Towards a real-time mapping of Finger Gesture to Sound

V. Matsoukas[a], S. Manitsaris[b] and A. Manitsaris[a]

[a]University of Macedonia, Dept of Applied Informatics, MTCG Lab, 156 Egnatia Street, 54006 Thessaloniki, Greece
[b]Laboratoire Signaux, Modèles et Apprentissage Statistique, 10 rue Vauquelin, 75231 Paris cedex 05
vmats@uom.gr

The scope of this research is to develop a system for real-time and continuous gesture following and recognition, given emphasis to the finger approach of gesture control of sound. The proposed methodology can be implemented in a low cost computer vision system, which provides the means to facilitate an interactive composition of contemporary music. A simple web camera is used to capture finger musical gestures in 3D space without any tangible instrument. Afterwards we use the PianOrasis system, which recognizes simultaneously the gestures of all five fingers by extracting meaningful features from each frame sequence. In order to achieve this in real-time, the "gesture following" method is applied. Hidden Markov Models are used to update "continuously" the gesture descriptors. We choose then to map high-level features of finger motions to low-level audio descriptors given emphasis to the temporal representation of mapping rather than the spatial. This is due to the need of real-time performance.

1 Introduction

In this paper we propose a computer vision methodology for control of the sound synthesis, which integrates finger gesture performed on a musical instrument, such as piano or woodwind instruments, or just in the surrounding space. We use image analysis techniques for detecting and identifying the fingertips on a video and stochastic modeling techniques for finger gesture recognition and prediction. There is also a methodology proposed for real-time finger gesture control of the sound, a part of which is actually implemented.

The term "digital musical instrument" is being used in the literature to represent an instrument that includes a separate gestural interface (or gestural controller unit) from a sound generation unit. Both units are independent and related by mapping strategies [1]. Once the gesture is visually recognized several parameters of instrumental sounds can be dynamically controlled. Obviously, as mentioned in [2], this separation of the DMI into two independent units is potentially capable of extrapolating the functionalities of a conventional musical instrument, although basic interaction characteristics of existing instruments, such as tactile/force feedback, may be lost and/or difficult to reproduce.

2 Related work

Recent developments show an increasing interest in controlling musical performance with human gestures. There have been several approaches proposed in this research field. Most of the research work is focusing primarily on the analysis and recognition of human gestures like in [3], [4] and [5]. Also data acquisition is an important issue. Most of the work done here is using expensive and non-portable equipment like a 3D optical motion capture system or a sensor acquisition system. Such equipment is expensive, complicated and not easily transportable, which makes it impractical for musical performances. Additionally, it should be mentioned that the above systems are usually able to recognize global gestures, such as body postures or hand gestures. They do not perform a finger gesture recognition for each finger individually in real-time. For the above reasons, we propose a standard video camera, which is simple and cost-effective. The video camera do not put constraints to the performer since his/hers fingers can move freely.

Real-time gesture recognition is also important for our system and the most recent work in this area is presented in [6], where HMMs are used for a system, which "continuously" updates parameters characterizing the performance of a gesture. Our work will be based on this "gesture following" method.

Finally the mapping sound to gesture is significant for our system. In order to achieve this we will use a set of Max/MSP externals and abstractions, such as the MnM mapping toolbox developed in [7].

3 Methodology

3.1 System architecture

In order to develop our system for continuous real-time finger control of music we need to follow several stages. The first stage is the data acquisition. The proposed system uses an installation of a camera in front of a piano or on a table, so as to fully record finger motions in two dimensions as proposed in [8]. PianOrasis recognizes simultaneously the gestures of all the five fingers of a hand.

The second stage is the feature vectors extraction from the video frame according to the method proposed in [5]. (Figure 1). The following feature vectors are extracted from each frame: (a) the differences of the ordinates between the fingers and the centroid, (b) the abscissa of the fingers and (c) differences between the abscissas of adjacent fingers. The sound produced is also being recorded.

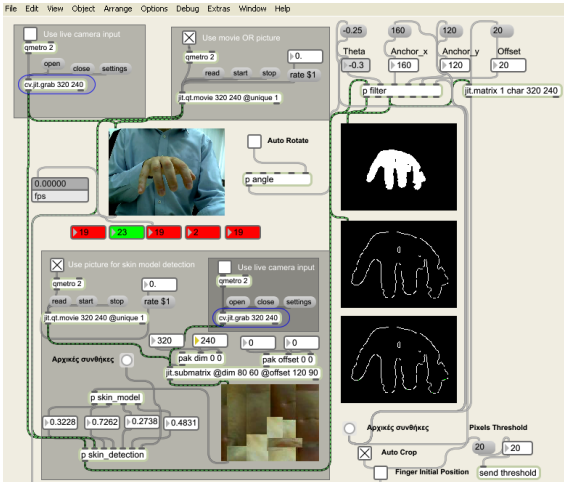

Figure 1: High-level feature extraction in Pianorasis:max Interface

A low-cost camera captures finger motions as streaming images. These images are a sequence of video frames, which are analyzed taking into consideration of a spatial model, in a determined time period. Pairs of notes and fingerings are produced. The prerecorded gestures along with their corresponding sound segments will be stored in a database. In [8], the video signal is imported into the computer and is processed using MATLAB Image

Acquisition Toolbox (PianOrasis). In our system we are currently implementing this procedure using Max/MSP in order to recognize finger gestures in real-time (PianOrasis:max). Currently PianOrasis performs static and dynamic finger gesture recognition while in PianOrasis:max only static recognition has been implemented and dynamic recognition is under development.

The next stage is the gesture following and recognition. The method proposed in [6], which relies on the modeling of multi-dimensional temporal curves using HMMs, will be used. The parameters used are "gesture time progression", which gives us information about where we are within the gesture and "gesture likelihood", which calculates the likelihood values between a performed gesture and pre-recorded gestures stored in the database. These parameters are computed by comparing the performed gesture with the modeled gestures, helping to predict its evolution. Since the algorithm of the "Gesture follower system" proposed in [6] works with any type of regularly sampled multidimensional data flow, we can use the 14 features extracted on the second stage to achieve our goal. At this stage we will also use the sonification feedback technique proposed by [18]. According to this work the sonification feedback will be an audio display, of the changing probabilities and observed states within the HMMs. A Gaussian function of the distances will be used to penalize large distances between gesture models

The final step is the mapping between gesture and sound. As referred to in [9] there are many mapping strategies that are separated in three different classes: one-to-one, one-to-many, many-to-one. Complex mappings, like many-to-many, can be built by combining these classes. Such complex classes seem to be more satisfactory after a learning phase, than one-to-one mappings. We concatenate high-level video features extracted from PianOrasis with low-level audio features. Some of the major principles describing sound characteristics are log of energy, spectrum balance, peak structure match, spectral centroid, RMS, RollOff, flux etc. We intend to use some of these major principles for our system. We will also record natural and instrumental sounds and perform noise reduction, sound-processing and audio feature extraction as proposed in [17]

The architecture of the proposed system is presented in Figure 2:

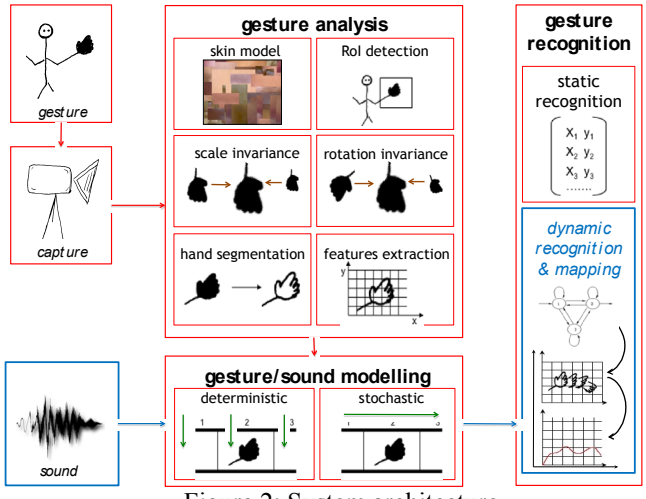

Figure 2: System architecture.
In red: already implemented; in blue: not yet

In order to implement complex mappings the MnM patch has been developed in [7] and will be used in our system. This toolbox is a series of Max/MSP externals and abstractions based on the FTM patch. It consists of a set of modules providing basic linear algebra, mapping and statistical modeling algorithms such as PCA, GMMs and HMMs.

3.2 Dynamic recognition

The dynamic recognition of the finger gestures is achieved via the PianOrasis system. As described in [5] the system uses a web cam to capture axes X, Y of the finger movements. X gives us the information about the location of the suitable region of keys and Y refers to detecting preparation and key push. Image segmentation is based on the human skin color that is dermal regions are detected based on a predefined skin model and the captured image is divided into regions including only the pianists hand. Afterwards the hand contour is extracted and the centroid of the hand is calculated for every frame. The local maxima are detected which give us five points matching to the five fingertips. Finally the feature vectors are extracted for each frame by calculating a) the differences of the ordinates between the fingers and the centroid, (b) the abscissa of the fingers and (c) the differences between the abscissas of adjacent fingers.

3.3 Gesture-sound mapping techniques

"Gesture-sound mapping" is called the procedure which correlates the gesture input data with the sound control parameters. Several approaches of gesture-sound mapping have been proposed in [9], [10].

A technique for geometric control is presented in [11] in which a mixture of Gaussian kernels maps a two dimensional control space into an M-dimensional sound space. Also [12] presents a technique for mapping from two dimensional control space (on-screen mouse position) into M parameters of sound processing.

In order to implement the gesture-sound mapping procedure we need first to decide, which gesture characteristics and sound synthesis variables are we going to use, that is, answer the question "what to map where" and secondly how this is going to be accomplished. In our system we choose to map high-level features of finger motions to low-level audio parameters and use the proposed method in [7], where mapping is considered as an operator that expresses each point in sound parameter space as a linear combination of the N parameter values of a given control input.

We also need to decide whether we will use explicit or implicit mapping. In explicit mapping, the mathematical relationships between input and output are directly set by the user. On the contrary, indirect mapping generally refers to the use of machine learning techniques, implying a training phase to set parameters that are not directly accessed by the user [13]. Our work emphasizes on the temporal representation of mapping rather than the spatial. This is due to the need of real-time performance. In order to achieve real-time gesture-sound mapping the user should be able to dynamically modify the control parameters in order to imprint any corresponding time behavior of the sound evolution as proposed in [13]. We are obviously talking about implicit mapping.

Van Nort et al. give in [14] a mathematical formulation of gesture-sound mapping. According to this work if the

mapping is described by a series of discrete couples of vectors $\{X_i, Y_i\}$, where $X_i \subset R_n$ (gesture parameter space) and $Y_i \subset R_m$ (sound parameter space) then mapping can be described as a function g: $R_n \rightarrow R_m$ and it can be seen as an interpolation problem. Other techniques like neural networks for many-to-few mapping and Principal Components Analysis or Canonical Correlation Analysis for quantitative analysis of the gesture-sound relationship are being used.

As mentioned though in [15] both variance-based methods suffer from a lack of temporal modeling which is important for real-time performance. Therefore suggests that HMM-based methods should be used in order to model the time profiles of the parameters since they allow for the temporal modeling of a sequence of incoming events.

After considering all the above-mentioned techniques and taking under consideration our requirements the most appropriate method to be used in our system is described in [13]. We use this method along with further extensions. To implement our approach we use the MnM toolbox of Max/MSP developed by F. Bevilacqua et al. in [7] and also used in [13]. The idea is to describe mapping procedures as a combination of relatively simple matrix operations. Consider a matrix X containing n gesture parameters and matrix Y containing m sound parameters. A simple mapping operation corresponds to a matrix multiplication with a matrix A of dimension (mxn).

$$Y = A * X \qquad (1)$$

In case of layered mapping as described in [2] we can consider for example three matrices and the mapping can be easily defined as a series of matrix multiplications:

$$Y = (A * B * C) * X \qquad (2)$$

In [7] there are also other matrix operations considered like element by element multiplication or combining a matrix with a function f applied to each element of the matrix which can be combined to design non-linear mappings.

In our system we can use the idea of matrices but we cannot consider linear mapping due to real-time performance. Our system needs to continuously update the parameters characterizing the performance of a gesture and therefore we will use the "gesture follower" approach presented in [6] and [13], where HMMs are used to achieve this goal. Our system gets a stream of 14 high-level features extracted from each video frame as input. These features describe fully the movements of the 5 fingers. Assuming the data is valid, input lists are interpreted as control space values; upon receiving a list, the input is mapped to a new output in sound space as proposed in the "gesture follower" system.

The preprocessed data is stored in a Matrix A of dimension NxM, where each gesture g_t is characterized by a vector $x(t)=(x_1, x_2,\ldots, x_m)$, where m=14. There are N such vectors for N different time moments from t=0 to t = (N-1)*Δt, where Δt is the time interval.

After storing the input data we need to train our system during a learning procedure that uses machine learning techniques and in particular HMMs.

Each gesture template is associated to a state-based structure: each data sample represents a "state" [16]. A probability density function is associated to each state, setting the observation probability of the data. This structure can then be associated to an HMM

As proposed in [13] during the learning procedure at least one gesture is stored in a Matrix A (NxM). For every matrix element there is a corresponding mean value mi of a normal probability function bi, which corresponds to the observation probability function. Variance and a global scaling factor, which operates on all the variance values, can also be set by the user.

For the online processing [13] suggests calculating 2 parameters to be used for the mapping procedure: a) time progression index and b) likelihood value. The first parameter is defined by computing the time warping during the performance.

The likelihood value is calculated by measuring the similarity between the gesture being performed and the templates as proposed in [15].

Finally, as mentioned before, the temporal mapping procedure needs to be discussed. During this procedure there are relationships between the gesture and audio temporal profiles defined. As discussed in [13] the time progression index helps us here to synchronize the gesture parameters and the sound parameters. The authors show an example of temporal mapping where hand acceleration and audio loudness are mapped together. There are though no limitations since any kind of gesture and sound parameters can be used.

3.4 Implementation

We implemented our system in 2 environments. Matlab has been used to implement PianOrasis where, as mentioned in 3.1. of this paper, feature vectors are extracted and static and dynamic recognition is available. We also used the Max/MSP environment to implement our system which currently only performs static gesture recognition, namely PianOrasis:max.

The gesture-sound mapping procedure will be implemented using Max/MSP. For this purpose we will use the FTM library in Max/MSP, which enables matrix handling and also the MnM patch to model our mapping approach.

As explained in [13] the main purpose of FTM is the representation and processing of sound, music and gesture data in Max/MSP extending the data types processed and exchanged by the Max/MSP modules. The implemented classes include matrices, dictionaries, sequences, break point functions and tuples.

The FTM fmat class implements a simple two-dimensional matrix of floating-point values providing methods for inplace matrix calculations and data import/export. An FTM track object allows for recording and playing of a stream of matrices as well as for the import/export of a stream in the SDIF file format.

The mat class acts as a 2-dimensionnal cell array of generic FTM objects, and in particular can handle matrices of fmat.

The externals of the libraries based on FTM use fmat as a generic representation for a variety of algorithms implementing analysis/synthesis, mapping, statistical modeling, machine learning and information retrieval. FTM enable to easily connect these algorithms in an application,

thus creating a tied link between gesture analysis and sound synthesis.

In the work of [13] the MnM patch, "Mapping is not Music", is presented as a set of Max/MSP externals based on FTM, taking advantages principally of the matrix classes fmat and mat. The construction of the mapping procedure is performed using both basic matrix operations from the FTM library and using the dedicated MnM set of externals and abstractions. Mapping can be thus built in a modular way. Different types of mapping approaches, including interpolation, regression and recognition are implemented.

The abstraction mnm.matmap is used to implement a multidimensional linear mapping. It can be seen as basic module to build complex n-to-m mapping.

The mnm.pca object performs Principal Component Analysis (PCA), which enables the reduction of the dimension of the effective control space, which can simplify the mapping procedure [11]. PCA can also be seen as a practical way to parameterize principal features of the control space

The objects mnm.matmap and mnm.pca are very simple to use and promising. Their main limitation resides in the fact that they model data linearly. However, such objects can be generalized for non-linear mapping using kernel methods.

4 Conclusion

We proposed a computer vision methodology for the recognition of the musical finger gestures performed in space without music instrument. The proposed method allows the recognition and prediction of finger musical gestures using machine-learning techniques following specific aims. It is vision-oriented to the image of the hand, without preliminary analysis, non-obtrusive, allowing the artist to feel free, with no need of special equipment and accessible and cost-effective, allowing its large-scale use. The proposed system can be used for educational purposes to help beginners. It can also be used for creating melodies and composing music without a musical instrument. Other possible applications, like virtual conducting without sophisticated sensor-based equipment, can also be considered as a further step going beyond finger control of sound and using the whole hand of the performer. The applications directly concern the contemporary music composition for performing arts, such as the contemporary musical theatre.

References

[1] S. Sapir, “Interactive digital audio environments: gesture as a musical parameter,” in Proc. *COST-G6 Conf. Digital Audio Effects (DAFx’00)*, pp. 25–30, (2000)

[2] M. Wanderley, P. Depalle: Gestural Control of Sound Synthesis, Invited paper, Proc of the IEEE, Vol. 92, No. 4, (2004)

[3] N. Rasamimanana, D. Bernardin, M. Wanderley, F. Bevilacqua, “String Bowing Gestures at Varying Bow Stroke Frequencies: A Case Study”, In *Lecture Notes in Computer Science*, Springer Verlag, vol. 5085, pp. 216–226, (2008)

[4] A. Hadjakos, E. Aitenbichler, B. Wetz, M. Muehlhaeuser, “Computer Analysis of the Indirect Piano Touch: Analysis Methods and Results”, In *3rd International Conference on Automated Production of Cross Media Content for Multi-channel Distribution*, ISBN 978-88-8453-677-8, pp. 33-38, Firenze University Press, (2007)

[5] S. Manitsaris, “Gesture Recognition in Music Interaction via Computer Vision”, *9th IASTED International Conference on Visualization, Imaging and Image Processing (VIIP 2009)*, Campridge, United Kingdom, 13-15 July, (2009)

[6] F. Bevilacqua, B. Zamborlin, A. Sypniewski, N. Schnell, F. Guédy, N. Rasamimanana, “Continuous realtime gesture following and recognition”, In *Embodied Communication and Human-Computer Interaction*, Springer Verlag, Berlin Heidelberg, vol. 5934 of Lecture Notes in Computer Science, pp. 73–84, (2010)

[7] F. Bevilacqua, R. Muller, N. Schnell, “MnM: a Max/MSP mapping toolbox”, In *Proc. of the International Conference on New Interfaces for Musical Expression*, Vancouver, BC, (2005)

[8] S. Manitsaris, “*Computer vision for the gesture recognition: gesture analysis and stochastic modelling in music interaction*”, Phd Thesis, University of Macedonia-Greece, (2010)

[9] D. Arfib, J. Couturier, L. Kessous, V. Verfaille: “Strategies of mapping between gesture data and synthesis model parameters using perceptual spaces” *Organized Sound* 7(2), 127–144 (2002)

[10] D. Levitin, S. McAdams, R. Adams: “Control parameters for musical instruments: a foundation for new mappings of gesture to sound”. *Organised Sound* 7(2), 171–189 (2002)

[11] A. Momeni and D. Wessel, “Characterizing and Controlling Musical Material Intuitively with Geometric Models,” in *Proc. of the 2003 conference on New interfaces for Musical Expression (NIME 03)*, pp. 54–62, (2003)

[12] R. Bencina, “The Metasurface: Applying Natural Neighbor Interpolation to Two-to-Many Mappings,” in *Proc. of 2005 Conference on New Interfaces for Musical Expression (NIME 05),* pp. 101–104, (2005)

[13] F. Bevilacqua, N. Schnell, N. Rasamimanana, B. Zamborlin, F. Guedy, “Online Gesture Analysis and Control of Audio Processing”, *Musical Robots and Interactive Multimodal Systems,* Springer Tracts in Advanced Robotics, Volume 74, Springer Verlag, (2011)

[14] D. Van Nort, M. Wanderley, P. Depalle “On the Choice of Mappings Based On Geometric Properties”, Proc. of the *International Conference on New Interfaces for Musical Expression*, (2004)

[15] B. Caramiaux, F. Bevilacqua, N. Schnell. "Analysing Gesture and Sound Similarities with a HMM-based

Divergence Measure" *Sound and Music Computing* (SMC'10), Barcelona, Spain (2010)

[16] A.F. Bobick, A.D. Wilson: “A state-based approach to the representation and recognition of gesture” *IEEE Transactions on Pattern Analysis and Machine Intelligence* 19(12), 1325–1337 (1997)

[17] A. Kapur, G. Tzanetakis, and P. F. Driessen, “Audio-based gesture extraction on the esitar controller” *International Conference on Digital Audio Effects*, 2004.

[18] J. Williamson and R. Murray-Smith, “Sonification of probabilistic feedback through granular synthesis,” *IEEE Multimedia, vol. 12, no. 2, pp. 45–52*, 2005