–581.

[15] E. Lukasik, "AMATI - multimedia database of violin sounds," in *SMAC*, 2003, pp. 79–82.

[16] E. Lukasik, M. Kurek, J. Kedzierski, A. Kedzierski, and C. Yang, "Modification of the TAE calculation algorithm for violin sound analysis," Inst. of Comp. Sci, PUT, Tech. Rep. RB-1/15, 2015.

[17] X. Bai and L. Latecki, "Path similarity skeleton graph matching," *PAMI*, vol. 30, no. 7, pp. 1282–1292, 2008.

[18] H. Kuhn, "The hungarian method for the assignment problem," *Naval Research Logistics Quarterly*, vol. 2, pp. 83–97, 1955.