TongArk: a Human-Machine Ensemble

Prof. Alexey Krasnoskulov, PhD.
Department of Sound Engineering and Information Technologies, Piano Department
Rostov State Rakhmaninov Conservatoire, Russia
e-mail: avk@soundworlds.net

Abstract

This work explains a software agent that applies the musical task of accompaniment. The author investigates computer system possibilities in creating and transforming musical material under artistic and procedural circumstances built on some perception and analysis of a sound realization of the creative concept, and the external manifestation of emotions by a human agent.

Software reveals the interactive duet "human / software agents" in which the latter perceives the part performed by the human agent, and makes an ambient sound in its own musical part using the genetic algorithm. The human agent controls the software agent's performance by means of some change in emotions shown on the face.

The software agent estimates the affective state of the human performer using face recognition to generate accompaniment. Every single change of any affective state is reflected in updating the timbre and reverberation characteristics used by a computer system of sound elements as well as in transforming the sounding in all of the software agent's part.

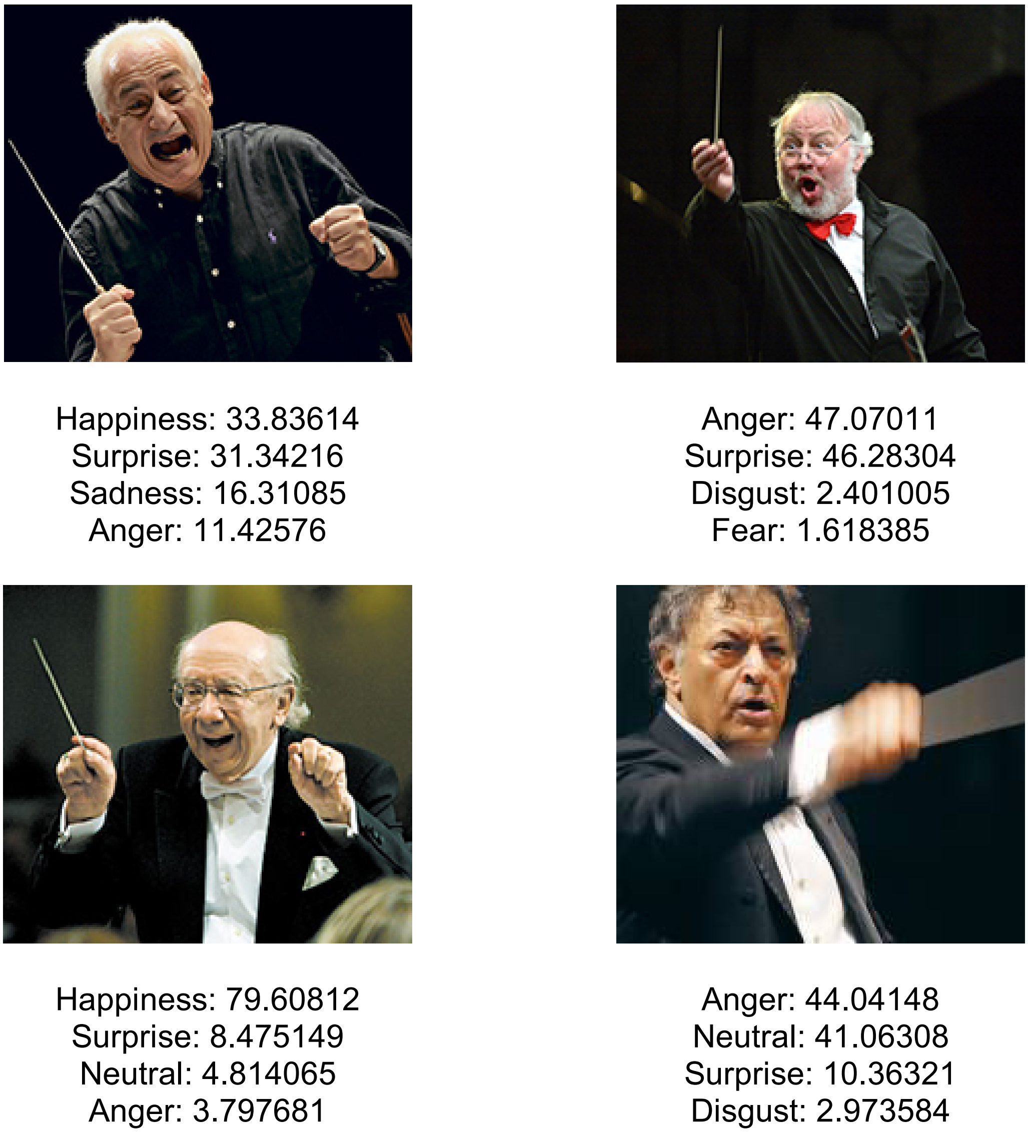
Researches on peculiarities of the correlation between sounds of a definite pitch and/or timbre and the affective states caused by them together with the listening tests carried out within the framework of this project – all these made possible to develop a structure of the correlation between the key emotions and frequency and space parameters of sound.

The system design combines Affective Computing, GAs, and machine listening. This article describes the algorithmic processes of the system and presents the creation of two music pieces. Details of the work and possibilities for future work are given.

Introduction

The incentive motivation for creating the *TongArk* project is the need for real-time communication between the human agent and the software agent when the former performs his/her part on the piano. Obviously, any professional pianist often uses not only his/her hands but both feet as well pressing and releasing the soft and sustain pedals, accordingly. In this way, special gestures used to communicate with a software agent turn out to be rather awkward, and the use of various sensors may cause some discomfort. Consequently, the affective states expressed on the face may become one of the idle but effective communication media.

In the music practice, a face expression is sometimes an extremely efficient communication medium. This is especially evident in the work of symphony orchestra conductors who always make use of mimicry to convey their intentions to the performing orchestra more efficiently (see Figure 1).



Happiness: 33.83614
Surprise: 31.34216
Sadness: 16.31085
Anger: 11.42576

Anger: 47.07011
Surprise: 46.28304
Disgust: 2.401005
Fear: 1.618385

Happiness: 79.60812
Surprise: 8.475149
Neutral: 4.814065
Anger: 3.797681

Anger: 44.04148
Neutral: 41.06308
Surprise: 10.36321
Disgust: 2.973584

Figure 1. Conductors' emotions identified by Microsoft Emotion API (four most probable affective states in the list of eight, with the "weight" of each corresponding emotion)

Certainly, in the performing art (by pianists, violinists, flautists, etc.) conveying intense emotions by face is more than an exception as this communication medium is optional, often excessive and even improper in the ensemble playing. Nevertheless, while correlating with the software agent this way of communication is likely to become an advantageous process of interaction.

Emotion recognition also makes a computational system manage the software agent's part during the creation of a musical composition (Winters, Hattwick and Wanderley 2013). Nowadays, face recognition is becoming increasingly prevalent as a controller because neural network algorithms and their software realizations are growing up rapidly and getting much faster and precise than ever before. By using face recognition (with Microsoft Cognitive Services (MCS) Emotion API) and the genetic algorithm, the *TongArk* project explores a real-time composition where a human performance leads to the software agent's response.

Preparatory Work

The key feature of the *TongArk* generative process is emotion recognition that controls the frequency range of current timbres and the reverberation type in the software agent's part. The sound equalization is based on a correlation between emotion characteristics and the pitch and dynamics of the sound. According to some researches (Chau, Mo and Horner 2014, 2016; Chau, Wu and Horner 2014; Wu, Horner and Lee 2014), specific emotions are caused by sounds of some specific frequency and dynamic ranges. In our case, it is important that such correlation would also appear between emotions and complex sounds in the tempered twelve-tone system. The above-mentioned researches state ten emotional categories (Happy, Sad, Heroic, Scary, Comic, Shy, Romantic, Mysterious, Angry, and Calm), but only three of them correlate with the Emotion API list of emotions (Happy – Happiness, Sad – Sadness, Angry – Anger). Therefore, a listening test has been developed to include the other five emotions from Emotion API (Contempt, Fear, Disgust, Surprise, and Neutral).

Twenty-two professional musicians, both teachers and students of the sound engineering faculty took part in the test. For this, piano sounds from the Native Instruments Akoustik Piano VST library were recorded in the range from C1 to C8. The test results gave a pitch range for the above-stated five emotions as shown in Figure 2.

Emotion	Pitch Range
Happiness	C5 – C7
Sadness	C1 – C8
Contempt	C1 – C6
Fear	C1–B1, C7–B7
Disgust	C4– C6
Surprise	C5 – C8
Anger	C1 – C3
Neutral	C4 – C8

Figure 2. Pitch range of the Emotion API list of emotions

Another example of the correlation between emotions and sounds are effects of the reverberation time on the emotional characteristics (Mo, Wu and Horner 2015). In order to correlate the reverberation characteristics with the Emotion API list of emotions, there was developed another listening test. In this test, there were 24 reverberation presets (using FMOD DSPs) with three different timbres. The same group of listeners chose the most appropriate reverb type for each emotion as shown in Figure 3.

Emotion	Reverberation Preset
Happiness	AUDITORIUM
Sadness	CONCERT HALL
Contempt	HANGAR
Fear	QUARRY
Disgust	OFF
Surprise	STONE CORRIDOR
Anger	UNDERWATER
Neutral	GENERIC

Figure 3. Correlation between the Emotion API list of emotions and FMOD sound library's reverberation presets

Implementation

The implementation is in C# using the Exocortex.DSP library[1] for FFT and NAudio library[2] for sound input. A snapshot from a web-camera is taken at regular intervals and then passed to the MCS cloud using Emotion API. The human agent's audio output signal is transmitted directly to a PC and simultaneously recorded, both as a wave stream from the microphone and MIDI messages. The sound engine is written using FMOD sound library[3] while the generative algorithm mostly uses its DSP (low-pass and high-pass filters) together with reverberation presets. The same library is employed for the pre-prepared sound playback.

The human agent's sound is captured from the microphone, and after FFT its spectrum (bands amplitude values) becomes a fitness function of the genetic algorithm (Miranda and Biles 2007). Therefore, the software agent's performance is never the same as the human performance but it always tries to achieve and/or copy it. In addition, all sounds are considered to be "neutral" outside any emotional context.

In the FFT output there is an array of complex numbers, each of them containing an amplitude value for a definite spectral band. Out of this array there are selected some amplitude values of those frequencies which fit the sounds of an equally tempered scale. As a result, there is an array of 84 amplitudes from C1 (32.703 Hz) to B7 (3951.044 Hz) which becomes a fitness function of the genetic algorithm. The population consists of 1000 members, each of them representing a volume values array of every downloaded sound element (one value per sound element making 84 values in all). While launching the program all the arrays are initialized with zeroes. Subsequently, the population runs phases of the two-point crossover and mutation with the pre-determined probability of event. For instance, the 0.8 pre-determined parameter of the crossover means the crossover occurring among 80% members. The parameters of the crossover and mutation are defined independently at the initialization and may change during the performance. It is also possible to define a mutation step which determines how smoothly or suddenly the mutation occurs. The speed of approximation to the fitness function standard is determined by the frequency of the epoch alternation as well (in the present project, the optimal quantity turns out to be 5 to 50 epochs per second). However, since the fitness function gets renewed frequently the selection process continues until the human agent's sounding breaks for a long time. If that is the case, the fitness function becomes an array of zero values. This makes the genetic algorithm create generations that would exactly fit the fitness function.

During the performance process, Emotion API returns the list of eight emotions (happiness, fear, disgust, etc.) with their "probabilities" or "weights" in every specific time interval (in the predetermined range of 1 second to 5 minutes). Therefore, when the human agent changes an emotion on his/her face and the software detects this, the list of emotions and their "weights" also change. Immediately, another group of sounds smoothly replaces the current group. At the same time, the software agent is making a real-time transformation of each sound using low-pass and high-pass filters, and the reverberation type changes globally in the software agent's part as shown in Figure 4.

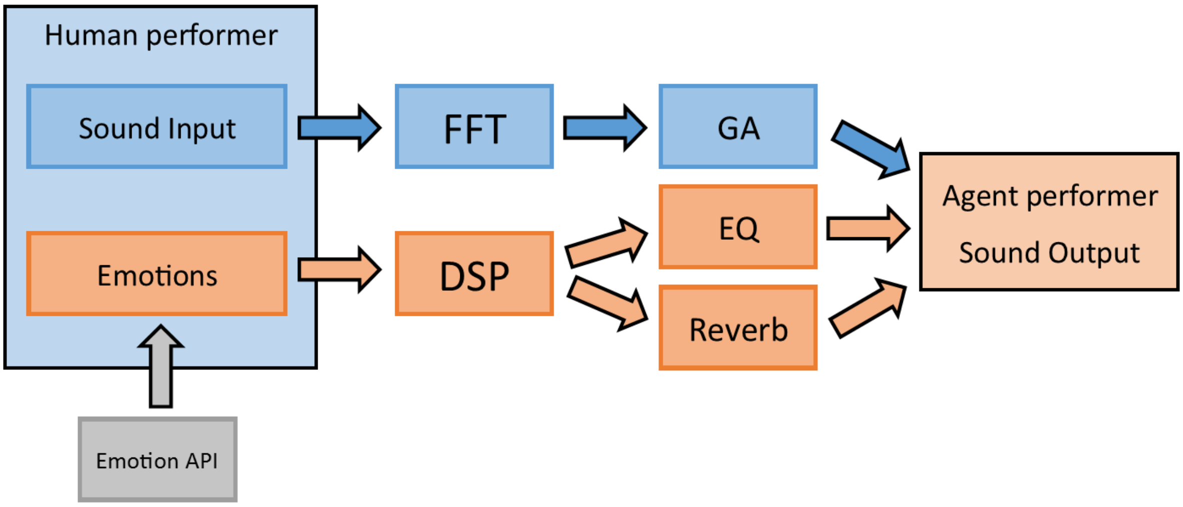


Figure 4. Scheme of the generative process

Demonstration

BigDukkaRiver[4] and *Ngauruhoe*[5] are examples of the system in action. For both compositions there were created four groups of sound samples. In the second composition, samples from one of the groups were "one-shot" samples (see Figure 5).

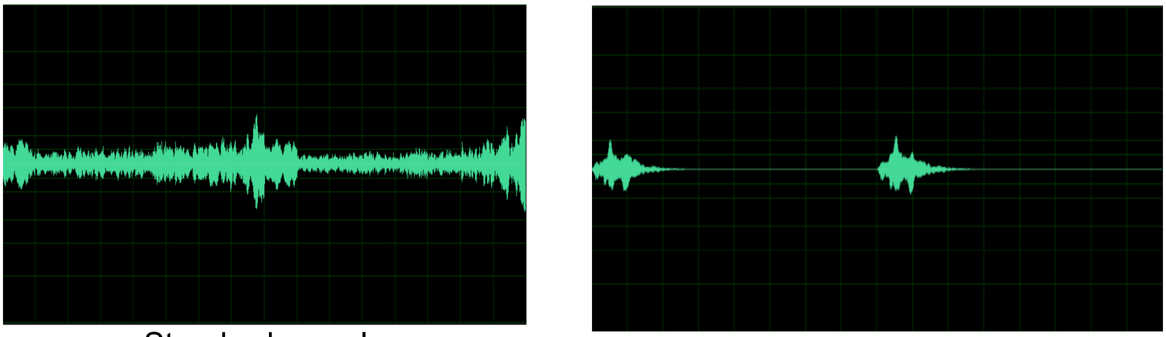


Figure 5. Standard and "one-shot" samples

All of the sound elements within one group are equal in their duration. In order to avoid the resulting monotony in the software agent’s part, the sound elements are reproduced with some delay one from another (randomly from 50 ms to 1 sec). The application of such methods to one-shot samples in combination with changes in volume determined by the genetic algorithm often result in generating interesting rhythmic structures.

Each group contained 84 sounds of some particular timbre in the tempered twelve-tone system (C1-B7). Sounds from different groups randomly filled the array of 84 sounds which was to be active at some predetermined time. The pre-prepared sounds were made using Camel Audio Alchemy and Steinberg HALionOne VSTs, the timbre of the piano was Native Instruments Akoustik Piano (Concert Grand D).

In the project, the human agent played the digital piano, and the sound stream was transformed in the array of amplitudes by the FFT function (the FFT window size is 4096 samples, the sample rate is 44100 Hz). This array became a fitness function of the genetic algorithm, and the result of each epoch controlled the volume of each sound in the software agent's part. The genetic algorithm implementation uses different parameters of the crossover and mutation as shown in Figure 6.

Composition	Epochs (qty per second)	Cross-over (0 to 1)	Mutation (0 to 1)
BigDukkaRiver	20	0.5	0.7
Ngauruhoe	6	0.3	0.8

Figure 6. Genetic algorithm settings for the musical pieces

For the reverberation type, only the emotion with the highest percentage of recognition is used. The “Neutral” emotion is ignored, keeping the previous settings active. In the final sound mix, the Great Hall reverberation was added to the human agent's part.

Future Work

The development of *TongArk* is ongoing. Some future work may include such improvements as acoustic recording, live performance and creating a system for numerous performers (both human and software agents). Today, the main problem is Emotion API itself, at least until significant improvements in the Oxford project are made. The main issue is that emotion recognition is first quite unstable and, second, it gives a result with some evident latency. A low quality of recognition forces the human agent to tense facial muscles and even grimace occasionally. Emotion API defines many facial expressions as the Neutral emotion regardless the human agent's actual emotions at that point of time. This motivates to use or develop another neural network and train it to recognize different expressions of a particular performer more precisely and in detail.

Notes

[1] http://www.exocortex.org/dsp
[2] http://naudio.codeplex.com
[3] http://fmod.org
[4] http://www.soundworlds.net/media/BigDukkaRiver.wav
[5] http://www.soundworlds.net/media/Ngauruhoe.wav

References

Chau, C.J., Mo, R., and Horner, A. (2014) *The Correspondence of Music Emotion and Timbre in Sustained Musical Instrument Sounds*. Journal of the Audio Engineering Society, vol. 62, no. 10, pp. 663–675, 2014.

Chau, C.J., Mo, R., and Horner, A. (2016) *The Emotional Characteristics of Piano Sounds with Different Pitch and Dynamics*. Journal of the Audio Engineering Society, vol. 64, no. 11, pp. 918-932, 2016.

Chau, C.J., Wu, B., and Horner, A. (2014) *Timbre Features and Music Emotion in Plucked String, Mallet Percussion, and Keyboard Tones*. In: Proceedings of the 40th International Computer Music Conference (ICMC), pp. 982–989, 2014.

Evolutionary Computer Music (2007) Miranda, Eduardo Reck; Biles, John Al (Eds.) London: Springer, 2007.

Mo, R., Wu, B., and Horner, A. (2015) *The Effects of Reverberation on the Emotional Characteristics of Musical Instruments*. Journal of the Audio Engineering Society, vol. 63, no. 12, pp. 966-979, 2015.

Winters, R. M., Hattwick, I. and Wanderley, M. M. (2013) *Emotional Data in Music Performance: Two Audio Environments for the Emotional Imaging Composer*. In: Proceedings of the 3rd International Conference on Music & Emotion (ICME3), Jyväskylä, Finland, 11th - 15th June 2013. Geoff Luck & Olivier Brabant (Eds.). University of Jyväskylä, Department of Music.

Wu, B., Horner, A., and Lee, C. (2014) *Musical Timbre and Emotion: The Identification of Salient Timbral Features in Sustained Musical Instrument Tones Equalized in Attack Time and Spectral Centroid*. In: Proceedings of the 40th International Computer Music Conference (ICMC), pp. 928–934, 2014.