REAL-TIME DETECTION AND VISUALIZATION OF CLARINET BAD SOUNDS[1]

Aggelos Gkiokas, Kostas Perifanos, Stefanos Nikolaidis

Voice and Sound Technology Department,
Institute for Language and Speech Processing,
R.C. “Athena”
Athens, Greece
agkiokas@ilsp.gr, kperi@ilsp.gr,
snikol@ilsp.gr

ABSTRACT

This paper describes an approach on real-time performance visualization in the context of music education. A tool is described that produces sound visualizations during a student performance that are intuitively linked to common mistakes frequently observed in the performances of novice to intermediate students. The paper discusses the case of clarinet students. Nevertheless, the approach is also well suited for a wide range of wind or other instruments where similar mistakes are often encountered.

1. INTRODUCTION

The growing use of computer software for music teaching led research to a new evolving domain, Music Visualization. So far, this term has not been given a strict definition. Various approaches have been presented in the recent years in the literature regarding music visualization.

Hiraga [1] presented a system that visualizes a whole music piece. The proposed model reads music pieces from MIDI files, and no waveform analysis is made. This approach can be rather considered as an alternative way to visualize a music score. At the same year Hiraga proposed a more advanced system in [2], however with same properties as in [1]. McLeod and Wyvill in [3] developed a system that computes and visualizes the pitch of a music performance in real-time. The visualization scheme is a 2 dimensional graph, with time on the horizontal axis, and pitch on the vertical axis. The system was tested by an expert violin teacher and the feedback was promising. Because of the rapid development of computers, designing such a system in 2003 is a remarkable work. Toivinianen in [4] provided a system that visualizes the tonal content of a musical piece using SOMs. Ferguson [5] proposed a very interesting work on visualizing music performance in real-time. The developed system provides visualization of important acoustic features, closely related with sound quality, such as harmonic content, tuning discrepancy and noisiness. In [6] Spyridis, although apart from the scope of music education, proposed an interesting novel image-to-sound, and sound-to-image transformation.

Music visualization can serve different purposes in the context of music visualization. As an offline tool, it can offer students a way to examine different aspects of their performance *after* they are done playing a music piece. Such aspects can include information about their timing, rhythm, stability and overall quality. However, as a real-time tool, the visualization of the sound is displayed *during* the student performance. There are a few arguments fore and against providing feedback during a student performance. These mainly relate to the possible distraction of students by any feedback during their performance, as common goal in music education is to push the students to finish up despite any errors or mistakes. Nevertheless, real-time feedback is well motivated; not only because the students get immediate feedback and can promptly try to correct their performance, but also since it provides an effective learning metaphor with the system playing the role of a sonic mirror where the students perform and get immediate feeling of the effects of their performances.

This paper presents work carried out in the context of the VEMUS project. VEMUS (Virtual European Music School) is a project funded by the European Commission under the Information Society Technologies (IST) Programme of the Sixth Framework Programme (FP6). The VEMUS project started on October 2005 aiming to design, develop and evaluate an open, highly interactive, and networked multilingual music tuition framework for popular instruments and a set of innovative pedagogically-motivated e-learning components addressing different learning settings [7].

The aim of work described in this paper is to provide a real-time music visualization tool for clarinet sound in the context of music education. From the discussion above, it is clear that any feedback provided in real-time should be short and simple, avoiding to distract the students and helping them to go on despite any errors. Visual feedback should be coarse avoiding to provide too much detail or to require too much attention.

Some additional obvious requirements imposed to this tool. It must be user-friendly and engaging, inviting the students to practice more. Furthermore, the tool must help the students to gain a perception of their progress. as the time goes by. The real-time visualization of a tone must be something “alive”, providing

[1] This work has been carried out under the VEMUS project which has been partially supported by the European Community under the Information Society Technologies (IST) priority of the 6th Framework Programme for R & D.

meaningful and intuitive visual feedback that the student can comprehend and exploit. Additionally, the tool must be responsive, keeping the visual response time of the sound to a minimum. Nevertheless, such a system has to cope with the real-time processing requirements.

The system presented in this paper is tailored to students of beginning to intermediate level. To meet the real-time processing requirements, the system needed to operate only based on rather simple spectral features, such as pitch, RMS energy and the partials amplitudes. The performance error detection machine had to be kept simple, providing also a scalar measure of the sound quality.

Viewed at a more abstract level, the overall task of such a visualization system is to formulate a mapping from the universe of discourse of sound frames to a visual space, preserving as many as possible from the significant features of clarinet playing. For our system, a simple, time varying 3-dimensional object has been selected as the rendering model.

The rest of the paper is organized as follows. Section 2 describes the basic clarinet errors that are common for students of the target levels, along with some discussion on the spectral properties of these errors. Section 3 provides the overall system architecture and a brief description of each component. Section 4 describes the method used for detecting performance errors. Section 5 presents the proposed visual model and the way the sound quality is mapped to an image. Conclusion and directions for further work are provided in section 6.

2. CLARINET BAD SOUNDS

In a related work, Zlatintsi [8] presented a classification of "bad" clarinet notes. The separation of these classes was made by taking into account two criteria: the cause of a mistake (e.g. bad finger coordination, air leaking), and the resulting sound quality (e.g. squeaks, hollow notes etc). These classes are described in summary below:

Hollow notes: The main cause of a hollow note is the bad airflow in the clarinet. The main attribute of a hollow note is that the energy of the individual harmonics is lower than the normal. However this cannot be described explicitly. “Hollowness” is related somehow with timbre. As the students progress, their technique becomes better and the airflow is more stable, resulting to a “better” tone quality. This admission leads us to the conclusion that it is preferable to address hollowness as a scalar property, i.e. assigning each note a degree of hollowness, rather than crisply characterizing a note as hollow or not.

Squeak notes: The main cause of a squeak is saliva (into the clarinet or when the reed gets calcified) or when students press and bite the reed resulting no free vibrations. In a squeak note all the partials amplitudes become much higher than the normal. Clarinet becomes unstable and the physical model of the instrument can no longer describe its behavior. A graph of the harmonics amplitude over time for a good note, a hollow and a squeak note are shown in Figures 1,2 and 3 respectively.

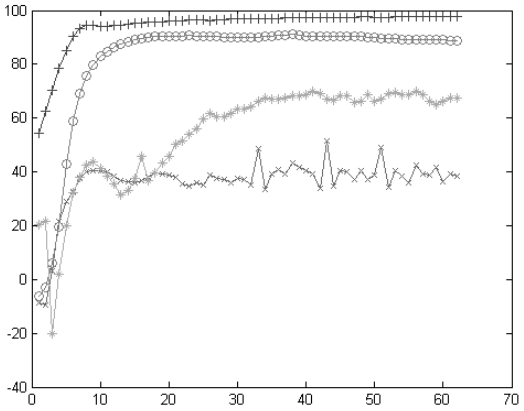

Figure 1: *The first four harmonics amplitudes over time for a good quality note. Index: “+”:1st, “x”: 2nd, “o”: 3rd, “*”: 4th. The same index will be used in Figures 2 and 3.*

Unstable notes: Unstable notes have many causes such as insufficient amount of airflow for the specific tone or not firm embouchure. Instability can be either pitch instability or RMS energy instability. Both can be easily detected by calculating the standard deviation of pitch and RMS-energy within a note (or part of note)

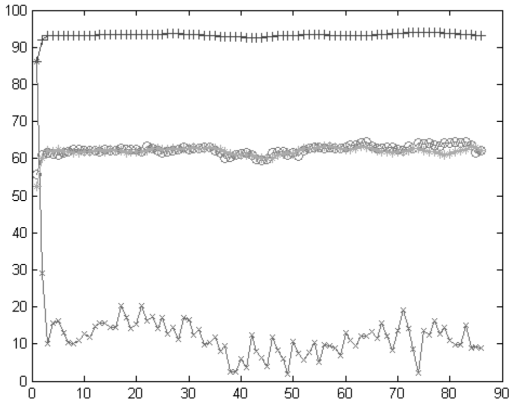

Figure 2: *The first four partials of a hollow note. We can easily observe the lower amplitude values of the partials.*

3. SYSTEM OVERVIEW

The individual components of the proposed system and the flow diagram are shown in Figure 4. The implementation of the system is parallel with all modules running concurrently. To ensure proper operation, appropriate memory sharing and event-driven synchronization mechanisms are built into the system.

Real-Time Audio Recognizer (RTAR) reads streamed data from the microphone as the student performs the musical piece. With a conventional front-end processing scheme RTRA process a window of 25 ms long every 10 ms with a 60% overlap. For every window it processes, RTAR writes output data to the Audio Buffer and sends a message to the synchronizer module that a new frame is processed. Output data consists of pitch, RMS energy, the partials up to the 6th and a single integer, representing the MIDI value of the tone played. The synchronizer activates the Error-Detection (ED) module. ED reads the Audio Buffer and computes the "Quality" or "Hollowness/Squakness" value as will be described in the next section. Every 4 iterations of this procedure (i.e. 40 ms) the synchronizer sends a message to the 2-Dimensional curve generator. The latter reads the sound quality value from the ED and produces the 2-dimensional curve. Finally the 2D curve is fed to the 3D-Curve Generator which draws the final shape. These rates are adjustable, to adapt visualization software to machines of different computing power.

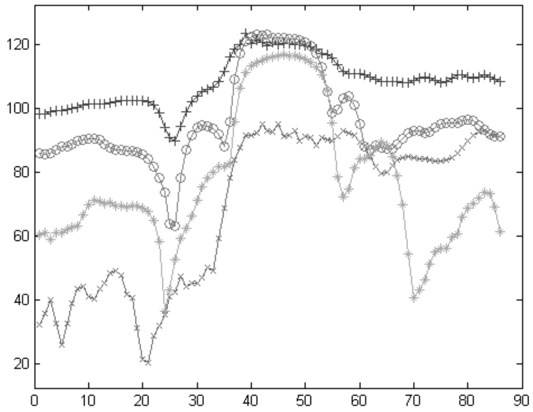

Figure 3: *A squeak note. The squeak sound appears at about 35th frame.*

4. BAD SOUND DETECTION

The intuition that a classic ML approach (labeling sounds as hollow, squeaks etc, training a machine, categorizing) would not succeed the desirable results led us to a different approach. In the case of a beginner, such a system would characterize many of the notes as hollow, or squeaks, in an objective way. This is not educationally desirable. Our system is based on the concept that a machine should be flexible on how it judges the students' performance. The teacher should be able to adjust the strictness of the module.

The limitation of the real-time processing led us to use only the partials extracted by the Audio Recognizer Tool, without any extra signal processing. Our approach is based on the fact that the mean values of the harmonics of a music piece performance can represent the clarinet's "sound". Any divergence from these values can be considered as a bad sound. Additionally, such a system can cope with the recording quality and the instrument condition.

4.1. Features Used

The basic features used for the detection of clarinet bad sounds are the partials of each frame up to the 6th. Specifically, because of the fact that the first harmonic is proportional to RMS energy, we used partial values from the 2nd to the 6th, divided by the amplitude of the first. Thus, mathematically, features used can be defined as

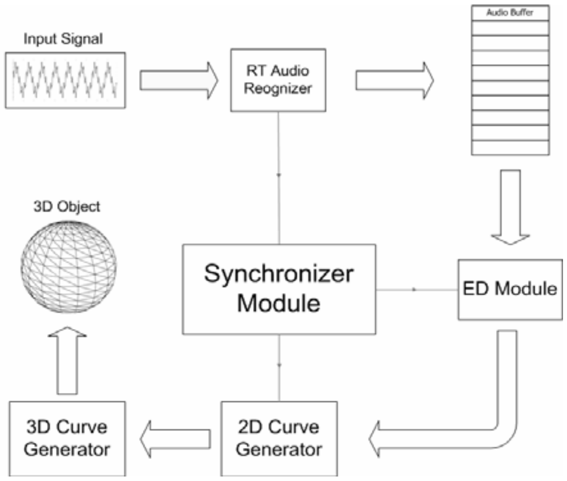

Figure 4: *The overall architecture of the deployed system*

$$f_i = \frac{h_{i+1}}{h_1}, i = 1..5 \quad (1)$$

where h_i denotes the ith partial amplitude.

4.2. Training

In the training phase we calculate the mean values and standard deviation of the features, in recordings where a clarinet teacher or professional player performed. From these values, we fit a Gaussian distribution to the training data. During this process, we measured that the relative values of the partial amplitudes depend strongly on the pitch, an observation accordant to the bibliography. Thus, for different pitch values, we fit a different Gaussian distribution to the relative partials. Specifically, for each individual musical tone (MIDI value), we train a different model. This process results *N* Gaussian distributions for each feature, a total of *5N.*

$$p_{ij} = p_{Gauss}(\mu_{ij}, \sigma^2{}_{ij}, f_j), i = 1..N, j = 1..5 \quad (2)$$

where index *i* is referred to the tone identity and *j* to the feature identity.

4.3. Error Detection

The error analysis presented in section 2 led us to the following admission. If the relative partial *j* is greater than the mean value calculated in the training phase for a specific tone, then contributes to the sound to be heard more squeak. Reversely if it's smaller, contributes to the sound to be heard hollower. The measure of this contribution is a quantity somewhat inverse proportional or decreasing to $p_{ij.}$ The final characterization will be a sum of these quantities.

We tested various such functions from simple linear combination, to more complex ones. We found that a sum of powers of $1\text{-}p_{ij}$ worked well enough. Formally, the final formula we used is the following:

$$Q = \sum_{i=1}^{5} [sng(f_j - \mu_{ij})(1 - p_{ij})^4] \tag{3}$$

The polynomial power of 4 is used to smooth small variations of $1\text{-}p_{ij}$ around zero. The sign function has the following role: when a feature value is smaller than the mean value μ calculated during the training process, the respective clause contributes to the sum a negative value. If it is greater, a positive value. Namely, if Q is positive, means that we probably have a squeak sound, if negative a hollow sound. Therefore, we managed to map sound quality to a one dimensional space, real values around zero. Closest to zero is Q, the better the sound is.

4.4. Time Averaging

As described in section 3, ED module computes as described above the quantity Q to characterize the sound quality of each frame. Because in time space, one visual frame corresponds to more acoustic frame, it is not desirable to characterize the sound between two subsequent visual frames from only one acoustic frame. Thus, we take the average value of Q between these two visual frames.

The extreme case, where between two visual frames exist both almost equal high valued squeak and hollow sounds, resulting an average Q close to zero is almost impossible.

4.5. Extensions to the Basic Algorithm

4.5.1. Onset Discarding

The sound modelling described before does not correspond to the case where the sound data processed is a part of the onset of a note. Onsets have very different statistical properties between the partials, thus it is ineffective to try characterizing such frames. In a very simple fashion, we discard onset frames, by ignoring the first frames of each note (depending on the frame rates).

4.5.2. Offset Discarding

The same stands for the offset of each note. Higher partials decay faster than lower. In contrast with onset discarding and because of the real-time nature of the problem, we do not have prior knowledge when a note will end. Therefore such discarding is impossible.

We handled this situation in terms of smoothing, as will be described in section 5.4. The basic idea is when we have decaying in RMS energy on the signal, implying the note ends; we limit Q from changing value greater than a certain ratio. This worked well enough.

4.6. Coping with the Different Level of Students

Visualizer module must have different behaviour in different levels of students. For the same waveform, produced by a beginner and an intermediate student, visualization feedback must be stricter for the latter. This can be easily adopted adding one more parameter to the model described in equation 2. We substitute standard deviation by a multiple of it. Equation 2 now can be written as

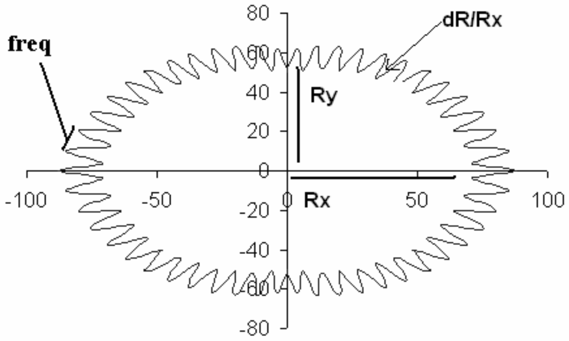

Figure 5: *The 2-Dimensional visual model and the four attributes used to change the shape.*

$$p_{ij} = p_{Gauss}(\mu_{ij}, a \cdot \sigma^2{}_{ij}, f_j), i = 1..N, j = 1..5 \tag{4}$$

Parameter α is global for all density components. The lesser α is, the more sensitive is system to mistakes. This extension allows the teacher by adjusting this value to personalize visualization according to the student level.

5. THE VISUAL MODEL

As described in the previous chapter a hollowness/squeakness value (Q) is fed to the Visualizer. The main idea is to represent a note as circle. This circle has four attributes to control (plus the color, a total five). These attributes can be shown in Figure 5. Changing the values according to the students performance produces a meaningful shape evolving over time.

Attribute *Ry/Rx* is controlling the capability of the shape to become more or less elliptic. When *Ry=Rx* the shape is a circle. On the surface of the shape a sinusoidal disturbance is added. The amplitude of the disturbance is controlled by the attribute *dR/Rx* (we use radius *Rx* as reference as the ratio *Ry/Rx* changes), and the frequency is labeled as freq. Finally the size of the shape is represented by *Rx*.

In the next sections we describe how the circle's attributes values depend on the errors made by the performer. The choice of this relationship is made using intuitive criteria, in accordance with clarinet teachers' opinions.

5.1. Visualizing a Squeak Frame

When a frame is classified as a squeak, the shape is drawn as "craggy" or "rough". As more squeak a frame is, the rougher the circle should be. The rules that determinate the circle's attributes values are the following.

1. $\frac{Ry}{Rx} = 1$
2. Attribute *freq* is high valued, and increases as squeakness increases.
3. $\frac{dR}{Rx}$ is proportional to squeakness

4. *R* is proportional to RMS energy of the frame

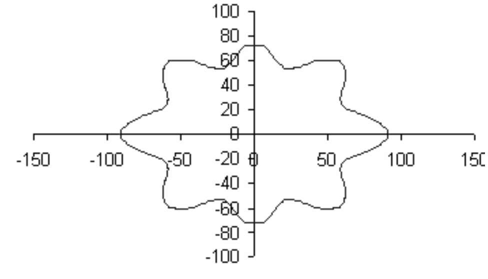

Figure 6: *The 2-Dimensional visual output for a hollow note*

5.2. Visualizing a Hollow Frame

A Hollow note is represented as a more “flabby”, “sleazy” shape, as shown in Figure 6. Attribute values in this case behave as

1. $\frac{Ry}{Rx}$ is decreasing as hollowness increases
2. Attribute *freq* is low valued and decreases as hollowness increases
3. $\frac{dR}{Rx}$ is proportional to hollowness
4. *R* is proportional to RMS energy

5.3. Pitch and RMS Instability

The RMS instability is directly related with the sphere shape, because of the proportional relationship between RMS energy and attribute *Rx*. Therefore an RMS unstable note is directly shown. The attributes’ values are selected as explained before, derived from the hollowness/squeakness value.
Pitch instability has not been yet explored. We have the intuition that relating the pitch instability with a light change of the color of the shape will result a meaningful message to the student.

5.4. Smoothing

As the hollowness/squeakness value evolve over time, sudden jumps of this value often occur. This fact results to rapid changes of the shape, making the view of the graphic annoying. To handle this problem, we deployed a smoothing on the final shape. Every attribute cannot change more than a fixed ratio between two consecutive visual frames. However this imports a tradeoff between a satisfactory and enjoyable viewing and visualizing quick, short-time errors.

In the case of offset discarding, when a consecutive decay of RMS energy is detected, the fixed ratio between the attributes values become grater.

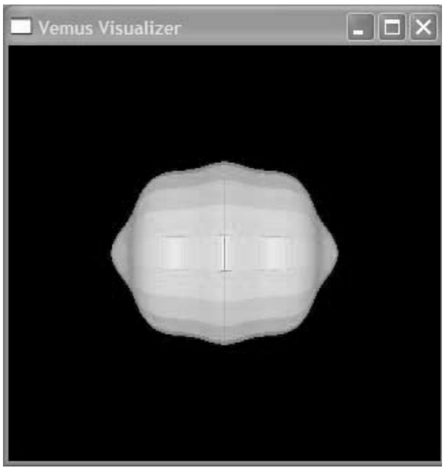

Figure 7: *The 3-Dimensional final output for a hollow note*

5.5. Transforming to the 3-Dimensional Object

The rendering system takes as input the 4th and 1st quadrant of the generated object, transforms this curve into an object by revolution, rotating the object around the x-axis by 180 degrees, in distinct steps. The number of steps (from a lower of 10 to a maximum of 180) used to create this object determines the quality of the generated object. More steps means that the object is smoother but this affects the rendering performance, since more arithmetic is required to generate the object itself and the normals described bellow. Since our module is real-time and targets to low-end machines, the performance of the implementation itself should be fast and effective. We actually can save some extra processing power, by using lookup tables for sine/cosine functions and avoiding unnecessary calculations when possible.

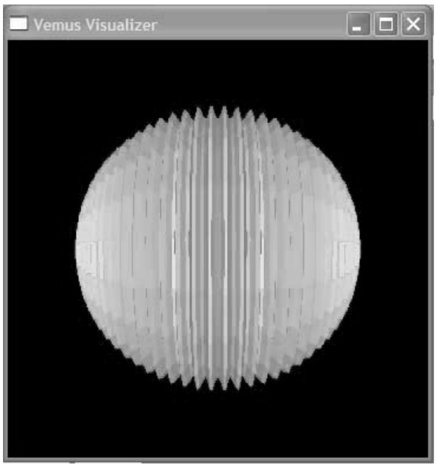

Figure 8: *The 3-Dimensional final output for a squeak note*

After transforming the 1stand 4th quadrants, we can easily get the other two by mirroring the object to the plane defined by x and y axes, to create the full symmetrical sphere.

Since we want to use lights and material features to our object, one more step is actually required, to compute the *normals* (i.e. the vectors perpendicular to the surfaces) for each triangle we are going to draw. After normals calculation, our object is ready to be drawn.

We are applying color, material and lighting and also (optionally) rotation to the object. The colors used in our implementation are chosen arbitrary, meaning that up to this point a survey for the parameters of the system is still pending. However, as mentioned before, there is the persuasion that associating changes in color with instability of pitch will result a meaningful output. The object is now ready for drawing as shown in Figures 7 and 8 (hollow and squeak note respectively).

6. CONCLUSION AND FURTHER WORK

In this paper, a tool has been presented that employs 3D graphics to provide real-time visualization of a beginner student performance in an educational context.

Students receive simple and intuitive visual feedback that can help them improve the quality of the sound they produce, while keeping the disruption of their practice to a minimum.

An important feature of the system is that it does not only provide a measure of how "good" or "bad" a performance is at a given time, but also visual feedback on different aspects of the performance (such as its hollowness or stability) integrated into a simple 3D object which offers an intuitive way of understanding *what* is wrong and, to a degree, what the student must do to correct it. This is a major concern in music education.

Real-time performance requirements impose limitations to the complexity of the calculations performed by the system. Nevertheless, the presented approach suggests a promising scope of view on how to visualize sound in a meaningful and useful way for music teaching.

The paper focuses on clarinet students. Nevertheless, the approach is also well suited for a wide range of wind or other instruments where similar mistakes are often encountered.

The architecture of our system separated two important subtasks; characterize and visualize a sound. This fact allows the development of different, more sophisticated techniques for both subtasks in an independent way. Sound quality, a term related to timbre, has been extensively explored in the past leading, however, to no explicit widely accepted definition. Our approach, although simple and straightforward, suggests an effective way to characterize a sound. Concerning the visual model, various different alternatives are also available to examine.

Some first feedback from music teachers has provided clear indications that the approach is well motivated in an educational context and that if appropriately integrated into a learning setting, it may help students gain better understanding of their errors. Further feedback from music teachers, but also from students, will be collected after the visualization tool has been integrated into the overall VEMUS platform and tested with users in realistic conditions.

7. ACKNOWLEDGEMENTS

We would like to thank Ballteck for its collaboration and supply of the Real-Time Audio Recognizer tool. We also would to thank Kjetil Falkenberg Hansen and Athanasia Zlatintsi for their helpful opinions during our discussions on the VEMUS project.

8. REFERENCES

[1] R. Hiraga, R.Mizaki and I. Fujishiro, *Performance Visualization: a new Challenge to Music Through Visualization*, Proceedings of the 10th ACM international conference on Multimedia, pp 239-242.

[2] R. Hiraga, F.Watanabe and I. Fujishiro, *Music Learning through Visualization*, Web Delivering of Music, 2002. WEDELMUSIC 2002 proceedings, pp 101-108.

[3] P. McLeod and G. Wyvill, *Visualization of Musical Pitch*, Computer Graphics International 2003 (CGI'03), p. 300.

[4] P. Toiviainen, *Visualization of Tonal Content with Self-Organizing Maps and Self-Similarity Matrices,* ACM Computers in Entertainment, 3(4), Article 3C, 2005.

[5] S. Ferguson, A. Vande Moere, D.Cabrera. *Seeing Sound :- Real-Time Sound Visualization in Visual Feedback Loops Used for Training Musicians,* Proceedings of the Ninth International Conference on Information Visualization (IV ’05), 2005, pp 97-102.

[6] C. Spyridis, A. Georgaki, G. Kouroupetroglou and C. Anagnostopoulou. *Image to Sound and Sound to Image Transform*. Proceedings of the 4th Sound and Music Computing Conference (SMC07)", 11-13 July 2007, Lefkada, Greece, pp 312-318

[7] www.vemus.org

[8] A. Zlatintsi, *When the Clarinet Sounds Bad – Identification Study*, Master of Science Thesis in Musical Acoustics, KTH, 2006.