Electronic Dance Music Analysis for Real-Time Synchronization of 3D Video Animations in Live Events

Emanuel Aguilera, Jose J. Lopez, Pablo Gutierrez-Parera and Carlos Hernandez

Instituto de Telecomunicaciones y Aplicaciones Multimedia,
Universitat Politècnica de València,
8G Building - access D - Camino de Vera s/n - 46022 Valencia (Spain)
Corresponding author: emagmar@iteam.upv.es

ABSTRACT

Video effects for Electronic Dance Music (EDM) events is an emerging and demanded but costly service for the industry. In this paper, a set of adapted and novel algorithms for automatic EDM feature extraction and segmentation are presented, suitable for the automation of video and 3D animation synchronized to the audio content. Features like frequency balance, color representation of the music, beat tracking and music segmentation have been specifically developed and tuned to EDM, taking into account a low computational cost. The segmentation algorithm has been trained with a database of 100 songs obtaining accurate event detection about 98% with low false alarm rate. It has been implemented in C performing in real-time and has been successfully integrated into a real-time commercial 3D graphics engine.

Keywords: Music analysis, beat detection, music segmentation, machine learning, electronic dance music.

1.- INTRODUCTION

Clubs, discos and music festivals are an important part of today's high income entertainment industry. Electronic dance music (EDM) is one of the most important styles according to number of venues and festivals dedicated to it. This audience expects a complete experience, so the music is always improved with lights and video projections as shown in Figure 1. The emerging work of video DJs does not yet have a mature software ecosystem for their work as the music DJ does, so there is still much room for improvement in this area.

Nowadays, the ability to provide compelling visual animation presentations during musical events in clubs, concert halls and other such venues, is restricted to large scale events promoters, artists, and mega-clubs. The main reason is that such visual animation must be designed in advance and specifically tailored for the music that will be played, increasing the budget. This work is focused on removing existing entry barriers and extending the use of this technology by developing affordable musical segmentation software to be used with a Real-Time 3D Motion Graphics for the Electronic Dance Music (EDM) industry.

One way of decreasing costs is to automate the visual animation presentations in real-time as the musical event is in session. That implies developing computational music analysis to extract the most relevant information needed for the visual content. This kind of analysis software has been employed to automatic classification of broadcast material [1] or musical databases [2], some of them also preserving real-time behavior [3, 4] and others are more difficult to implement in real-time [5, 6]. Apart from working in real-time, the software and its related algorithms developed in this work should employ low computational cost to meet the usage requirements of the EDM industry.

In this work we have identified three groups of music analysis features that are useful for

synchronizing video effects: those related to the fast evolution of signal energy, those related to the spectral composition of the signal, and those related to the song structure. The first group is in connection with the rhythm of the music and allows to link fast movements. The second group is related to slow changes in the signal as instruments change along the song. Finally, the third group of features appears only a few times during the song, when there are audible changes in the structure of the song and are more suitable to be linked with large changes in the video animation, such as topic, scene or camera view change. All of these groups of features, once linked with the visual animations have a powerful effect on the audience.

The rhythm of the music is also known as beat. In the literature many beat tracking algorithms have been proposed. Great advances have been reached and applied to different kinds of music with variable success. However, even state-of-the-art algorithms experience failures analyzing EDM music in real-time. For this reason, a specific algorithm tailored for EDM music has been developed in this work and is explained in section 2.

To follow slow changes in the composition of the music, different frequency balance and statistical parameters can be employed. Some statistical parameters as flatness, skewness or the kurtosis of the spectrum have been employed in the literature. However, in this work the effort has been put only in frequency balance parameters. After a thorough test, we found that they provide a better correlation with the perceptual sensation experienced by the listener and are therefore more suitable for video synchronization. Frequency band energy analysis has been employed in this work and it is explained in section 3 and 4.

Finally, regarding the structure of the song we have focused our efforts on the development of a segmentation system aimed at detecting changes in the song related to the different sections that it is composed by. Before going into segmentation details, a previous detailed study of the structure of EDM music is presented.

As commented before, section 2 describes the beat tracking algorithm developed for this work. Section 3 gives the mathematical details related to the frequency analysis of the signal in order to calculate energy level features by frequency bands. Section 4 describes the process of calculating a color representation of the music derived from previous parameters. A specific algorithm for EDM segmentation is described in section 5. Finally, section 6 summarizes the conclusions.

Figure 1: Lights and video projections in a club.

2.- BEAT ANALYSIS

Beat in musical language can be defined as a sequence of instants that might correspond to the taps that a human listener would do with his foot when listening to music. Many musical signal analysis applications require the accurate detection of the beat. In the current application, a stable and accurate detection of beat is mandatory, so that the video animation can be synchronized with the music.

The beat analysis usually consists of two parts: tempo estimation and onset determination of each individual beat pulse. The tempo is usually referred in EDM as beats per minute (BPM). In EDM and some other modern music genres, the tempo is likely to stay the same throughout the whole song, but sometimes can change, especially when the DJ mix two songs, a specific case that must be taken into account.

In general, beat tracking algorithms can be classified as offline [7] and online [8, 9]. The

first group has access to the whole song to label the beat instants, whereas the second group should work online, providing the beat instant in a causal way at the same time the song is reproduced. Obviously, the first group of algorithms usually obtain better results than the second group. Sometimes, algorithms prepared to work offline can be modified to work online with small changes, tricks or adding some latency. In this work the focus has been put in pure online algorithms that also exhibit low latency.

There are various difficulties when tracking the beats in real musical acoustic signals. The simple technique of peak-finding with a threshold is not enough, since there are many energy peaks that are not directly related to beats. The best way to discriminate the real beat from other peaks is to obtain the BPM in a first stage.

Convolution techniques have been successfully used to obtain BPM [9]. To implement that in a practical way, convolution is not applied directly to the sound samples. Instead, the signal is windowed and the energy of each window is computed. Then, the resulting vector of energies is used for convolution and BPM calculation. Better results can be obtained if the energy is calculated for different bands, for example Mel bands [7], then converted to dB scale and the first-order difference along time calculated in each band. The remaining positive differences are then summed up across all frequency bands. This technique has proven to be robust for any kind of music. In our work we follow this strategy, but instead of computing many different Mel bands, that usually require FFT computation, we employed only 3 bands, the same audio bands used for the frequency features described in section 3, greatly reducing the computational power required for this stage. These 3 bands strategy has been proven enough for EDM music, where percussion is present most of the time and always produces detectable changes in any of the 3 bands.

One problem that usually arises in BPM computation is that, depending on the type of rhythm, the maximums in the convolution may appear in the human-identifiable BPM or at twice this value. To solve this problem, we have limited the BPM to a range where most of the EDM song is found, thus avoiding the problem of double BPM, or even the half BPM if it appears.

The BPM is identified at the beginning of the song in a setup stage and is tracked throughout the whole playback. Different circumstances may arise during the song:

1. The BPM slightly and progressively changes during the song.
2. The BPM suddenly changes during the song.
3. The BPM cannot be calculated in some part of the song.

A finite state machine controls the calculation of the BPM during the song and acts depending on the circumstances discussed above. In the case 1, the algorithm verifies that the change in BPM is progressively increasing or decreasing and is different from noise or the precision related. In case 2, the algorithm checks that the change has been maintained for at least N frames to confirm it. In case 3, the algorithm continues with the same previous and stable BPM until a new valid BPM (the same or one included in case 1 or 2) is detected.

The second part of beat tracking, the onset estimation, has also been discussed in depth in the literature. Complex algorithms have been shown to be valid for different types of music with a high success rate. However, 100% of accuracy is still not reached because of the complexity of the problem. We have tested generic state-of-the-art algorithms [7] in EDM obtaining poor results, discarding them for this application. If the problem is restricted to a specific kind of music, performance can be improved. That is the reason why we have developed our own specific onset algorithm.

In EDM music, the patterns and musical phrases are quite constant, they do not change during the song at all. This means that

if an onset is determined with a high degree of confidence, it can usually be maintained throughout the song without changes. Even if the main drums that follow the beat disappears in some moments of the song, just extrapolating the position of the next beat from the tempo can be enough to follow the onset. Later in the song, when a clear beat is present, usually related to percussion, the algorithm can re-link the onset to this beat.

The resulting onset algorithm is implemented as a finite-state machine with an initial set-up state where a high confidence onset is obtained. In each window the algorithm checks the beat using a beat monitoring function similar to [9]. If the function provides the higher value for the current onset, the algorithm continues with the same onset synchronization. If the value is not the highest but it is high enough, nothing is changed; that situation can cope with temporary changes in the percussion composition. Only in the case that a different onset persists during a long period of time, a decision on onset changing is performed by the state machine. The different level thresholds, time thresholds, and other fine-tuning parameters have been adjusted by testing with a database of more than 100 EDM songs. Details of this procedure are out of the scope of this paper.

3.- FREQUENCY BALANCE ANALYSIS

Music experiences progressive changes in its frequency composition along its duration. The frequency balance at each moment is generally well perceived by the listener. In this way, a moment in the song can be described as bassy, trebly, voiced, etc. These changes are generally slow compared with the beat, so they describe the song softly. Attending to this criterion, we have defined some frequency balance parameters that are computations of the energy in some frequency bands. They provide a high correlation with the perceptual sensation experienced by the listener and therefore suitable for video synchronization. Moreover, low computational cost and simplicity of interpretation have been taking into account.

The frequency balance analysis is mainly based on a 3 bands filtering decomposition of the audio signal. Figure 2 describes the workflow of the feature extraction process. There is a common block of the music analysis, containing the bands decomposition, energy computation and softening. The resulting parameters of this stage would not only represent directly usable information for video synchronization (Bass/Mid/Treble), but they also constitute the feed for other stages: color representation (RGB, HSV) and segmentation (events), that will be explained in next sections.

The Common Audio Analysis Block comprises the next stages:

Windowing: As this algorithm is related with video signal, the processing window rate employed is linked with the frame rate of the video signal, typically 50 or 60 Hz, corresponding to windows of 20 or 16.6 ms. Smaller windows do not provide any advantage in latency and larger windows would increase it. Hamming windowing is employed.

Filters: The signal is divided into three bands using 4th order IIR filters. Filter 1 comprises frequencies up to 250 Hz, filter 2 comprised frequencies between 250 Hz and 4000 Hz and finally filter 3 is for frequencies over 4000 Hz.

Energy computation: The energy for the three frequency bands (low, mid & high) for each window is computed and converted to a logarithmic scale.

Features filtering: To smooth out the energy features with minimum latency penalty, an algorithm from the image processing field called bilateral filtering [10] has been adapted to audio. When a feature changes rapidly during the time, in order to adapt it for visualization (animation), smoothing is the common solution. A standard Kalman filter can be employed, but is not the best solution, because the output does not react to sudden changes of the music volume. Using bilateral filtering the outputs of the previous stages can be smoothed while preserving edges.

The bilateral filtering represented by f(x) in Eq. 1 can be summarized as a combination of a range s(f(ξ),f(x)) and a domain filtering c(ξ,x) as

$$\mathbf{h}(\mathbf{x}) = k^{-1}(\mathbf{x}) \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} \mathbf{f}(\xi) c(\xi, \mathbf{x}) s(\mathbf{f}(\xi), \mathbf{f}(\mathbf{x}))\, d\xi \quad (1)$$

where k(x) is the normalization factor.

$$k(\mathbf{x}) = \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} c(\xi, \mathbf{x}) s(\mathbf{f}(\xi), \mathbf{f}(\mathbf{x}))\, d\xi \quad (2)$$

The bilateral filtering has been applied to the 3 frequency bands and employed in a causal way successfully.

Different weights are applied to each frequency band with a gain transformation matrix M1, because high frequencies usually need to compensate their lower energy to achieve similar variations than the lower bands. As a result, the energy features Bass, Mid and Treble are obtained.

An example of application of these features to video effects or animations is the association of the size of 3D objects with them, for example the width to Bass, height with Mid and depth with Treble.

Figure 3 shows the evolution of the three parameters during a full song. A pseudo-color track has been also added for visualization of the parts at a glance.

4.- COLOR REPRESENTATION

It is powerful to show visual color effects synchronized with the songs. There have been many studies on the synaesthesia between audio and colors, some very interesting ones study the association of colors to musical notes [11]. In this article they postulate as convenient to use the chromatic scale from red to green to blue in an orderly manner with the musical scale from DO to SI, that is, increasing with frequency. In our case, we have also adopted a de facto standard in commercial DJ software where the low frequency is associated with red, mid frequencies with green and high frequencies with blue.

To this end, based on the calculation of the energy in bands with their corresponding stabilization, a color transformation matrix M2 is applied to obtain 3 values, R, G and B. The tone is clearly associated to R with low frequency, G with medium frequency and B with high frequency. In addition, brightness and saturation are set to maximum to allow the colors to be as eye-catching as possible. Another color transformation matrix M3 is also applied to obtain HSV values for these colors and to facilitate their use.

A common application of the obtained values for colors is to apply them to the visual background or 3D object materials so that viewers naturally relate them to the different parts of the songs they are listening to.

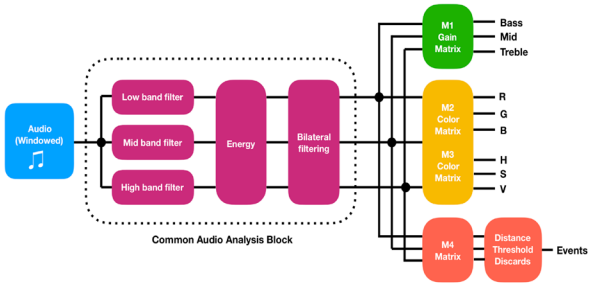

Figure 2: Block diagram of features extraction.

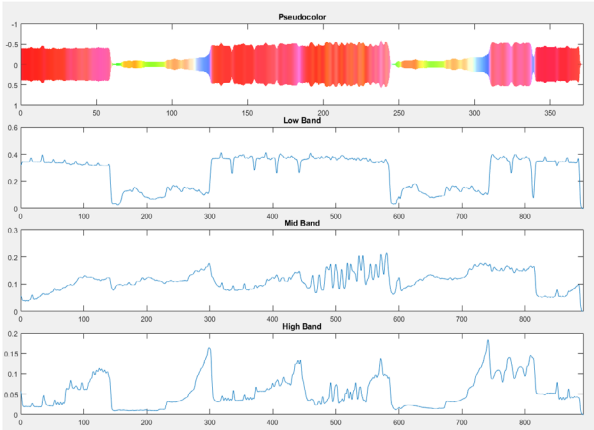

Figure 3: Evolution of the three values at the output of the common block during a full EDM song. A pseudo-color track has been created for visualization of the different parts at a glance

5.- MUSIC SEGMENTATION

5.1.- Electronic Dance Music Structure

When it comes to characterizing music, the genre particular features and the intention of the analysis will influence the choices of the recognition and segmentation system. In general, contemporary and pop music usually have a structure based on parts typically

named as: Intro, Verse, Chorus, Solo, Bridge, Outro, that are interleaved during the song.

- **Intro:** It is optional and is reproduced only once at the beginning of the song. In pop music should last 16 or 32 bars.
- **Verse:** This is the first main part of an arrangement. It is repeated a few times before moving on to the chorus. It usually contains lyrics.
- **Chorus:** The main part of the song. It's the hook, what the audience remembers and the most powerful passage. It will be usually repeated interleaved with the Verse.
- **Solo:** It can appear at any time to add a jam effect. It can create cool sections in music. They are very common in live performances.
- **Bridge:** It is used to break up what the listener has paid attention to. In electronic music the drums usually produce an ascending sound that is added to the next part. A bridge can be more powerful by adding new instruments or changing the key.
- **Outro:** It is used to resolve the song and go towards a soft landing.

However, the EDM structure works differently:

1. Quite frequently there are no lyrics or at least they are not as important as in pop music, so the concept of verse does not apply here.
2. Instead of a chorus, it is more common a single powerful loop which acts as the hook of the song.
3. Then the EDM song is built up adding and subtracting different instrument layers over the beat structure and driving the emotions of the listener by means of different tricks.
4. In EDM, the different parts are not necessarily repeated again. On the contrary, the parts are usually an evolution or modification of previous parts.

The most common structure in EDM is composed by:

- **Intro:** This is usually either a kick of some percussion and melodic elements which introduce the feeling of the track or a fade in of a melodic layer.
- **Breakdown:** It is the mellowest part of a track, where the melody is introduced and the atmosphere use to be calm.
- **Build-up:** All the melody layers in the breakdown are taken and some of the layers that are going to be used in the drop are added, establishing the most energetic part of the song.
- **Drop:** It changes according to the subgenre, but in general it is a passage in a dance track in which the tension is released and the beat kicks in. In some dance styles, it is known as a climax.

Moreover, the identification of this structure is more complex and less predictable when working with live DJ music. Songs do not start and end clearly because the DJ removes intros and outros to mix continuous music, making detection and segmentation more difficult.

On the other hand, within EDM music, different subgenres can be classified: house, techno, trance, dubstep, drum & bass, minimal and others. Each of these sub-genres has its own rules that can be adapted differently to the structure of the EDM music discussed above. This significantly complicates the development of universal software that can solve the segmentation challenge, even more if we consider that human listeners have no consensus in this field either; two different people would probably label segments differently.

Therefore, from a practical point of view, when asking what a segment is and what utterance it determines, we can focus our attention on the final application in order to narrow the problem. With the aim of giving a visual output that complements the visual effects, each segment can be considered as a part of the song with a minimum and maximum duration that can be noticed by the EDM audience. It includes detection of breaks and the main layers added to or subtracted from the music, among others events.

5.2.- Segmentation algorithm

In this work, a simple low-latency algorithm has been employed for the extraction of the important events in the structure of EDM music. It employs low computational cost and is efficient for extraction EDM features. Although FFT is used in other works in order to segment and classify the parts of a song with great detail [4], in this work an algorithm that does not use FFT have been developed instead, saving computational cost and reducing latency but obtaining enough information for the final purpose.

For practical reasons in this work, we have focused our efforts on detecting events that capture the attention of the listener during playback. In this way, these moments can be associated with a major change in the video animation. Four easily audible key events have been considered in the course of a song:

1. The beginning and the ending of the breaks that alternate between high and low energy moments of the music.
2. The different layers that are added or removed from the main base that build-up the song.
3. The moments when a new passage begins in crescendo or decrescendo.
4. Attractive punctual sounds, something very common in EDM.

The source for the segmentation algorithm is the features vector consisting of the three instantaneous values of the smoothed energy per band as explained in section 3. As commented in that section, the rate of the features matches the video frame rate (typically 50 or 60 Hz) and low latency should be desirable in the segmentation algorithm as well as in the features extraction. The proposed segmentation process consists of the following stages:

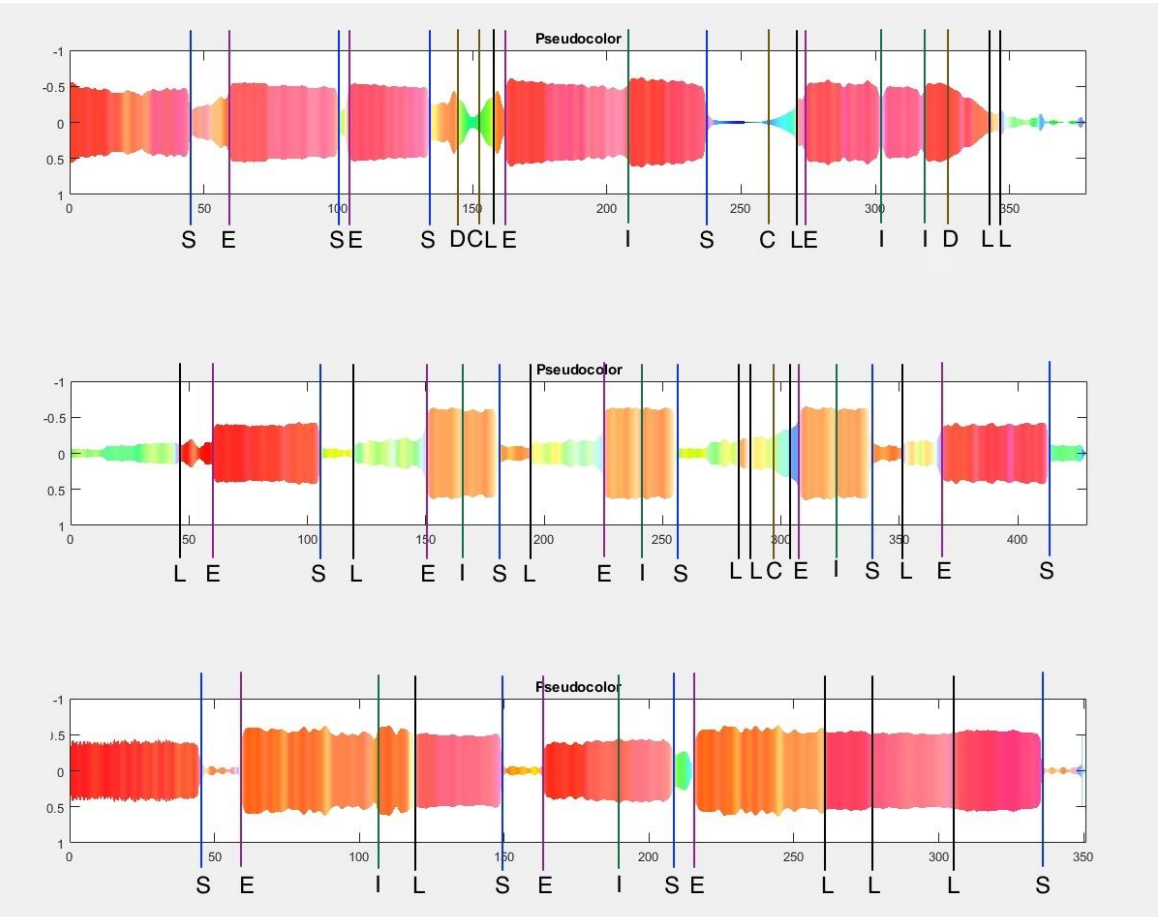

Figure 4: Resulting of labelling for three songs (S = Break Start, E = Break End, L = Layer Change, I = Instant Event, C = Crescendo, D = Decrescendo)

1) A transformation matrix M4 is applied to the three energy values to properly weight them and provide better detection of changing sections in the song.

2) The Euclidean distance between the actual and the previous feature vector is computed. The bilateral filtering applied in the previous stages guarantees some immunity to small changes but preserving big ones, which improves the detection.

3) Distance is compared to a threshold set in the training process which is described later.

4) If one of the previous thresholds is triggered, a mechanism to eliminate false positives is also implemented by attending to a minimum distance between events.

5.3.- Training of the algorithm

To adjust the different weights, threshold and minimum distance between events of our algorithm, it is necessary to fine-tune its values. This adjustment should be made using samples of the music style of interest.

A training set consisting of 100 songs of different EDM styles has been compiled. A balance between the different subgenres commented in section 2 has been taken into account. Next, a musical expert has hand-labeled the events in all the songs creating a database of time events.

The same expert labeled all songs because, due to the special characteristics of EDM music, sometimes two experts may have different criteria. Figure 4 shows an example of labeling for three different songs. The following events have been labelled: S = Break Start, E = Break End, L = Layer Change, I = Instant Event, C = Crescendo, D = Decrescendo.

Four parameters have been selected for the training process: two weight factors for mid and high frequency bands with respect to the low band, the features-vector distance threshold and the minimum time between events.

A simple optimization algorithm consisting of these steps is then started:

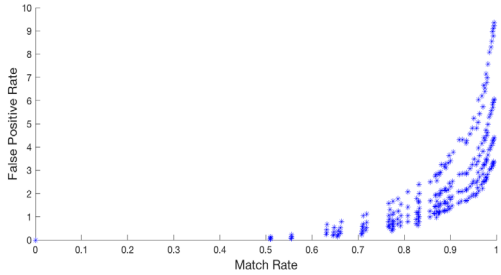

Figure 5: Match Rate vs False Positive Rate cloud of results for the first training iteration

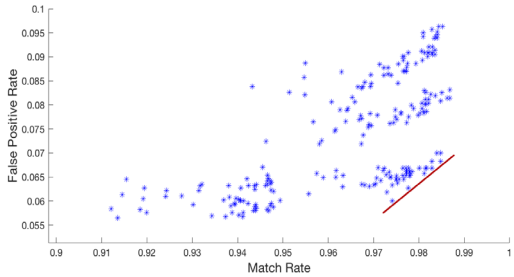

Figure 6: Match Rate vs False Positive Rate cloud of results of the last training iteration. Optimal solutions are along the red line.

1) The entire database is automatically labeled with a 4D grid of four equispaced values for each parameter, totaling 256 combinations

2) The match rate (matches / labeled events) and false alarm rate (false positives / labeled events) are calculated and plotted for each combination. This analysis gives us the information needed for step 3. Figure 5 shows the first iteration of this training with a very poor false alarm rate of almost 1000%.

3) Around the best combination, the grid is refined and step 2 is repeated. In the case that the best result of any variable is in the border of the grid, instead of refining it, the grid is extended in this direction and step 2 is repeated.

4) The iteration stops when there is no improvement in various iterations or a number of iterations are reached. In Figure 6 we can see the last iteration with the best results, all below 10% false alarm rate.

Since a perfect result with 100% match and 0% false positives cannot always be obtained on a large database, a trade-off was explored between the two values. Thus, it was necessary to choose which values of the four variables

produce the best solution. Figure 6 shows a line of optimal solutions marked in red. The solution selected for our purpose, which belongs to that line, has a 98.48% match rate and 6.82% false positives rate. This solution corresponds to the 4 specific values combination that we were looking for.

These results are quite good considering that the false alarm rate is not of prevalence importance in our case. In practice, finding more subtle changes in the song (than those a human would label) would not cause visual disturbances in the video animation.

5.4.- Results

The segmentation algorithm has been validated with a different set of songs, obtaining a matching detection rate of about 96%, which is similar to the results of the training set. The false alarm rate is constrained in an acceptable rate of 10-20%, depending on the set of songs saved for the validation test. Anyway, considering what was commented in the previous section, this quantity of false alarms does not represent a major problem.

A typical application for video effects of the detected segmentation events would be to change into a different animation scene or to move camera viewpoints synchronized with song section changes.

6.- CONCLUSIONS

In this work, different adapted and novel algorithms to extract easily audible musical features from Electronic Dance Music (EDM) have been presented. The focus has been put on its application for the automation of video and 3D animation synchronized with the audio content.

The algorithms implement low-cost time-frequency features, some of them smoothed with the novel application of a known image filtering algorithm to audio, resulting in a low latency output. The segmentation stage is based on a simple energy band distance threshold algorithm specifically tuned for EDM. Through a database of songs, a training stage has been carried out for the fine tune of weights, threshold, and minimum events separation. The balance between match rate and false alarm rate has been heuristically fixed according to the practical application.

The final algorithms have been implemented in C performing in real-time and have been successfully integrated into a real-time commercial 3D graphics engine called "Visual Music" that is under development, as it can be seen in figure 7. This piece of software is now being tested by video artists who are applying it to real scenarios. The feedback from these early users will be employed to improve the algorithm and to add new features.

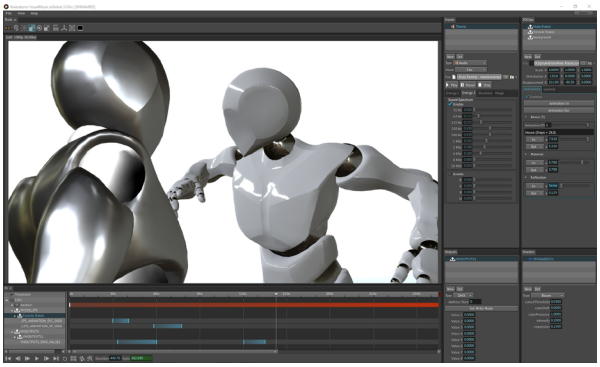

Figure 7: Visual Music software with audio module included.

7.- ACKNOWLEDGMENTS

This work has been supported by the European Commission under the H2020 program, project VISUALMUSIC, ID# 731986

8.- REFERENCES

[1] H. Meinedo and J. Neto, "Audio segmentation, classification and clustering in a broadcast news task.", in IEEE International Conference on Acoustics, Speech, and Signal Processing, 2003. (ICASSP '03). vol. 2, pp. 2–5. Hong Kong, China, 2003.

[2] G. Xia, D. Liang, R. B. Dannenberg and M. J. Harvilla, "Segmentation, clustering and display in a personal audio database for musicians.", in 12th International Society for Music Information Retrieval Conference (ISMIR 2011) pp. 139–144. Miami, Florida, USA, 2011.

[3] A. Bietti, "Online learning for audio clustering and segmentation", École Normale Supérieure de Cachan, Inria IRCAM. 2014.

[4] A. Dessein and A. Cont, "An information-geometric approach to real-time audio segmentation.", in IEEE Signal Processing Letters, 20(4), pp. 331–334, 2013.

[5] M. A. Siegler, U. Jain, B. Raj and R. M. Stern, "Automatic Segmentation, Classification and Clustering of Broadcast News Audio.", in Proc. DARPA Speech Recognition Workshop pp. 97–99, 1997.

[6] M. Levy and M. Sandler, "Structural Segmentation of Musical Audio by Constrained Clustering.", in IEEE Transactions on Audio, Speech and Language Processing, 16(2), pp. 318–326, 2008.

[7] D. Ellis, "Beat Tracking by Dynamic Programming", in Journal of New Music Research, Special Issue on Beat and Tempo Extraction, vol. 36 no. 1, pp. 51-60, 2007.

[8] Y. Shiu and J. Kuo, "Musical Beat Tracking via Kalman Filtering and Noisy Measurements Selection", in ISCAS 2008. IEEE Int. Symposium on Circuits and System, pp. 3250-3253, May 2008.

[9] Y. Shiu and J. Kuo, "On-Line Musical Beat Tracking with Phase-Locked-Loop (PLL) Technique", in IEEE Int. Conference on Consumer Electronics, April 2007.

[10] C. Tomasi and R. Manduchi, "Bilateral filtering for gray and color images", in Sixth International Conference on Computer Vision (IEEE Cat. No.98CH36271) pp. 839–846. Bombay, India, 1998.

[11] K. Itoh, H. Sakata, I. L. Kwee, and T. Nakada, "Musical pitch classes have rainbow hues in pitch class-color synesthesia," in Sci. Rep., vol. 7, no. 1, p. 17781, Dec. 2017.

9.- BIOGRAPHIES

Emanuel Aguilera received a telecommunications engineering degree in 2004 and a M.S. degree in Artificial Intelligence, Pattern Recognition and Digital Image in 2011, both from the Universitat Politècnica de València, Spain. He is a researcher and senior programmer at the Institute of Telecommunications and Multimedia Applications (iTEAM), where he has been working since 2006 on the area of digital signal processing for audio, multimedia, virtual reality and mobile devices applications. He is interested in wave-field synthesis, image processing, real-time multimedia processing for telecommunications and audio applications for mobile platforms.

Jose Javier Lopez was born in Valencia, Spain, in 1969. He received the telecommunications engineer degree and the Ph.D. degree, both from the Universitat Politècnica de València, Valencia, Spain, in 1992 and 1999, respectively. Since 1993, he has been involved in education and research at the Communications Department, Universitat Politècnica de València, where he is currently a Full Professor. His research activity is centered on digital audio processing in the areas of spatial audio, wave field synthesis, physical modeling of acoustic spaces, efficient filtering structures for loudspeaker correction, sound source separation, and development of multimedia software in real time. He has published more than 160 papers in international technical journals and at renowned conferences in the fields of audio and acoustics and has led more than 25 research projects. Dr. Lopez was workshop co-chair at the 118th Convention of the Audio Engineering Society in Barcelona and has been serving on the committee of the AES Spanish Section for nine years, currently as secretary of the Section. He is a full ASA member, AES member and IEEE senior member.

Pablo Gutierrez-Parera was born in Córdoba, Spain in 1982. He received a telecommunications engineer degree in 2008 from the Universidad Politécnica de Madrid. In 2010 and 2013 he obtained a M.S. degree in digital postproduction and a European graduate in telecommunication systems, sound and image engineering, both from the Universitat Politècnica de Valencia. Currently, he is a PhD grant holder from the Spanish Ministry of Economy and Competitiveness under the FPI program and is pursuing his PhD degree in telecommunications at the Institute of Telecommunications and Multimedia Applications (iTEAM) working in the field of spatial audio.

Carlos Hernandez is a Telecommunications Engineer from the Higher Polytechnic Institute "José Antonio Echeverría" (ISPJAE), Havana, Cuba, in 1989 and Ph.D. in Telecommunications Engineering from the Polytechnic University of Valencia (UPV), Spain, in 1998. He is teacher at the Communications Department of The UPV since 2001 and its current lines of research are related to the processing of signals and their application in the field of sound. He develops educational projects and teaches courses and workshops for teachers in order to improve educational methods and the quality of teaching in the university, as well as the study of the social impact of technologies as dynamic elements of sustainable human development and cooperation. He is a teacher at the UPV SENIOR University where he has been teaching courses since 2009 on the use of Internet and its services by elder people. He also develops musical applications based on Smartphones and Tablets for being used by people with intellectual and sensory disabilities.