Journal of Korea Multimedia Society Vol. 19, No. 5, May 2016(pp. 826-836)
http://dx.doi.org/10.9717/kmms.2016.19.5.826

Design of Music Learning Assistant Based on Audio Music and Music Score Recognition

Ahmad Wisnu Mulyadi[†], Carmadi Machbub[††], Ary S. Prihatmanto[†††], Bong-Kee Sin[††††]

ABSTRACT

Mastering a musical instrument for an unskilled beginning learner is not an easy task. It requires playing every note correctly and maintaining the tempo accurately. Any music comes in two forms, a music score and it rendition into an audio music. The proposed method of assisting beginning music players in both aspects employs two popular pattern recognition methods for audio-visual analysis; they are support vector machine (SVM) for music score recognition and hidden Markov model (HMM) for audio music performance tracking. With proper synchronization of the two results, the proposed music learning assistant system can give useful feedback to self-training beginners.

Key words: Hidden Markov Model, Support Vector Machine, Chroma Feature, Histogram of Oriented Gradients

1. INTRODUCTION

A journey of a thousand mile begins with a single first step. Likewise, the journey of mastering a musical instrument for an unskilled beginning learner starts with repeated practicing of new music scores. They have to struggle with playing every note accurately and keeping the tempo correctly. At times they face some difficulties when they have to practice them without a teacher. At this point, a music learning assistant system with music score-tracking capability will come to the rescue. A music learning assistant is a system designed to help beginning learners practice music using realtime music score reading and feature tracking. An additional function required for the system is the alignment of audio-to-score that involves synchronizing audio performance produced by the learner and the musical symbols in the score [1,2,3,4].

Music score-following problem has been studied by a number of researchers. It was first introduced by Dannenberg who used classical approximate string matching and heuristic techniques [2,4]. Since then, researches using stochastic approaches have begun to appear. Raphael was a pioneer in applying HMM for music score following [2,3]. Later on, researchers began to use HMM for pitch class profiles or chroma features and reported some interesting results [5,6].

In their music score recognition study, Rabelo et al. [7] reported a research effort on optical music score recognition which consists of image preprocessing, music symbol recognition, musical no-

※ Corresponding Author : Bong-Kee Sin, Address: (608-737) 45 Yongsro-ro, Nam-Gu, Busan, TEL : +82-51-629-6256, FAX : +82-51-629-6230, E-mail : bkshin@pknu.ac.kr
Receipt date : Feb. 24, 2016, Approval date : Mar. 21, 2016
[†] Dept. of IT Convergence and Applications Engineering, Pukyong National University, Rep. of Korea
(E-mail : wisnumulyadi@gmail.com)
[††] School of Electrical Engineering and Informatics, Bandung Institute of Technology, Indonesia
(E-mail : carmadi@lskk.ee.itb.ac.id)
[†††] Center of ICT Research, Bandung Institute of Technology, Indonesia
(E-mail : asetijadi@lskk.ee.itb.ac.id)
[††††] Dept. of IT Convergence and Applications Engineering, Pukyong National University, Rep. of Korea
※ Acknowledgement: This work was supported by the Research Grant of Pukyong National University (2015)

tation reconstruction and final representation construction as shown in Fig. 1. In music symbol recognition there are various methods published to date which involve Neural Network (NN), Nearest Neighbour (kNN), Hidden Markov Model (HMM) and Support Vector Machine (SVM). Among them, SVM has been the most popular method exhibiting the best performance in recognition of both handwritten and synthetic music scores [7].

Motivated by those previous research efforts, this paper presents a design of music learning assistant based on audio-visual analysis. Here, two popular pattern recognition methods are employed, SVM for visual analysis and HMM for audio decoding.

In the rest of the paper, Section 2 explains the design of the proposed music learning assistant for audio-visual analysis. Then, Section 3 will cover feature extraction for visual pattern recognition method to be described in Section 4. The next section discusses the experimental results followed by the conclusion in Section 6.

2. DESIGN OF MUSIC LEARNING ASSISTANT

The design of the proposed music learning assistant is illustrated in Fig. 2. There are two type of inputs, music score images and audio music

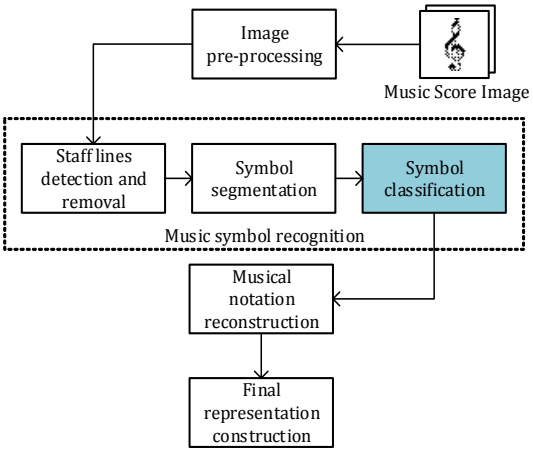

Fig. 1. Typical architecture of an OMR system [7].

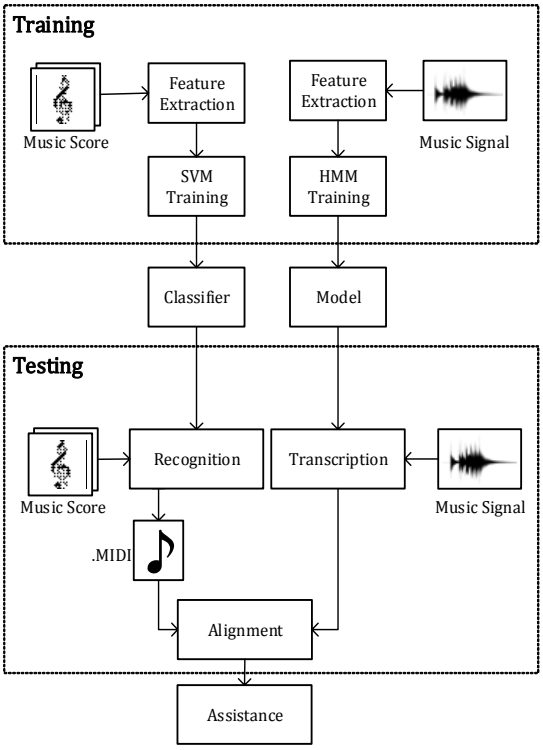

Fig. 2. Organization of the music learning assistant.

signals. Therefore, we have two tasks, music score recognition and music signal transcription.

For each task, we employ a popular pattern recognition method, SVM for music score recognition task and HMM for music signal transcription task. The success of the system lies in the optimality of the model parameters. In order to estimate the optimal set of model parameters, the system undergoes a training phase for both the SVM classifier and the HMM classifier.

3. FEATURES EXTRACTION

3.1 Histogram of Oriented Gradients Features

Histogram of Oriented Gradients or HOG was first introduced as a way of detecting pedestrians in video with a high performance [8]. Later on, the feature has also been applied to digit and character recognition tasks [9,10]. In our research, we employ HOG as a descriptor for music symbol recognition.

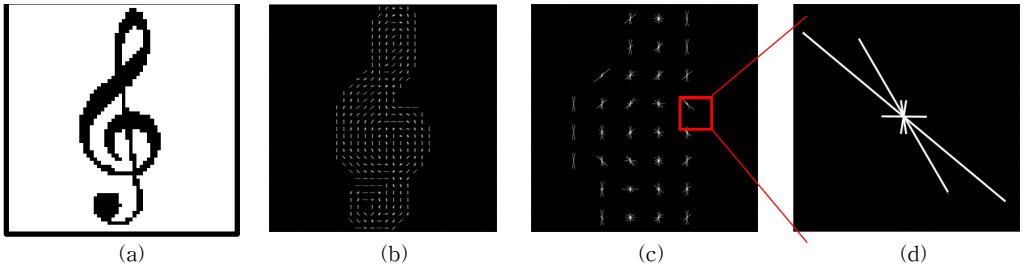

Fig. 3. (a) A music symbol image, (b) its HOG features using 2x2 cells, (c) 8x8 cells and (d) the magnification of a particular cell.

HOG is defined as the number occurrences of gradient orientation in parts of an image [9]. Given an input image divided into a set of small regions called cells, we compute a histogram of gradient directions or edge directions for each cell [8,9,10]. Fig. 3 shows HOG features with different cell sizes. Smaller cells tend to have more spatial information but will increase the number of dimensions, and vice versa. In this research, we choose the 2×2 cell since it provides more spatial information and turns in higher recognition rate. The HOG features are extracted using the method developed in [10].

3.2 Chroma Features

In Western music notation, there are 12 pitches in an octave which are denoted by an ordered set of symbols {C,C#,D,D#,E,F,F#,G,G#,A,A#,B}. They represent a cycle and repeat in octaves below and above [11,12] as shown in Fig. 4. The distance between two adjacent notes is defined as a halfstep. The distance we perceive as a halfstep is equal in all octaves. We capture this perceptual distance using the chroma features developed for music analysis [11,12].

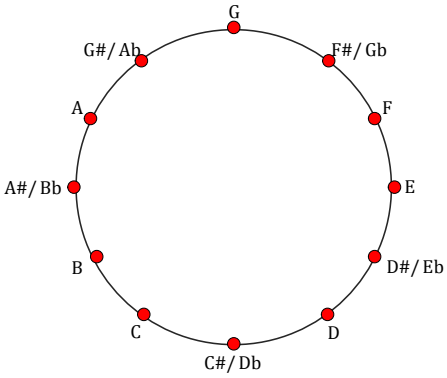

Fig. 4. Twelve semitones arranged in a cycle in Western music notation.

Chroma features are used to represent audio music signals where the entire spectrum for a short time frame is projected onto twelve bins corresponding to the twelve distinct semitones in the chromatic scale [12].

Given a music signal as shown in Fig. 5(a), we compute chroma features like Fig. 5(b) by summing up the log-frequency magnitude spectrum across octaves [11] as follows:

$$C_f(b) = \sum_{z=0}^{Z-1} \left| X_{lf}(b + z\beta) \right|,$$

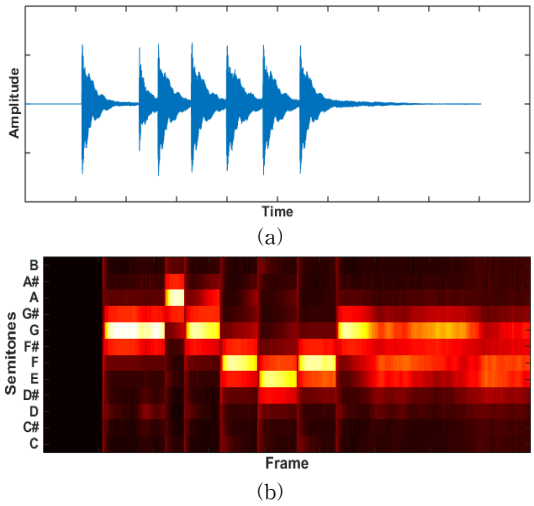

Fig. 5. (a) Input audio music signal and (b) the corresponding chromagram.

$$b = 0,1, \dots \beta - 1,\ z = 0,1, \dots, Z$$

where X_{lf} is the log-frequency spectrum, Z the number of octaves, b the pitch integer identifier representing the class index and β the number of bins per octave.

4. PATTERN RECOGNITION METHOD

4.1 Support Vector Machine (SVM)

Support vector machine or SVM is basically a binary classification method that constructs a hyper-plane in high order space separating samples of two classes[9,13].

SVM is designed using kernels which are typically based on linear, polynomial, radial basis function (RBF) or sigmoid kernels [9]. Given a feature vector $\boldsymbol{x}$, the kernel K is defined by the inner products of features $\phi(\boldsymbol{x_a}) = [\phi_1(\boldsymbol{x_a}), \phi_2(\boldsymbol{x_a}), \dots \phi_d(\boldsymbol{x_a})]^T$ with $\phi(\boldsymbol{x_b})$ given by

$$K(\boldsymbol{x}_a, \boldsymbol{x}_b) = \phi(\boldsymbol{x_a})^T \phi(\boldsymbol{x_b})$$

Once the kernel function is chosen, the classifier function $f(x)$ can be written as

$$f(\boldsymbol{x}) = sgn(\sum_{i=1}^{l} \alpha_i y_i (K(\boldsymbol{x}_i, \boldsymbol{x})) + b)$$

where $a = [\ a_1 \cdots a_l\ ]^T$ is the vector of l non-negative Lagrange multipliers to be determined, y_i the target value of support vectors and b the bias [13].

For music symbol recognition, we first specify the number of musical symbol classes such as accidental, bar, braces, clef, digits, dot, note and rest as illustrated in Table 1. Given a set of extracted HOG features from segmented music symbols from the traning set, we construct an SVM classifier using the toolbox provided in [10].

4.2 Hidden Markov Model (HMM)

4.2.1 Overview of HMM

HMM is a generalization of Markov chain. According to the first order Markov chain, a state at time t depends only on the single preceding state at time t-1 instead of the whole history of the process prior to the state[14,15]. In HMM, each state q_t of the Markov chain generates an observation o_t at time t as shown in Fig. 6.

The HMM is characterized by a number of parameters [14,15]:

- N, number of states, in the set of states $S = \{1,2, \cdots N\}$.
- M, number of distinct observation symbols in $V = \{1,2, \cdots M\}$.
- $A = \{a_{ij} : i, j \in S\}$, the state transition probability

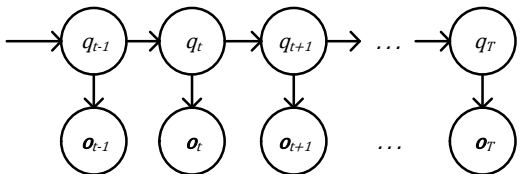

Fig. 6. HMM.

Table 1. Music symbol classes for SVMs

Classes Name	Music Symbols	Classes Name	Music Symbols
Accidental	♯ ♭ ♮	Dynamic	*mf mp p*
Bar	▌	Dot	●
Brace	⎨	Note	𝅝 𝅗𝅥 𝅘𝅥 𝅘𝅥𝅮 𝅘𝅥𝅯 𝅘𝅥𝅰
Clef	𝄞 𝄢	Rest	𝄻 𝄼 𝄽 𝄾
Digit	1234567890	Time signature	𝄴 𝄵

matrix, where $a_{ij} = P(q_{t+1} = j \mid q_t = I)$ is the probability of changing states from i to j at time t+1.

- $B = \{b_j(m)$: $i \in S$, $m \in V\}$, the observation probability matrix, where $b_j(m)=P(O_t = m \mid q_t = j)$ is the probability of observing symbol m in state j.
- π, the initial state probability distribution, where $\pi_i = P(q_1 = I)$ with $i \in S$, denotes the probability that a Markov chain starts in state i.

The model is often denoted simply by a triple $\lambda=(A, B, \pi)$, consisting of probabilistic parameters.

4.2.2 Continuous observation density HMM

In the previous section we defined V as a set of discrete observation symbols from each state. Since, however, we are dealing with audio music captured as a continuous-valued signal, we employ continuous observation densities in the HMM.

A continuous density HMM is often characterized by a parametric family of density functions or a mixture of certain density functions in each state [14,15]. Assuming the use of Gaussian mixtures, the emission density of state j is defined as:

$$b_j(\boldsymbol{O}) = \sum_{k=1}^{K} w_{jk} \mathcal{N}\left(\boldsymbol{O}; \boldsymbol{\mu}_{jk}, \Sigma_{jk}\right), \quad j = 1,2, \dots N$$

where K is the number of mixtures and w_{jk} the mixing coefficient for the kth Gaussian in state j subject to the following stochastic constraints :

$$\sum_{k=1}^{K} w_{jk} = 1, \quad j = 1,2, \dots N$$

where N denotes the Gaussian density function with mean $\mu_{jk} \in R^d$ and covariance matrix $\Sigma_{jk} \in R^{dxd}$ for the kth mixture.

4.2.3 Model Topology

In music transcription tasks, the pitches to be modeled are {C,C#,D,D#,E,F,F#,G,G#,A,A#,B} according to Western music notation. By using an HMM for the music transcription task, we try to describe the pitch trajectory given a sequence of chroma features of music signal. A music can be viewed as an ordered sequence of timed notes, but basically any note can be followed by any of the twelve notes. Based on this, we design a continuous HMM with 13 states, 12 states for the 12 pitches and one state for the silence. In addition, we choose a fully connected ergodic topology for our transcription HMM shown in Fig. 7. Since the input frame rate is generally much faster than the corresponding musical notes progression, the self loop transition parameters a_{ii}, $i = 1,2, \cdots, N$ dominate the distribution in each row of the transition matrix.

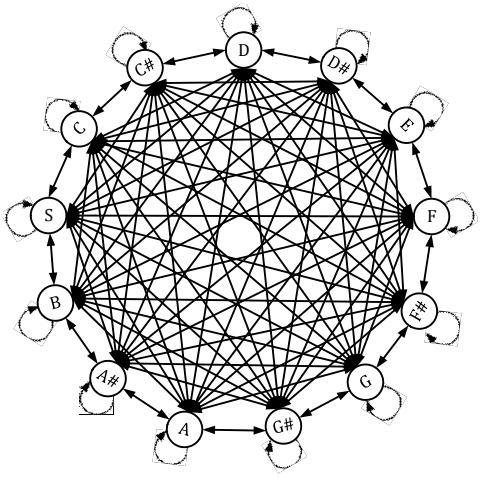

Fig. 7. HMM topology for music pitch transcription.

4.2.4 Viterbi Decoding

As in the case of many practical applications, HMMs are trained by the Baum-Welch algorithm. But most model decoding methods employ Viterbi algorithm for a detailed analysis into model behaviour. It is also adopted here for pitch tracking.

Viterbi algorithm is the most popular technique for finding the optimal path in an HMM given an observation sequence. The Viterbi algorithm is used to find the single best state sequence that most likely have produced the observations [14,15].

Using the trained HMM parameter λ we aim to find the single most likely pitch sequence $q=(q_1, q_2, \cdots, q_\tau$ given chroma features as an observation

sequence $\boldsymbol{O} = (\boldsymbol{o}_1, \boldsymbol{o}_2, \cdots, \boldsymbol{o}_t)$. In this case we define

$$\delta_t(i) = \max_{q1,q2\ldots,qt-1} P(q_1\, q_2 \ldots, q_t = i,\ \boldsymbol{o_1}\, \boldsymbol{o}_2 \ldots\, \boldsymbol{o}_t\, |\lambda)$$

as the probability along a single best partial path that ends in state i at time t with the partial output sequence [6,14,15]. By using induction, we can calculate the probability at time $t+1$ as

$$\delta_{t+1}(i) = [\max\ \delta_t(i)\, a_{ij}]\, b_j(\boldsymbol{o}_{t+1})$$

In online decoding tasks based on Viterbi algorithm, the information about the future notes is not available. This problem can be handled by an approximate decoding based on $\boldsymbol{L}$ local frames of the chroma features stored in the observation buffer as proposed in [6]. A correspondingly modified Viterbi algorithm will decode the short term sequence as shown in Fig. 8 [6]. To save computation, each buffer decoding calculation $\delta_t(i)$ will reuse previous buffer calculations $\delta_{t-1}(i)$ except for the initial for $\delta_1(i)$. Regarding the decoding result of each observation in the buffer we employ a voting system so that each buffer has only one decoding result which becomes the pitch estimate.

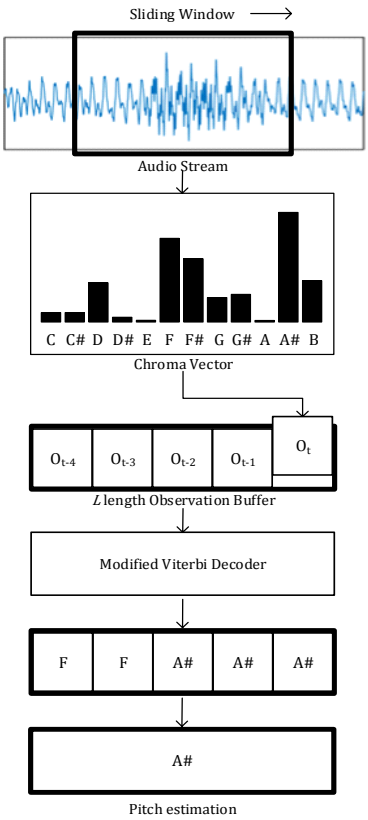

Fig. 8. Viterbi decoding workflow [6].

5. EXPERIMENTS

Since there are two subtasks in the music learning assistant, two experimental results will be described separately. First music score recognition

Table 2. Music symbols training data

Classes	Sub Classes	Number of Music Symbol
Accidental	Flat	61
	Natural	15
	Sharp	52
Bar	-	38
Brace	-	52
Clef	Clef F	36
	Clef G	58
Digit	Digit 0	16
	Digit 1	34
	Digit 2	23
	Digit 3	15
	Digit 4	26
	Digit 5	14
	Digit 6	16
	Digit 7	12
	Digit 8	42
	Digit 9	9
Dot	-	71
Dynamic	Dynamic MF	3
	Dynamic MP	2
	Dynamic P	1
Note	Crotchet	109
	Crotchet Reverse	284
	Minim	28
	Minim Reverse	72
	Quaver	38
	Quaver Reverse	23
	Semibreve	66
Rest	Crotchet	26
	Minim	52
	Quaver	17
	Semiquaver	3
Time Signature	Time signature 34	7
	Time signature 44	10

using SVM classifier and then pitch transcription of the music using an HMM.

5.1 Music score recognition

In order to train the music symbol classifier, we prepared a training set as described in Table 2. Music symbols in the training data are obtained from several beginner piano music scores which are already processed through line staff removal, gap stitching and segmentation. Then we extract HOG features and use them as the training data for training the SVM classifier.

The trained SVM classifier is used to recognize music symbols in test music scores. The result of music score symbol recognition is shown in Table 3. The overall average performance of the classifier is 96.02% correct recognition for beginner's piano music scores.

There are some misclassification cases as shown in Table 4. Most of the errors are attributed to missegmentations. Current system handles only single head notes so it fails on multiple head notes. When this problem is resolved in the future work, we can expect a higher performance in music score recognition.

Once all symbols are recognized, we can create a reconstruction matrix for the music score, and then finally convert it into a MIDI file using a method provided in [16]. Fig. 9 shows a sample result.

Table 3. Music score recognition result

Music Score Name	Number of Symbol	Number of Hits	Accuracy (%)
London Brige is Falling Down	64	64	100.00
Twinkle Twinkle Litle Stars	90	84	93.33
Peter Peter Piano	76	76	100.00
Au Clair De La Lune	99	98	98.99
Mary Had a Little Lamb	90	79	87.78

Table 4. Misclassification cases

Music Symbols				
Classification Result	Digit 1	Note Crocthet	Digit 1	Note Minim
Ground truth	Time Signature 44	Note Minim	Note Crotchet	Note Crotchet

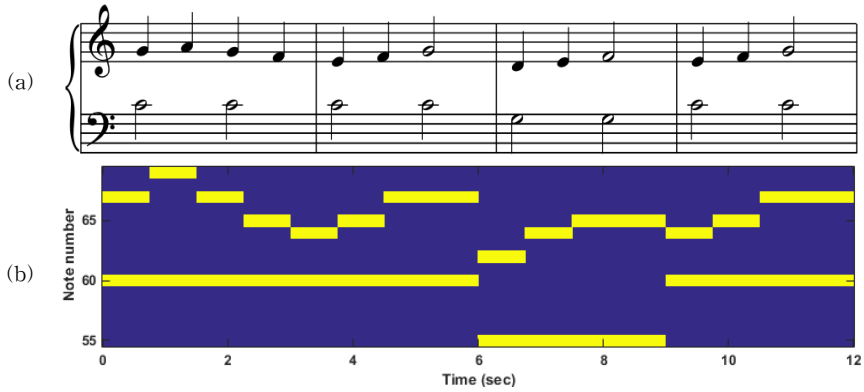

Fig. 9. (a) A music score and (b) the corresponding MIDI rendering.

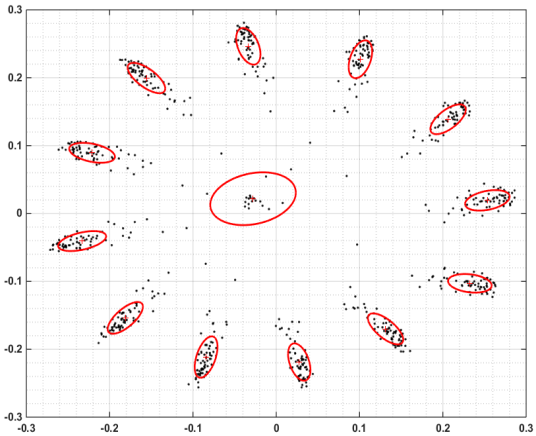

Fig. 10. HMM training result.

5.2 Music transcription

For music transcription task, we have built a continuous HMM trained by the Baum-Welch algorithm. Then we have applied the Viterbi algorithm to decode out the most likely sequence of pitches given chroma features. Each state of the HMM models the local patterns of chroma features, using a single Gaussian. When projected to the feature space with reduced dimensionality, the thirteen Gaussians make an interesting pattern as shown in Fig. 10.

Each dot represents a chroma feature vector. And each cluster of dots represents an HMM state corresponding to a particular pitch. Given a feature vector in context, the HMM returns the most likely pitch via the Gaussian densities represented by the ellipses. The Gaussian ellipse in the center represents silence whereas all the other ellipses around it represent the corresponding pitches. The visualization show that the HMM managed to model the pitches intuitively well.

The trained HMM is applied to label input audio music using Viterbi algorithm. The parameters for online Viterbi decoding include: the frame size is 2048 and L=5 is the length of the observation buffer. Fig. 11 shows a sample decoding result given a music scale performance. It shows that the HMM with 13 states accurately segment and label the pitches including silence. Finally, given a beginner's piano music performance on the familiar song "London Brigde is falling down", the HMM also turned in a very accurate result are shown in Fig. 12, which is deemed sufficient for musical learning assistance.

There is a remaining problem of synchronization, that is defined as a task of aligning the audio music to music score. In audio music we distinguish three events for each note, namely *attack*, *sustain* and *rest* events [3]. By detecting these events given an audio music, the system can estimate the starting and ending time of some notes. Based on this, the proposed system can estimate

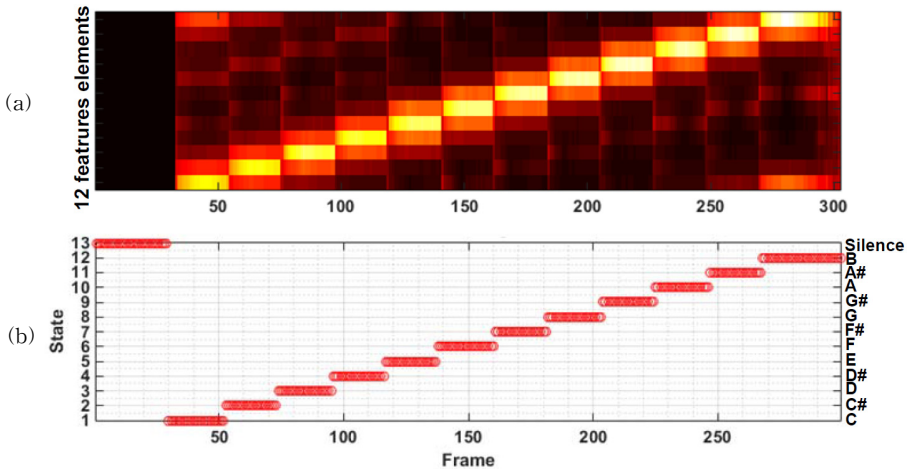

Fig. 11. (a) A chromagram of music scale performance and (b) the corresponding Viterbi decoding result.

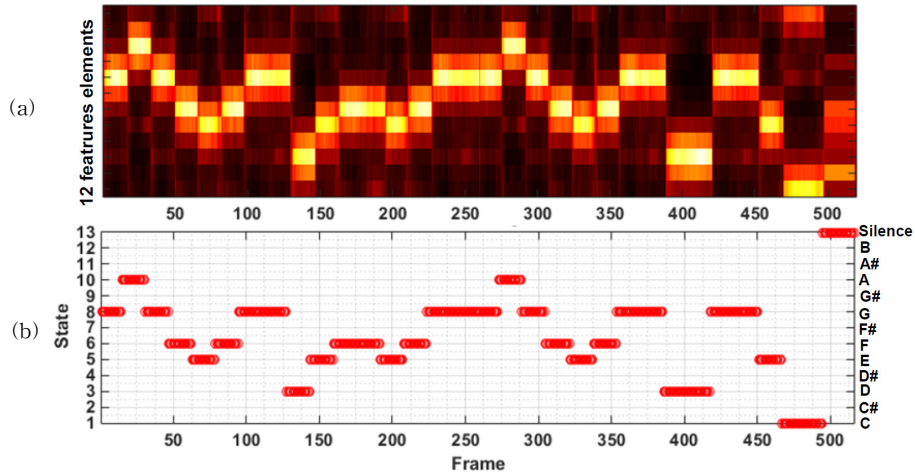

Fig. 12. (a) Music performance chromagram of "London Bridge is Falling Down" and (b) corresponding Viterbi decoding result.

the tempo of the notes being played and compare it with the music notation duration in music score as the ground truth. If all the results from the SVM classifier of music score, the HMM decoder, and the tempo estimation is properly synchronized, then we can give a useful feedback about the music performance, whether the learner has played pitches correctly and keeping the tempo accurately or not with respect to its music score [17,18].

6. CONCLUSION

This research presents a design of music learning assistant which involves two popular pattern recognition methods. The SVM classifies music symbols in a music score and the HMM tracks the sequence of pitches given a audio music. Based on the current implementation with limited training sets, the SVM shows an average performance about 96.02% correct symbol recognition. Whereas the HMM decoding found pitches almost perfectly with a few errors found only in the note boundaries by just one or two frames. Finally, if the result of both models are integrated with a proper synchronization based on attack times and beats, the proposed system will be able to give a useful feedback according to learner's performance.

REFERENCE

[1] P. Cano, A. Loscos, and J. Bonada, "Score-Performance Matching Using HMMs," *Proceeding of the International Computer Music Association*, pp. 441-444, 1995.

[2] N. Orio, "An Automatic Accompanist Based on Hidden Markov Models," *Proceeding of Advances in Artificial Intelligence*, pp. 64-69, 2001.

[3] A. Cont, D. Schwarz, and N. Schnell, "Training IRCAM's Score Follower," *Proceeding of International Conference on Acoustics, Speech, and Signal Processing*, pp. 253-256, 2005.

[4] B. Pardo and W. Birmingham, "Modeling Form for On-line Following of Musical Performances," *Proceedings of the 20th National Conference on Artificial Intelligence* Vol. 2, pp. 1018-1023, 2005.

[5] A. Sheh and D. Ellis, "Chord Segmentation and Recognition using EM-Trained Hidden Markov Models," *Proceeding of 4th International Symposium on Music Information Retrieval*, pp. 185-191, 2003.

[6] T. Cho and J.P. Bello, "Real-Time Implementation of HMM-Based Chord Estimation in Musical Audio," *Proceeding of the Interna-*

tional Computer Music Conference, pp. 16-21, 2009.

[7] A. Rabelo, I. Fujinaga, F. Paskiewicz, A.R.S. Marcal, C. Guedes, and J.S. Cardoso, "Optical Music Recognition : State-of-the-art and Open Issues," *International Journal of Multimedia Information Retrieval*, Vol. 1, No. 3, pp. 173-190, 2012.

[8] N. Dalal and B. Triggs. "Histograms of Oriented Gradients for Human Detection," *IEEE Computer Vision and Pattern Recognition*, Vol. 1, pp. 886-893, 2005.

[9] R. Ebrahumzadeh and M. Jampour, "Efficient Handwritten Digit Recognition Based on Histogram," *International Journal of Computer Applications,* Vol. 104, No. 9, pp. 10-14, 2014.

[10] ___, "Digit Classification Using HOG Features," in *MathWorks^{TM} Documentation*. http://www.mathworks. com/help/vision/examples/digit-classification-using-hog-features.html. (accessed on June 2, 2016)

[11] J.P. Bello, "Chroma and Tonality," *New York University's Music Information Retrieval Course*. http://www.nyu.edu/classes/bello/MIR.html (accessed June 2, 2016).

[12] C. Tralie, "Musical Pitches and Chroma Features" *in Duke University's Digital Signal Processing Course*, http://www.ctralie.com/Teaching/ECE381_DataExpeditions_Lab1 (accessed on June 2, 2016).

[13] D, Boswell. "Introduction to Support Vector Machines", 2002. http://dustwell.com/PastWork/IntroToSVM.pdf. (accessed on June 2, 2016).

[14] L.R. Rabiner, "A Tutorial on Hidden Markov Modles and Selected Applications in Speech Recognition," *Proceeding of the IEEE*, Vol. 77, pp. 257-286, 1989.

[15] M, Nilsson, *First Order Hidden Markov Model : Theory and Implementation Issues*, Blekinge Institute of Technology Research Report, pp. 9-54, 2005.

[16] K. Schutte, "MATLAB and MIDI". http://kenschutte.com/midi (accessed on June 2, 2016).

[17] A.W. Mulyadi and B.-K. Sin. "Chroma Based Music Score-Following Using Hidden Markov Model," *Proceeding of the Fall Conference Paper on Korean Multimedia Society*, 2015.

[18] T. Hur, H. Cho, G. Nam, J. Lee, S. Lee, S. Park, et al., "A Study on the Implementation of the System of Content-based Retrieval of Music Data," *Journal of Korea Multimedia Society*, vol. 12, iss. 11, pp. 1581-1592, 2009.

Ahmad Wisnu Mulyadi

He received his Bachelor degree of Computer Science Education at Indonesia University of Education in 2010. Motivated to study further, he decided to pursue his Master degree in Digital Media and Game Technology at Institut Teknologi Bandung in 2014 and joined Master Double Degree Program between Institut Teknologi Bandung and Pukyong National University in 2015. He attended Intelligent Media Lab under supervision of Professor Bong-Kee Sin. His current research interests are in machine learning, educational media and games development.

Carmadi Machbub

In 1991 he got a Doctorat degree in Engineering Sciences majoring in Control Engineering and Industrial Informatics from Ecole Centrale de Nantes, France. As an academic faculty of the Institut Teknologi Bandung, in the period between 1998 until 2011 he was assigned as Head of Electrical Engineering Department, Senior Vice Rector for Resource Management, and Vice Rector for Academic and Student Affairs. He is now Professor and Head of Control and Computer Systems Research Division, School of Electrical Engineering and Informatics, Institut Teknologi Bandung. His current research interests are in pattern recognition, optimization, perception and control.

Ary Setijadi Prihatmanto

He graduated with B.E. and M.S. in Electrical Engineering at Institut Teknologi Bandung in 1995 and 1998, and received his PhD in Applied Informatics from Johannes Kepler University of Linz, Austria in 2006. He is an associate professor & lecturer of School of Electrical Engineering & Informatics, Institut Teknologi Bandung since 1997. He is also the chair of IEEE Computer Society Chapter Indonesia Section and the president of Indonesia Digital Media Forum since 2009. And currently, he has also served as the Head of Information & Communication Technology Research Center at Institut Teknologi Bandung and involved in various governmental, non- governmental and international research projects in Indonesia. His main interests are Human-Content Interaction, Computer Graphics & Mixed-Reality Application, Machine Learning & Intelligent System, Intelligent Robotics & Cyber-Physical System.

Bong-Kee Sin

1985, Bachelor degree from the Department of Mineral and Petroleum Engineering, Seoul National University.
1987, Master degree from the department of Computer Science, Korea Advanced Institute of Science and Technology.
1995, PhD from the Department of Computer Science, KAIST.
1987~1999, Seanior Researcher, SW Research Laboratories, Korea Telecom.
1999~present, Professor in the Department of IT Convergence and Applications Engineering, Pukyong National Universwity.
Research interest : pattern recognition, machine learning, computer vision and artificial intelligence.