**Tatsuo Unemi**

# *Paper:* Synthesis of sound effects for generative animation

***Topic: Sound synthesis***

***Authors:***
***Tatsuo Unemi***
Soka University,
Department of
Information Systems
Science
Japan
www.soka.ac.jp

***References:***
[1] Tatsuo Unemi, "*SBArt4 as Automatic Art and Live Performance Tool*", GA 2011 – XIV Generative Art Conference, Rome, 2011.
[2] www.intlab.soka.ac.jp/~unemi/sbart/4/

***Abstract:***
As everyone knows, the sound effect and background music of motion picture is effective to emphasize what the author wanted to express. However, in case of fully automated generative animation, it is difficult to introduce such a method for design of accompanying sounds because the generator has no intention behind the process. This paper introduces a method to synthesize waveforms of sounds for an automated evolutionary animation [1, 2] by computer. To emphasize the emotional effects for viewers, it was designed as to fit the psychological effect of sounds with visuals under some intuitive correspondences between these two different modalities, such as a brighter image is associated with a higher pitch, a more complex texture inspires a noisier or more solid tone, and so on. The other mappings between statistical features of image and parameters of sound synthesis and modulation are also effective to produce richer audio outputs. In addition, the two types of restriction on pitches for the scale and on timing for the rhythm were also examined for automatic music composition. There are many potential applications and extensions from this research, including evolutionary production of sound effects, as future works.

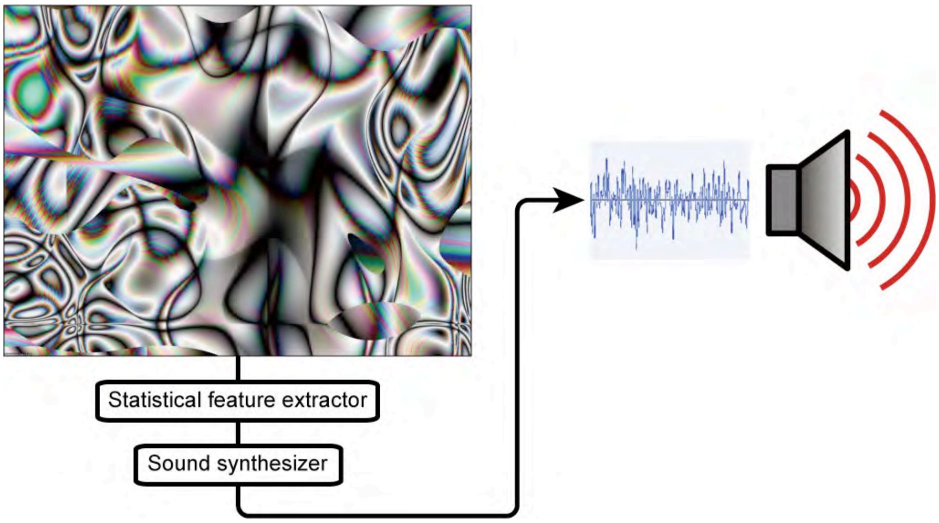


*Illustration of sound synthesis from visuals.*

***Contact:***
***unemi@t.soka.ac.jp***

***Keywords:***
Sound synthesis, abstract animation, automatic art

# Synthesis of Sound Effects for Generative Animation

Prof. T. Unemi, BEng, MEng, DEng.
Department of Information Systems Schience, Soka University, Hachioji, Japan
www.intlab.soka.ac.jp/~unemi/
e-mail: unemi@iss.soka.ac.jp

## Premise

This paper introduces a method to synthesise sound effects from video images for an automated evolutionary animation. To emphasise the impression of visuals, it was designed as to fit the psychological effects of two different modalities under some intuitive relation between visual and audio stimuli. The other mappings from the statistical features of image to the parameters of sound are also effective to produce richer audio outputs even if there is no clear correlation between them. In addition, the two types of restriction on pitches for the scale and on timing for the rhythm were examined for automatic music composition. There are many potential applications and extensions from this research, including evolutionary production of sound effects, as future works.

## 1. Introduction

The sound effect and background music of motion picture is effective to emphasise what the author wanted to express. However, in case of fully automated generative animation, it is difficult to introduce such a method for design of accompanying sounds because there is no established method for automatic sound design. Furthermore the machine has no intention behind the process. This paper introduces our approach to synthesise sound waveforms of by the computer for an automated evolutionary animation [1, 2]. To emphasise the emotional effects for viewers, it was designed as to fit the psychological effect of sounds with visuals under some intuitive correspondences between these two different modalities, such as a brighter image is associated with a higher pitch, a more complex texture inspires a noisier or more solid tone, and so on. The other mappings between statistical features of image and parameters of sound synthesis are also effective to produce richer audio outputs.

In addition, the two types of restriction on pitches for the scale and on timing for the rhythm were examined for automatic music composition, which is expected to be enjoyable for audience from wider variation of backgrounds. The design of envelope, scale, harmony, and rhythm are important for music composition as same as timbre.

The sound design for motion picture has been a target of research for many years. The early works mentioned sound effects for film such as [3], but some of the recent works are relating to automatic sound synthesis from the simulation of physical entities, such as sounds of flowing water [4] and flames [5]. These approaches are to produce both sounds and visuals from a single computational model of physical objects, but not sound synthesis from visual data. Another approach to produce sounds from real-time motion visuals was examined by Dannenberg et al [6]. They proposed and implemented a system that synthesises sounds from video images of the waves on water surface in a small shallow container. Such an approach using fluctuation of natural phenomena is useful to produce a sound of natural flavour by an artificial system on the digital computer, avoiding a mechanical flavour. The technical requirement is almost same with ours, but the concept is facing almost the opposite orientation, because our installation is to build up another nature in the machine without any connection to the real nature.

The following sections describe intuitive relation between sounds and visuals, methods for synthesising waveforms, modulation, automatic music composition, and adjustable parameters by the user.

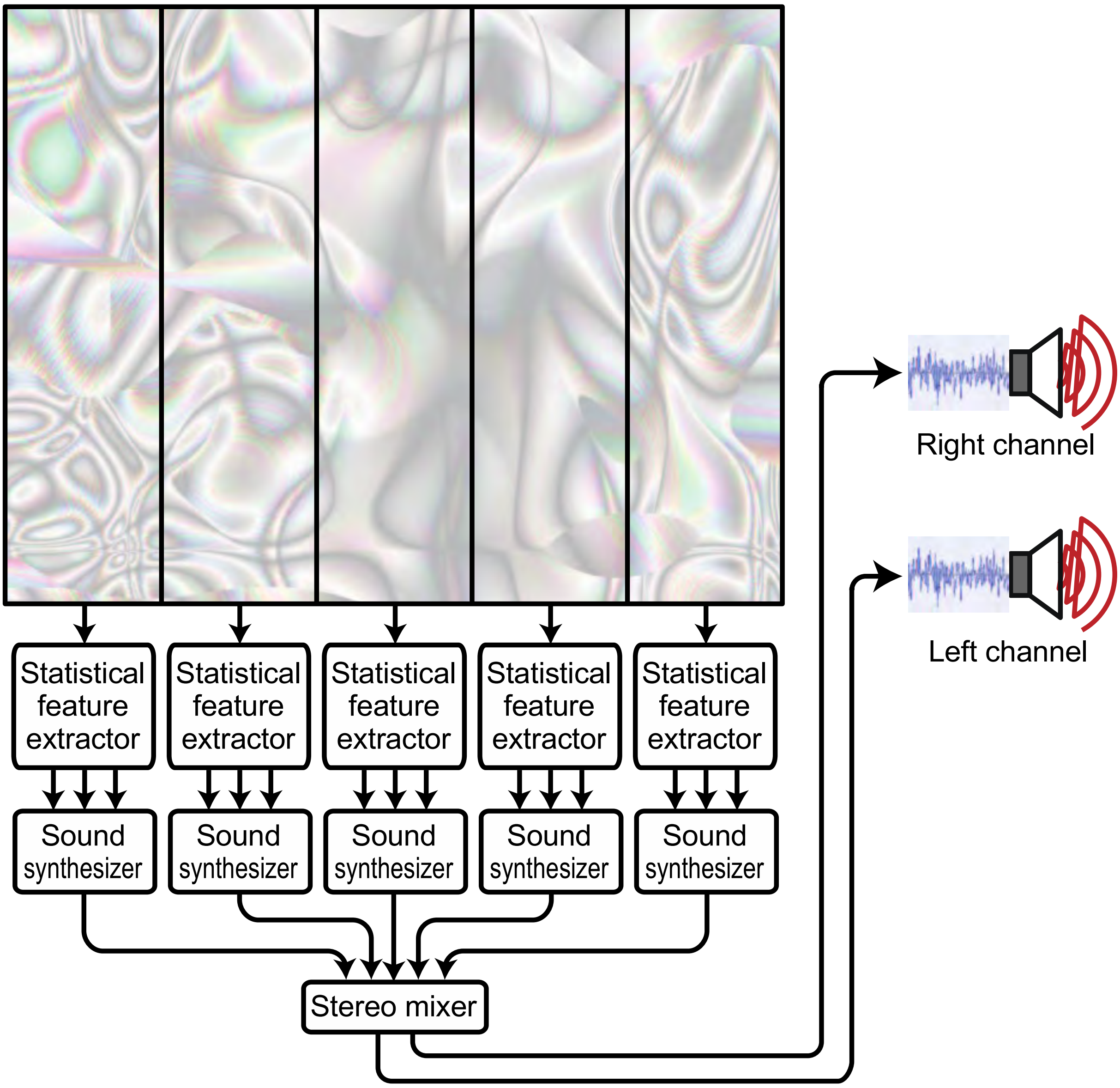

Statistical feature extractor
Statistical feature extractor
Statistical feature extractor
Statistical feature extractor
Statistical feature extractor
Sound synthesizer
Sound synthesizer
Sound synthesizer
Sound synthesizer
Sound synthesizer
Stereo mixer
Right channel
Left channel

Figure 1. *Data flow of sound synthesis. A frame image is split into five sub-regions.*

## 2. Intuitive relation between sounds and visuals

Hearing senses by ears and visual senses by eyes are of course in different types of modality in human's sensation. However, it is also well known that people has an intuitive association between the stimuli of different modalities, such as brightness of both colour and sound. We can assume that the origin of these associations have been acquired from frequent experiences of physical phenomena commonly having happened among people. It causes both visible and audible signals for human at the same time, such as a sand storm that makes a scratchy image and a noisy sound. In the context of spatial perception, the positioning of a visible object in the personal view also has strong association with the positioning of sound it produces. For example, it is natural to position a sharp sound at the left audio channel when a bright light appears at the left side.

In this system, we implemented four features from this point of view; positioning of a multiple sound source, colour brightness and sound pitch, motion speed and loudness, and rough image and white noise.

### 2.1 Multiple sound sources

Multiple sound sources are a typical technique to make a richer sound. In the music, a solo play of a single instrument is interesting but an orchestra sound by tens of different types of instruments and players has large possible variations and can produce organised complex sounds of a high dynamic range in the audible sound frequencies. One problem to organize such a large scale of orchestration by the computer is the complexity from the large number of elements. It causes not only a problem of combinatorial organization but also of computational power to calculate them in real-time.

Because the number of sound sources should be restricted within a reasonable range to guarantee the sound synthesis is processed in real-time by a personal computer, we use at most eight sound sources that separately produce a sound for each. Each sound source corresponds to the sub-region of the frame image vertically split as shown in figure 1. The statistical features extracted from the image data in each sub-region are fed to the sound synthesizer. And then, those sound signals are mixed so that each one is positioned from left to right as same as the sub-regions are arranged, by adjusting the balance between stereo loudspeakers.

### 2.2 Brightness and pitch

The most effective feature of the melodic sound is the pitch. The system maps the average value of brightness over all pixels in the sub-region into the frequency of sound wave within the range from 110 Hz to 880 Hz. Therefore a brighter image makes a higher pitch, and a darker image makes lower pitch. This association is intuitively natural, but it sometimes results a cheap sound when the values of average brightness for all sub-regions are almost same. To guarantee the produced sound includes a wide dynamic range, we introduced a mechanism to add a variation of pitch ranges by shifting them in a proper number of octaves from the waveforms synthesized by the normal method. The lowest frequency is 55 Hz, and the highest frequency is 3.52 kHz. The highest pitch might seem still too low because human ears can hear the sound of higher than 15 kHz, but it is not a critical problem because a component of higher frequency is usually included in the basic waveform as described in later section 3.

### 2.3 Motion speed and loudness

When the image is changing fast, it looks hectic and presents strong stimuli since it provides dense information. A louder sound is suitable for such a case. If the motion is slow and calm, a monotonic sound of pianissimo seems to be appropriate. We implemented a calculation of average difference between brightness of two pixels in the same spatial position in the current frame and the image of weighted average over recent frames in order to compute the motion speed in visual frames. The brightness $v_t$ of the pixel at the time $t$ in the image of weighted average is revised in each step using an expression

$$v_t = \alpha \cdot b_{t-1} + (1 - \alpha) \cdot v_{t-1} \tag{1}$$

where α is the weight constant of $0 < \alpha < 1$, $b_t$ is the brightness of the pixel at the same position in the frame image in time $t$, and $v_0 = b_0$. The resulted value expressing the motion speed is also used to producing a synchronized rhythm for musical sound as described later in section 5.

### 2.4 Roughness and noise

A rough image, such as a ground surface with sands, associates noisy sound. To measure a degree of roughness, we calculate the average value of absolute differences of brightness among all of neighbouring pixels in a sub-region of the frame image. This value is mapped to the amplitude of white noise by adding a random numbers to each sound sample. To avoid this noise always happens, we introduce a threshold value of roughness. The random number is added only when the roughness is larger than the threshold value, and its value is multiplied by a coefficient that gradually increases from zero to one following the strength of roughness.

## 3. Synthesis of waveform

We introduced two types of methods to construct a basic waveform. The first one is to make a sequence of average brightness values of rows scanning from the top to the bottom of each sub-region in the frame image. The values are linearly normalized within a range [–1, 1]. The absolute value of samples in the top part is modified to gradually increase and the bottom part is modified to gradually decrease, so that the waveform can connect smoothly to the next phase of the waveform's repetition, as show in figure 2. The sharpness of the sound is adjusted using a type of low pass filter by calculating a moving average over the sequence of sample values.

Another method is to compound the harmonic overtones of which amplitudes are determined from the average brightness in sub-regions vertically divided as shown in figure 3. The number of harmonic overtones is 12 in the current implementation. In this method, the sharpness is adjustable by a decay coefficient γ ($0 < \gamma < 1$). The sample value $s_t$ at time $t$ is defined as

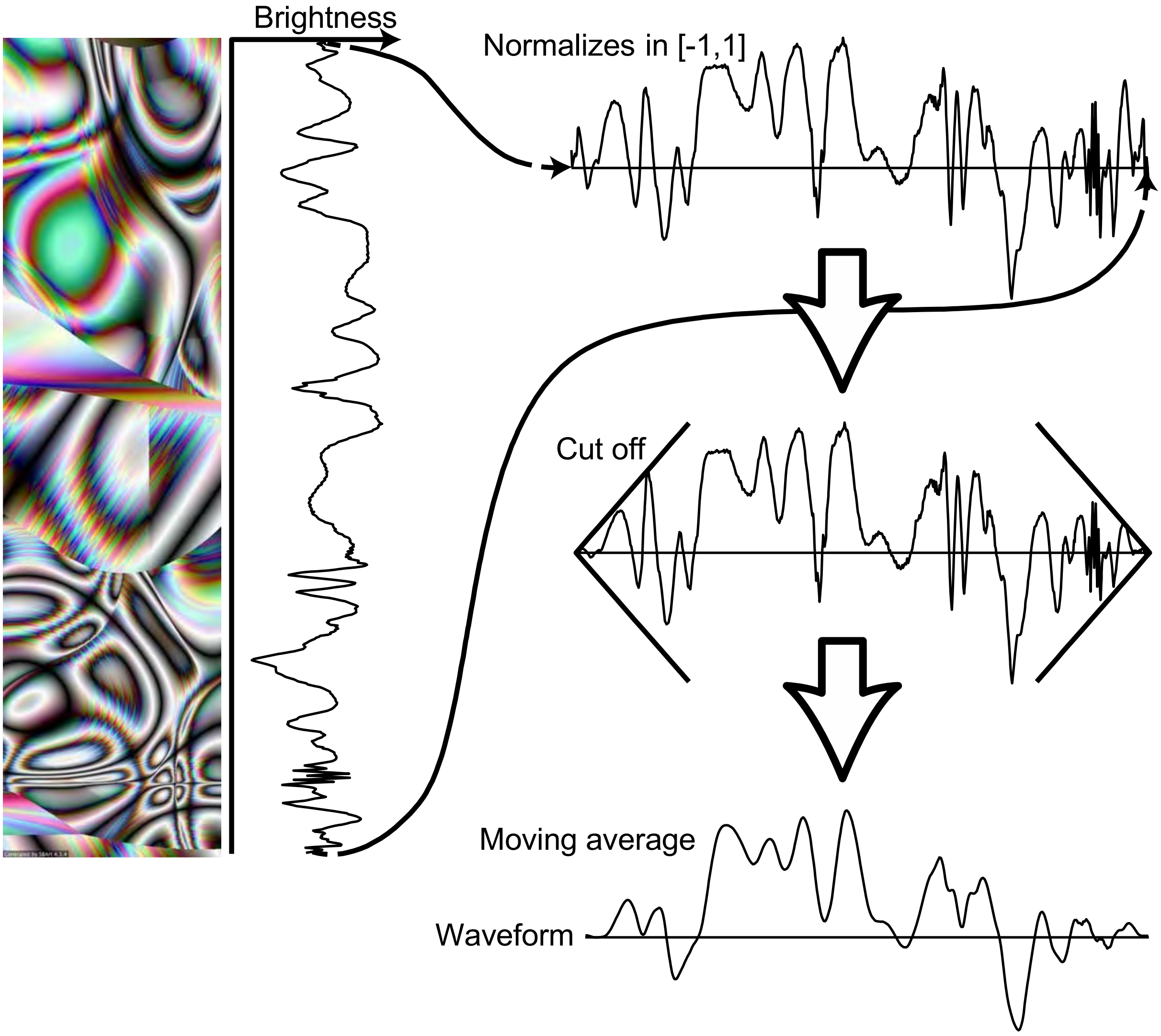


Figure 2. *The first method for synthesis of the waveform.*

$$s_t = \frac{1}{12}\sum_{i=1}^{12} B_i \gamma^{i-1} \sin 2\pi \cdot \phi \cdot t \cdot i \qquad (2)$$

where $B_i$ is the average brightness in the $i$th row in the sub-region, and φ is a coefficient for the base frequency of the sound that is determined from the average brightness in the sub-region, the range of sound frequencies, and the sample rate.

These two methods produce different timbres, but the common characteristic is that a smooth image pattern makes a clear sound by sine curves and a complicated pattern makes a sharp sound by complicated waveforms. In the latter case, the sound includes a component of high frequency.

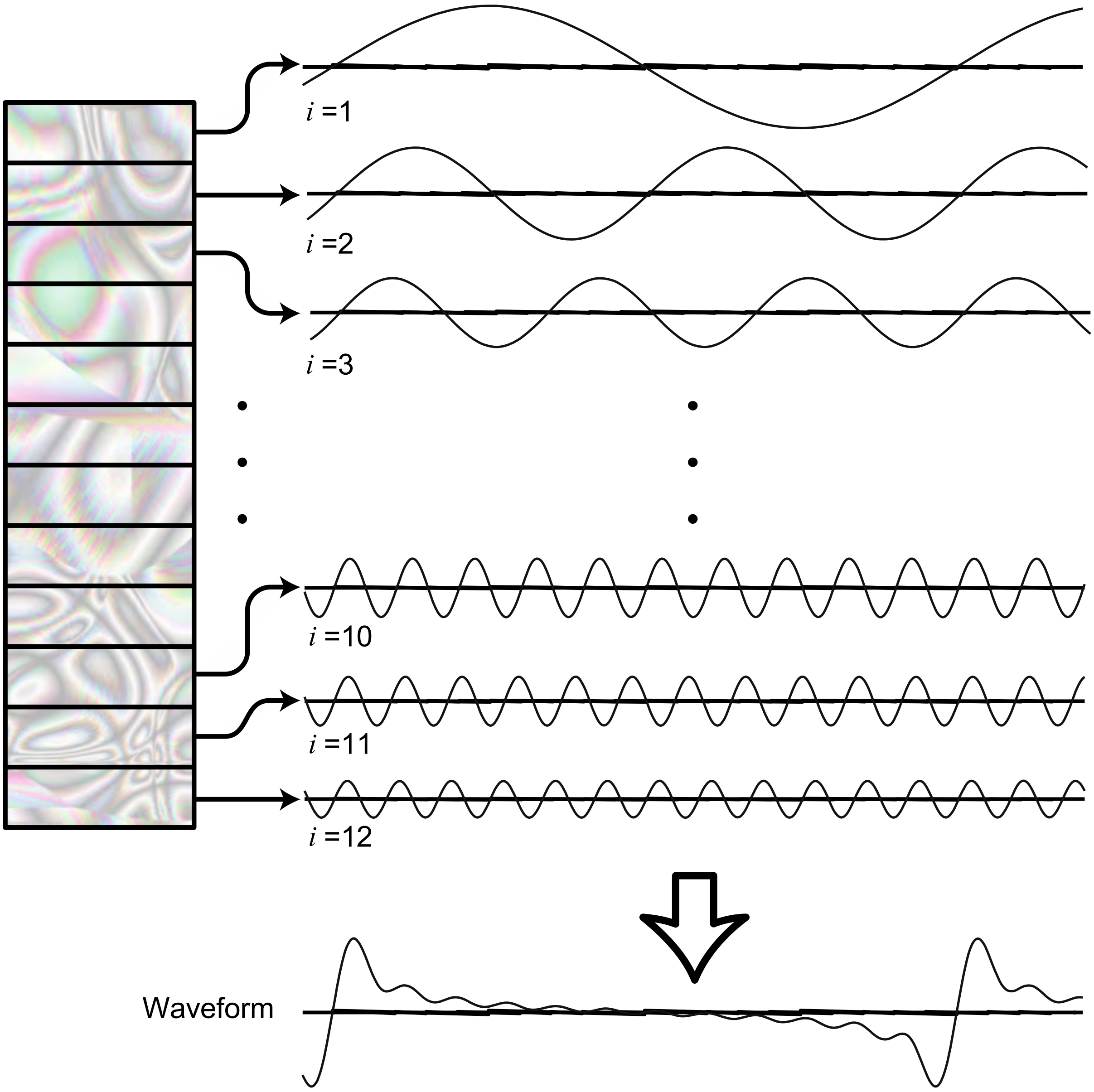


Figure 3. *The second method for synthesis of the waveform. It takes weighted summation of 12 harmonic overtones.*

## 4. Sound modulations

It is useful to introduce a variety of parameters for sound synthesis in order to emphasise the changes of visuals even if there is no obvious association between the features in image and sound. The statistical factors we use in the current implementation are (1) the standard deviation of brightness, (2) the average and (3) the standard deviation of saturation, (4) the average and (5) the standard deviation of hue values over all of pixels in the sub-region, (6) the average of variances for brightness for each row, (7) the horizontal and (8) vertical position of the center of gravity on brightness. The target parameters of sound synthesis corresponding to each of these statistical factors are (1) bandwidth of low-pass filter for basic waveform, (2) frequency and (3) strength of amplitude modulation, (4) frequency and

(5) strength of frequency modulation, (6) phase shift between left and right channels, (7) time delay of attack for musical note (8) tail length of envelop. The last two parameters are applicable only for musical sound described in the later section 5. These correspondences are not effective when the statistical factors in the image are stable, but they are very effective when the factors are dynamically changing. The ranges of sound parameters are adjustable by human as described in the later section 6.

## 5. Automatic music composition

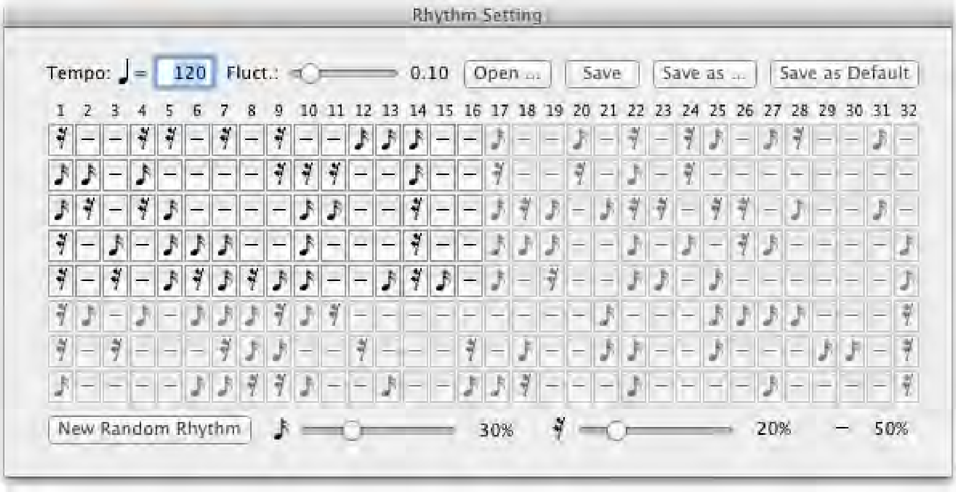
Human being has a long history of sound design for the life. The voice is an important for communication; the bell, chime, whistle, siren, etc. are useful for warning and attention; and the music is enjoyable in a festival and a ceremony.

Using the methods of sound synthesis described above, it is possible to produce a sound continuously changing because the parameter values extracted from frame images are also changing as the frame alternation goes on. The usual frame rate of smooth animation is not fast enough for continuous change of sound, that is, it sounds changing stepwise when the same parameter values are used until the next frame is displayed. To make the sound changing smoothly, we introduce a mechanism of interpolation between parameter values in consecutive frames in the animation. A simple linear interpolation is effective enough for this purpose.

Such a continuous sound is interesting but it sometimes sounds scary for audience, typically for young children. One method to make the installation enjoyable for wider audience is to modify the sound to be more musical, that is, the sequence of separated notes expressing melody, harmony and rhythm. By quantization of the continuous sound stream, it is divided into a sequence of separated notes each of which has a pitch of restricted set of frequencies.

The variation of pitches is chosen from the candidates of harmonic and alternated musical scales, such as pentatonic, major, blue notes, whole notes, diminished, and chromatic. In the piano and the other keyboard instruments and chromatic percussion the pitches are fixed, but it is usually possible to bend the pitches in the other types of instruments such as strings, horns, and talking drums. To add such flexibility that useful to make the sound more expressive, we introduced a partial frequency alternation of each note as to be gradually changing toward the continuous frequency extracted from the current frame. The rate of alternation is also a subject to adjust by human.

We implemented two alternative methods to make a rhythm. One method is to use a random sequence of three types of marks, note on, note off, and continued. These marks are applied in constant tempo in the order they are arranged when the system generates the sound. “Note on” starts a new note, “note off” stops the current note if it exists, and “continued” keeps the current on/off state. A sequence is arranged for each sound sources independently in the same number of timings such as 16 beats. It is easy to make an interesting rhythm pattern by random generation under constraints of appropriate probabilities for each mark. We implemented a pattern editor shown in figure 4 that allows the user to design his/her favourite rhythm pattern.



Tempo: ♩ = 120 Fluct.: 0.10

| 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11 | 12 | 13 | 14 | 15 | 16 | 17 | 18 | 19 | 20 | 21 | 22 | 23 | 24 | 25 | 26 | 27 | 28 | 29 | 30 | 31 | 32 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 𝄾 | – | – | 𝄾 | 𝄾 | – | 𝄾 | – | 𝄾 | – | – | ♪ | ♪ | ♪ | – | – | ♪ | – | – | ♪ | – | 𝄾 | – | 𝄾 | ♪ | – | ♪ | 𝄾 | – | – | ♪ | – |
| ♪ | ♪ | – | ♪ | – | – | – | – | 𝄾 | 𝄾 | 𝄾 | – | – | ♪ | – | – | 𝄾 | – | – | 𝄾 | – | ♪ | – | 𝄾 | – | – | – | – | – | – | – | – |
| ♪ | 𝄾 | – | 𝄾 | ♪ | – | – | – | – | ♪ | ♪ | – | – | 𝄾 | – | – | ♪ | 𝄾 | ♪ | – | ♪ | 𝄾 | 𝄾 | – | 𝄾 | 𝄾 | – | ♪ | – | – | ♪ | – |
| 𝄾 | – | ♪ | – | ♪ | ♪ | ♪ | – | – | ♪ | – | – | – | 𝄾 | – | – | ♪ | ♪ | ♪ | – | – | ♪ | – | ♪ | – | 𝄾 | ♪ | – | – | – | – | ♪ |
| 𝄾 | – | 𝄾 | – | ♪ | 𝄾 | ♪ | 𝄾 | ♪ | ♪ | – | – | ♪ | 𝄾 | ♪ | – | ♪ | – | 𝄾 | – | – | ♪ | ♪ | – | ♪ | – | – | – | – | – | – | ♪ |
| 𝄾 | ♪ | – | ♪ | – | ♪ | ♪ | ♪ | 𝄾 | ♪ | 𝄾 | – | – | – | – | – | – | – | – | – | ♪ | – | – | – | ♪ | ♪ | ♪ | ♪ | – | – | – | 𝄾 |
| 𝄾 | – | 𝄾 | – | – | – | 𝄾 | ♪ | ♪ | – | – | 𝄾 | – | – | – | 𝄾 | – | ♪ | – | – | ♪ | ♪ | – | – | ♪ | – | – | – | ♪ | ♪ | – | 𝄾 |
| ♪ | – | – | – | – | ♪ | ♪ | 𝄾 | 𝄾 | ♪ | – | – | ♪ | – | – | ♪ | ♪ | 𝄾 | – | – | – | ♪ | – | – | – | – | ♪ | – | – | – | – | 𝄾 |

♪ 30% 𝄾 20% – 50%

Figure 4. *GUI of rhythm editor.*

The other method is to start a new note when the image changes faster than a threshold speed if the time longer than minimum limit of note duration has passed since the proceeding note started. This method does not produce a pattern of constant tempo but a type of synchronized rhythm with the motion in the frame images. It is more effective to emphasize the impression of visuals than the first method.

Another important factor for musical notes is the envelope that determines the time alternation of loudness for a single note. Usually, an envelope is defined by some parameters to draw a relation between time and amplitude, but we use only two parameters for the time delay of attack and the length of tail in the current implementation as shown in figure 5. If the delay is long, the note sounds like bowing on strings or normal blowing on horns. It sounds pizzicato on strings, piano or guitar, when the delay is short. The amplitude of each note decays exponentially by multiplying a coefficient. The long tail sustains the sound for long time until it explicitly stops or the next note starts.

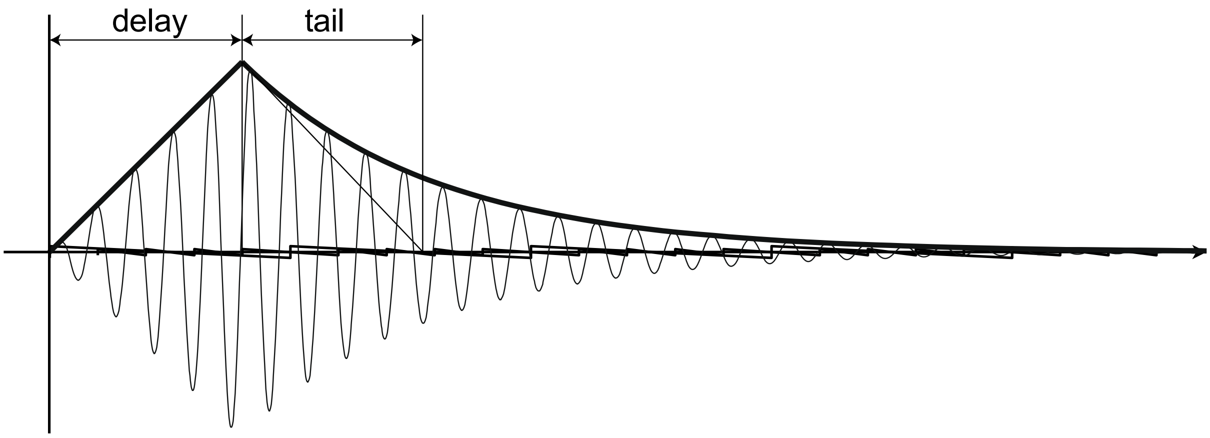
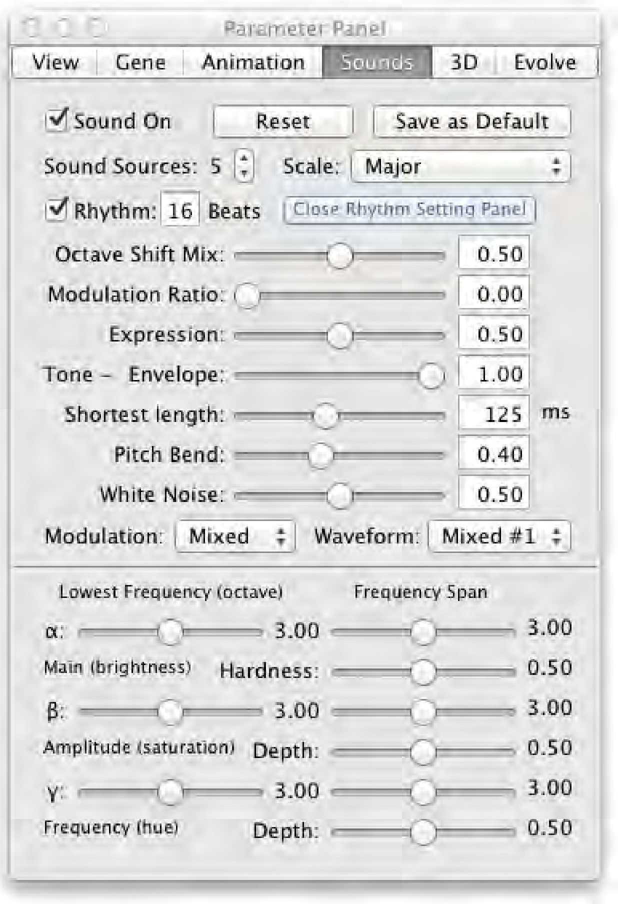


Figure 5. *Parameters for envelope.*



☑ Sound On

Sound Sources: 5

Scale: Major

☑ Rhythm: 16 Beats

Octave Shift Mix: 0.50

Modulation Ratio: 0.00

Expression: 0.50

Tone – Envelope: 1.00

Shortest length: 125 ms

Pitch Bend: 0.40

White Noise: 0.50

Modulation: Mixed

Waveform: Mixed #1

| Lowest Frequency (octave) | | Frequency Span | |
|---|---|---|---|
| α: | 3.00 | | 3.00 |
| Main (brightness) | | Hardness: | 0.50 |
| β: | 3.00 | | 3.00 |
| Amplitude (saturation) | | Depth: | 0.50 |
| γ: | 3.00 | | 3.00 |
| Frequency (hue) | | Depth: | 0.50 |

Figure 6. *GUI for parameter alternation for sound synthesis.*

## 6. Adjustable parameters

As described in the former sections, the user is allows to adjust some parameters for sound synthesis to add his/her favourite taste. These parameters include the frequency ranges of pitch, amplitude modulation and frequency modulation within the range of audible frequencies for human ears. The bandwidth of low-pass filter and the modulation strengths of both amplitude and frequency are targets alternation by the standard deviations of brightness, saturation and hue as described in former section 4, but their maximum values are also adjustable by the user. The statistical features are ignored when these maximum values are set to zero.

Figure 6 shows the window image of graphical user interface to adjust those values. It contains a stepper for the number of sound sources, a popup button to choose the musical scale, a checkbox to determine if rhythmic or synchronous, and a text field to input the number of beats for random rhythm pattern. The slider entitled “octave shift mix” is to indicate the mixing ratio between original tone and the tone of which pitch is shifted in octaves as described in former section 2.2. The shifted tones are not mixed when this value is 0, and the sound is constructed only from the shifted tones when it is 1. The next slider entitled “modulation ratio” is to control the ratio of strength of modulation between the original tones and the shifted tones. As described above, the maximum strengths of each modulation are limited by the other parameters, but this parameter is to multiply one more coefficient either the original tones or the shifted tones. The combination of coefficients for the original tones and the shifted tones becomes one and zero when this parameter is 0, one and one when it is 0.5, and zero and one when it is 1. The sliders entitled “expression,” “envelope,” “pitch bend” and “white noise” are to indicate how much each effect should be applied. If the expression is set to 0, the motion speed in the visuals does not affect the amplitude of the sound.

Two popup buttons under these sliders are to alternate the method to apply the parameter values for each sound source. It would be better to allow those parameters separately for each sound sources in order to produce richer variation of sound output, but we would need to solve the problem of complicated operation on a large number of parameters by the user. To avoid this problem, the popup button entitled “modulation” allows choosing the direction of parameter value alternation corresponding to the statistical features extracted from the image. The parameter value increases when the feature value increases in the normal mode, but it goes in opposite direction in the reversed mode. In addition to the choices applying the same direction for all sound sources, the mixed mode is allowed so that the directions are alternated between odd columns and even columns. The other popup button entitled “waveform” allows changing the combination of methods for waveform synthesis for each sound source. It contains four alternative choices to apply the first method in section 3 for all sound sources, the second method for all, the first method for odd columns and the second method for even columns, and the first method for even columns and the second method for odd columns.

All of these parameters are registered as the properties belonging to the sound controller object in the framework of AppleScript, so that another application software is allowed to refer and change the values via inter-software communication. This feature is useful for a batch process of automatic production to construct the audio track in a movie file as a final product of the animation. By choosing random settings for each production, it is possible to make audio tracks of various different favours.

## 7. Concluding remarks

A method to automatically produce a sound from real-time animation was described above. The design of the mechanism is based on mappings the impression from the visuals to the audio using statistic analysis of the image and motion and a parametric method of sound wave synthesis. Through several times of application, this method seems successful to achieve an acceptable quality.

There is a wide range of possible variations for this kind of mappings. There are many alternative methods to extract the visual features from the motion image. Our second method to synthesise the waveform described in section 3 is similar to the method proposed by Dannenberg *et al* [6], but the combination of the other features are of course useful to produce richer sounds. If we would employ a method to trace a movement of typical image fragment, it could be possible to implement a moving sound source of which channel balance follows the position in the sound scape.

There are also alternative methods for sound synthesis. One obvious method is to use a set of audio filters connected each other. The features of visuals would be applicable to the filters as the parameters such as the bandwidth for low-pass and high-pass filters, decay and delay for echo, amplitude for distortion, and so on. Another possible method is to use MIDI instruments. It is easy to organize a virtual band by arranging a number of instruments to apply a melody, harmony, rhythm and other controls to play a type of music automatically composed in real-time. Some of the sound programs embedded in the MIDI standard other than musical instruments are also useful to produce an ambient sound.

It must be helpful to develop a type of graphical user interface to arrange the image filters, audio filters, and other processing units in order to examine a large number of possible combinations, similarly to some programming tools such as MAX and Pure Data.

The proposed method was designed for evolutionary animation, but it is also applicable for sound synthesis from any type of moving visuals. It might not be suitable for a video image captured in ordinary sceneries because the viewer would expect a sound of physical objects shown in the scene. However, we think there are many possibilities of potential applications for art and entertainment such as an interactive installation, a visual audio game, movie authoring, and so on.

We have been organized three types of application of this system so far, a live performance of real-time breeding, an installation of automatic evolutionary animation, and a web-based art of daily evolutionary animation. The first one of the previous version was performed in the Generative Art Conference in Rome in 2011, the second one is exhibited in the same conference of 2012 in Lucca, and the third one is exhibited on the Internet accessible at the web page of “SBArt4 Daily Evolved Animation on WebGL [7].”

This web site is designed using the latest web technologies of HTML5, JavaScript and WebGL. You can enjoy ten new animations everyday without any image loss by compression on the large screen if your machine has a recent product of graphical processing unit for high definition TV. The captured video of live performance, the summarized explanation of exhibition, and daily and weekly digests of daily productions are also available at the author’s YouTube channel [8]. The software SBArt4 is runnable on MacOS X 10.6 or later. The binary code is available from the “SBArt4 Home Page [9].” We hope that it will provide some inspiration to as many persons as possible.

## References

1. Tatsuo Unemi: SBArt4 as Automatic Art and Live Performance Tool, in C. Soddu ed. Proceedings of the 14th Generative Art Conference, pp. 436–447, December 4–7, Rome, Italy, 2011.

2. Tatsuo Unemi: SBArt4 for an Automatic Evolutionary Art, Proceedings of the IEEE World Congress on Computational Intelligence (WCCI 2012 – IEEE CEC 2012), pp. 2014–2021, June 10–15, Brisbane, QLD, Australia, 2012.

3. Scott D. Lipscomb and Roger A. Kendall: Perceptual Judgement of the Relationship between Musical and Visual Components in Film, Psychomusicology, No. 13, pp. 60–98, 1994.

4. William Moss, Hengchin Yeh, Jeong-Mo Hong, Ming C. Lin and Dinesh Manocha: Sounding Liquids: Automatic Sound Synthesis from Fluid Simulation, ACM Transactions on Graphics, Vol. 28, No. 4, Article 110, December 2009.

5. Yoshinori Dobashi, Tsuyoshi Yamamoto and Tomoyuki Nishita: Synthesizing Sound from Turbulent Field using Sound Textures for Interactive Fluid Simulation, Eurographics, Vol. 23, No. 3, 2004.

6. Roger B. Dannenberg, Barbara Bernstein, Garth Zeglin and Tom Neuendorffer: Sound Synthesis from Video, Wearable Lights, and 'The Watercourse Way,' Proceedings: The Ninth Biennieal Symposium on Arts and Technology, New London, CT: Ammerman Center for Arts and Technology, pp. 38–44, 2003.

7. Tatsuo Unemi: SBArt4 Daily Evolved Animation on WebGL, http://www.intlab.soka.ac.jp/~unemi/sbart/4/DailyWebGL

8. — : une0ytb – YouTube, http://www.youtube.com/user/une0ytb

9. — : SBArt4 Home Page, http://www.intlab.soka.ac.jp/~unemi/sbart/4/