Educational Violin Transcription by Fusing Multimedia Streams

Ye Wang Bingjun Zhang Olaf Schleusing
Department of Computer Science, National University of Singapore
E-mail:{wangye,bingjun,olaf}@comp.nus.edu.sg

ABSTRACT

Computer-assisted violin tutoring requires accurate violin transcription. For pitched non-percussive (PNP) sounds such as from the violin, note segmentation is a much more difficult task than pitch detection. This issue is accentuated when the audio is recorded during an instrument practice session at home which is acoustically inferior to a professional recording studio. This paper presents a new approach to the problem by using the correlation between different media streams for e-learning applications. We design a capture mechanism to record one audio and two video streams simultaneously, and exploit the relationships among them for enhanced transcription. State-of-the-art audio methods for note segmentation and pitch estimation are implemented as the audio-only baseline. Two web-cameras are employed to track the right hand (bowing) and the left hand's four fingers (fingering) on the fingerboard, respectively. The audio and visual information is then fused in the feature space. Our new approach is evaluated with an audio-visual violin music database containing 16 complete music pieces of different styles with 2157 notes in total. Experimental results show that our multimodal approach achieves a 10% increase in true positives, and a 8% reduction in false positives of overall transcription performance in comparison with the audio-only baseline.

Categories and Subject Descriptors

H.5.5 [**Sound and Music Computing**]: Signal analysis, synthesis, and processing, Systems

General Terms

Algorithms, Design, Experimentation, Human Factors

Keywords

Computer-Assisted Tutoring, Music Transcription, Note Segmentation, Onset Detection, Detection Function, Audio-Visual Fusion

EMME'07, September 28, 2007, Augsburg, Bavaria, Germany.

1. INTRODUCTION

The pedagogical foundation of our work is David Perkin's *Theory One*. *Theory One* says that "people learn much of what they have a reasonable opportunity and motivation to learn" [2]. *Theory One* summarizes four essential aspects of effective learning:

- Clear information;
- Thoughtful practice;
- Informative feedback;
- Strong intrinsic or extrinsic motivation.

Inspired by *Theory One*, we aim to create an interactive Digital Violin Tutor (iDVT) whose initial version was reported in [1]. It represents a genuinely new learning experience based on combinations of physical and virtual resources and interactivity (blended learning). An important attribute of iDVT is that it makes violin practice both fun and effective, in step with the available technologies and learning stimuli.

This pedagogical tool adds value to learning experiences in the targeted area of learning: playing violin. It provides clear information, motivation and fosters thoughtful practice in contrast to repetitive and uninspired rehearsal, thereby increasing the efficiency while reducing the cost of learning.

iDVT fosters self-paced learning whereby students can learn at the rate they prefer. It is convenient for students to access any time, any place, and furthermore, it reduces travel time and travel costs for students and parents. Furthermore does this system create a constructive learning environment where learners may network online, work together and support each other. Even in the case when a teacher or a parent is not available, iDVT will act like an intelligent learning companion to provide instant and informative feedback.

All the above benefits rely on an accurate violin transcription, which plays a critical role during the development of the iDVT system. In this paper we focus on enhancing violin transcription.

In the analysis and understanding of music, the 'Note' is a basic event. Finding the pitch of notes of pitched non-percussive (PNP) sounds such as from a violin is relatively easy, but identifying the precise beginning and end of specific notes and correlating them with the pitch (note segmentation) automatically is a challenging and critical task for computer aided tutoring at home [1]. Existing methods exhibit poor results in note segmentation or onset detection for PNP sounds [3]. As pointed out in [4], a promising development for onset detection schemes lies in the combination of cues

from different audio detection functions. In this paper, we enhance this idea by fusing detection functions from different media streams with the hope of significantly improving onset detection and thereby transcription performance. Our work is mainly motivated by the following observations and hypotheses:

1) The bow stroke reversals (right hand) and vertical movements are associated with note onsets;
2) The trajectories of fingers (left hand) are associated with note onsets. We believe that these are the most important visual cues which can assist in the note segmentation task.

We first derive audio and video detection functions, which later are fused in the feature space. The fused detection function is then sent to a peak picking module, energy based onset/offset separation function, followed by pitch estimation (Figure 1). This simple architecture is adequate for the proof of concept and for determining whether and to which extent the visual cues improve the onset detection performance. In addition, the audio-visual approach can also provide useful visual information to the player as learning feedback, such as playing gestures, and fingering and bowing trajectories.

The rest of the paper is organized as follows. In the next section, we review related work. Our conceptual framework and methodology are outlined in Section 3. Sections 4 and 5 detail the audio and video processing components. We then discuss our system integration process in Section 6. In the last few sections, we analyze system performance and conclude with comments on current and future work.

2. RELATED WORK

This section surveys the works which have inspired our current research. There are few published works on music transcription fusing audio-visual features and drum transcription in [5] is the first system dealing with percussive instruments using audio and video inputs. To our knowledge, our system is the first audio-visual transcription system with string instruments. Cognitive brain research has shown that the temporal structure of violin music depends on a number of combinatorial mechanisms of bowing and fingering [6]. Such research has stimulated our initial attempt to augment existing audio-only transcription with visual information. The hypothesis is that the different modalities (e.g., audio and video) are generated from the same information source simultaneously and they ought to be correlated. We assume the complementary information from different modalities is helpful in improving violin transcription performance. The design philosophy of the proposed method is in many ways comparable to audio-visual speech recognition [7, 8, 9]. The main difference is that their work used facial features to improve speech recognition, while our work exploits the motion features of bowing and fingering to assist violin music transcription. Furthermore, multimedia fusion approach has been applied in emotion recognition [10].

3. SYSTEM DESCRIPTION

We have attempted an initial design of a capture system to simultaneously record audio and video streams. Our system consists of one microphone and two web-cameras. To simplify the tracking of bowing and fingering trajectories, we have employed color markers on the fingernails and the end of the bow (see Figure 3). Distance and position of the web-cameras were chosen to obtain the largest finger and bow excursions.

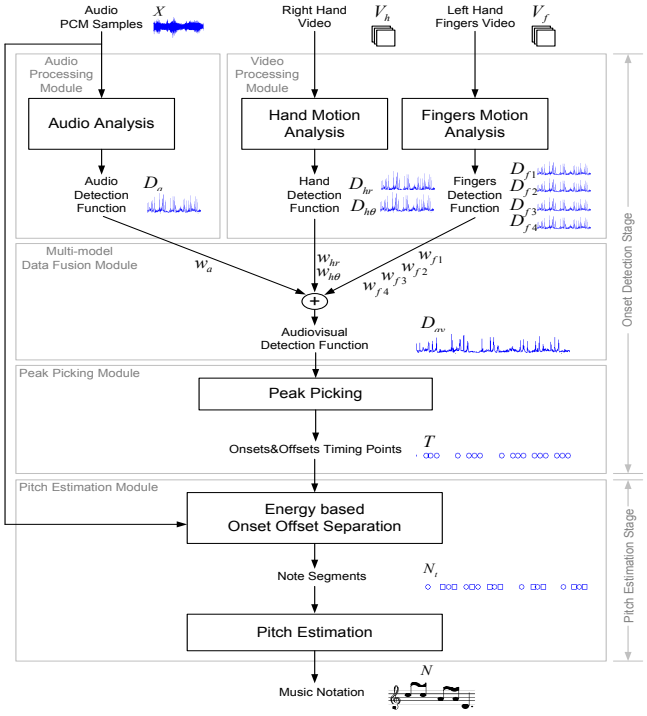


Figure 1: Audio-visual violin transcription system.

The system architecture is illustrated in Figure 1. The first step is a simultaneous processing of audio, fingering and bowing motion in order to derive the individual detection functions which are fused in the feature space. The fused audio-visual detection function is then sent to peak picking and onset/offset distinction module, followed by pitch estimation.

4. AUDIO PROCESSING

This section describes the basic audio transcription method used in our system. This task is subdivided into two parts; onset detection and pitch estimation. We have implemented two types of best existing transcription methods, general and instrument-specific, to be our audio-only baseline. That comprises three different methods for onset detection and three different methods for pitch estimation.

4.1 Onset Detection

The detection of onsets has been addressed in many publications before. Nevertheless this essential problem is far from being solved. Note onsets technically represent a transient segment of the audio signal, or more specifically they mark the instant in time, when the signal starts to evolve from a steady state to another steady state, i.e. from one note to the next. Since audio signals are oscillatory in their nature it is not possible to just use the first order derivative of the time domain signal to detect higher-level changes like onsets. Most approaches perform an optional pre-processing step, employ an intermediate signal, which is at a significantly lower sampling rate than the audio signal itself and reduce the signal to a more favorable representation. This

representation, which is called a detection function or in the style of video processing a novelty function [11], is then used by a peak picking method to find local maxima i.e. onsets. A comprehensive evaluation of different detection functions and peak picking methods for different types of audio signals was published recently in [4]. The authors explained and compared different methods for finding onsets in musical signals and provided information on which methods they found work best for which signal class and which application. Based on these comparisons Collins set up a more comprehensive evaluation of onset detection functions, which he described in [3]. As a conclusion of his experiments with PNP sound he proposed a new method in [12], which was developed under the assumption that the perception of stable pitch cues could be linked to the segmentation of notes. We investigate in this paper three different methods, namely an equal-loudness contour based method [3], a pitch-based method [12] and an instrument specific version of the inverse correlation method described in [13]. The optimal set of parameters for each method is exploited by running a comprehensive test on our musical database.

Two different methods are employed for the task of picking the peaks from the detection functions. The first method low pass filters the detection function and then finds local maxima in this function based on a fixed threshold. The value of this threshold is found via comprehensive tests against our music database and is specific to every detection function. The second method uses a base threshold and a median filter to calculate an adaptive threshold, see [4] for details. We refrain from implementing some sophisticated post-processing, like machine learning methods, since the focus of this work is on the proof of concept of multimedia transcription.

4.2 Pitch Estimation

For the pitch estimation of the segmented notes we evaluate three different approaches. We use re-implementations of Klapuri's generic pitch estimation methods based on auditory models described in [14] and [15] and compare it with the violin specific pitch estimator described in [16]. All methods perform very well on violin music.

5. VIDEO PROCESSING

This section describes the video processing module, illustrated in Figure 2. In the following subsections, we discuss the right hand motion (bowing) and left hand fingers motion (fingering) capturing and analysis to explore the relationship between trajectories of bowing and fingering and onset events of violin music. For violin playing, the right hand motion and left fingers motion are directly related with bowing and fingering, which are two of the most important playing techniques to produce notes (onsets, offsets and pitches) [17]. The discussions for both hand and fingers motion share the same procedure listed below:

- Camera view calibration for motion capturing, in which part the camera view and position to capture proper motion are discussed;
- Marker(s) tracking and polar coordinate system setup, where we describe the marker tracking algorithm to generate hand trajectory and finger trajectories of the violin player in the original Cartesian coordinate system of the video frame and illustrate how we set up a new polar coordinate system for both hand and finger parts to best decompose the motion trajectories;
- Motion decomposition and detection function construction. In this part, we formulate the motion decomposition procedure to generate new hand and finger trajectories, based on which we construct detection functions to aid onset detection of transcription by fusing with the audio detection function discussed in Section 4.

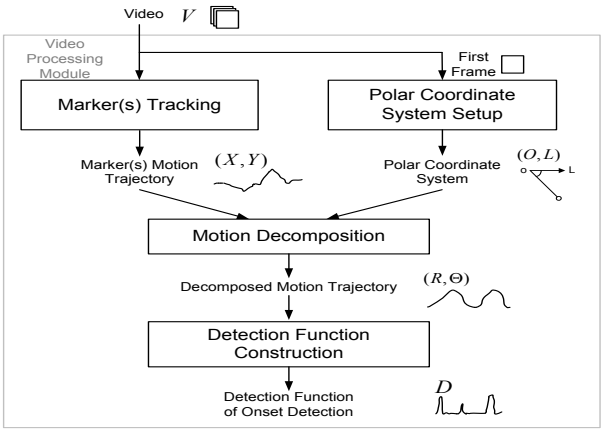


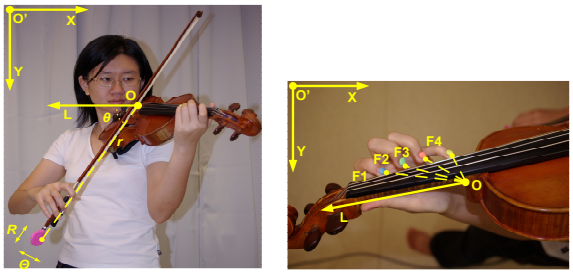
Figure 2: Video processing module.

5.1 Camera View Calibration

In the motion capturing step, we use two regular web-cameras to capture the right hand and the left hand fingers motion simultaneously. The capturing setting is 30 frames per second (fps) for the web-cameras with ordinary indoor lighting condition, which can be readily achieved on the sort of computer systems available to home users.

After intensive study of violin playing technique [17], the side view of violin player, shown in Figure 3(a), is chosen to capture the right hand motion. From this view, two dimensions of hand motion can be best captured: the movement parallel to the bow direction and the movement orthogonal to the bow. The hand motions in the two dimensions are both critical to violin playing, because hand movement in either of the two dimensions will produce new notes during violin playing. From a side view, the capturing direction is orthogonal to the two dimensional motion, which allows us to best capture the two dimensional hand motion. In order to best decompose hand motion in the two dimensions, the hand polar system, shown in Figure 3(a), is set up as the analysis basis for hand motion. The intersection point of bow and fingerboard of violin is set as its origin O and the direction to the left of origin is set as its polar axis L.

Based on the study of violin fingering, four fingers of the left hand move to press and release the strings on the fingerboard to produce notes during violin playing. In addition, the four fingers move independently towards and away from the wrist of the left hand of the violin player because of the physical constraints of human fingers. Also, the wrist is usually overlapped with the origin of the finger polar coordinate system, the lower crossing point of the fingerboard and upper bout (body) of violin, as can be seen in Figure 3(b). Therefore, a birds eye view is chosen to capture the fingers motion of the violin player. To best decompose fingers motion, the lower crossing point of the fingerboard and upper



(a) (b)

Figure 3: Camera view and polar coordinate system. (a) The right hand. (b) The left hand fingers.

bout of the violin is set as the origin O, and the direction to the left of the origin is set as the polar axis L of the finger polar coordinate system.

As can be seen in Figure 3, we use color markers to aid the tracking of hand and fingers motion. In this phase of our research in music transcription, we focus on the discussion about hand and fingers motion analysis and audio-visual information fusion for the creation of audio-visual transcription system. Bare hand and bare finger trackings are also active research areas in the object tracking field of computer vision [18, 19, 20]. However, the exploration of object tracking algorithms for bare hand and bare fingers is out of the scope of this paper.

5.2 Right Hand Motion Analysis

5.2.1 *Right hand marker tracking*

The right hand marker with red color and round shape is placed at the lower tip of the bow (Figure 3(a)). Because the right hand of violin player firmly holds the bow and the relative position of the right hand to the bow does not change at all during playing. Therefore, we track the trajectory of the marker and use it as the hand trajectory.

To track the marker, we use the color information to derive the tracking algorithm (Algorithm 1). As the marker motion is captured by a regular web-camera in ordinary indoor lighting condition, the color of the marker does not always remain the same during violin playing. It is important to employ a color information update mechanism in step 3 to maintain the robustness of the marker tracking algorithm.

Before marker tracking, in the first frame of a video sequence, we manually pick up the position (x_{h0}, y_{h0}) of the intersection point of the bow and fingerboard as the origin of the hand polar system. From the observation of violin playing, the slight movement of the intersection point of bow and fingerboard can be ignored. As a result, we assume the origin (x_{h0}, y_{h0}) does not move during playing, and fix it according to the first frame of the video sequence.

5.2.2 *Right hand motion decomposition*

After tracking, the trajectory (X_h, Y_h) of the right hand is obtained, which is based on the original Cartesian coordinate system of the video frame. We need to decompose this trajectory (X_h, Y_h) into the two useful directions (R_h, Θ_h) of the hand polar coordinate system as Equation (1).

Input: Right hand video sequence V_h with n frames
Output: Red marker positions in all frames (X_h, Y_h)

1 Set red color RGB component value range as RGB;
2 In the first frame I_1 of V_h, find all pixels whose RGB component values fall into RGB;
3 Calculate the gravity center (x_{h1}, y_{h1}) of pixels found in step 2 as the marker position in I_1, add it into marker position set (X_h, Y_h) and calculate the average RGB value of those pixels to update RGB;
4 **for** $i \leftarrow 2$ *to* n **do**
5 Search in frame I_i around the square region with length l around (x_{hi-1}, y_{hi-1}) to find all pixels whose RGB component values fall into RGB;
6 Do the same as step 3 to get marker position (x_{hi}, y_{hi}) and update RGB;
7 **end**

Algorithm 1: Color marker tracking algorithm, where the initial RGB value is set as the marker color in the first frame with a certain color range. Search square length l is set to 100 in pixels in the implementation, which is based on the observation of hand motion speed of violin playing and the frame rate (30 fps) of the web-camera.

$$\begin{bmatrix} r_{hi} \\ \theta_{hi} \end{bmatrix} = \begin{bmatrix} \sqrt{(x_{hi} - x_{h0})^2 + (y_{hi} - y_{h0})^2} \\ \arctan \left| \frac{y_{hi} - y_{h0}}{x_{hi} - x_{h0}} \right| \end{bmatrix} \quad (1)$$

where (x_{h0}, y_{h0}) is the position of the origin of the hand polar coordinate system, $(x_{hi}, y_{hi}) \in (X_h, Y_h)$ is the hand position in frame I, and $(r_{hi}, \theta_{hi}) \in (R_h, \Theta_h)$, $1 \leq i \leq n$, is the decomposed hand motion values in the hand polar coordinate system.

Radius vector R_h reflects the hand motion trajectory along the bow, and angle vector Θ_h reflects the hand motion trajectory orthogonal to the bow. In Figure 4, after hand motion decomposition, we can clearly see that the hand motion R_h and Θ_h trajectories are highly correlated with the human annotated onset timing points.

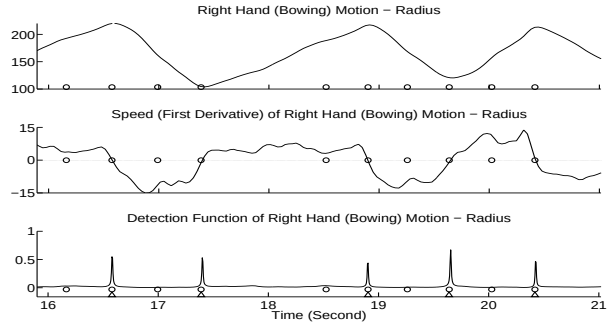
5.2.3 *Detection function of right hand motion*

First, we discuss the correlation between radius trajectory R_h and onset events.

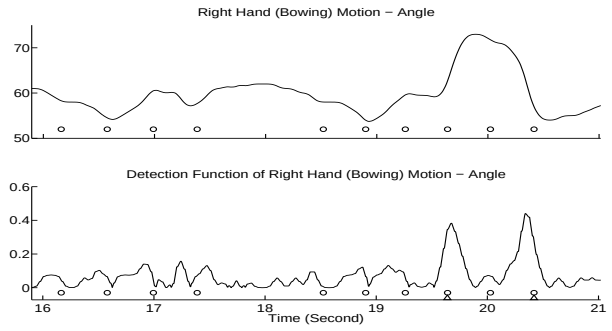
From Figure 4, we can see that at each extremum point of trajectory R_h, there is always a corresponding onset timing point annotated by human. Bowing in violin playing is a basic approach to produce separate notes. The changing of bowing direction r (bow reversal) produces new notes. In addition, when the bow is moving upwards (or downwards) along bow direction r, the player suddenly slows down and speeds up the bow to produce an onset. The two cases of note production by bowing can be reflected in the constructed detection function of hand radius motion.

From the analysis of the relationship between violin bowing and note onset, we conclude that when the motion of bowing suddenly slows down and speeds up, a new onset will be produced. In other words, when the speed of hand motion R is lower, we are more confident that the corresponding timing is an onset.

To model the above reasoning, we derive the first derivative R'_h of hand motion R_h to represent the motion speed, shown in Figure 4(a). And then we calculate the reciprocal



(a)



(b)

Figure 4: Right hand motion relationship with human annotated onsets (drawn as circles). The triangles point out the onsets which can be correctly detected by the corresponding detection function. (a) The analysis of right hand motion along the bow (radius) (b) The analysis of right hand motion orthogonal to the bow (angle).

R_{hr} of R'_h and normalize R_{hr} into [0,1]. As illustrated in Figure 4(a), from R_{hr} we can tell there is more likely an onset when the value of R_{hr} is closer to 1. With this property, R_{hr} is defined as the detection function D_{hr} from hand motion R_h for onset detection.

Another observation of violin bowing is that the angle θ_{hi} changes during playing to produce new notes played by different strings from previous notes. However, the changing of angle θ_{hi} is much less frequent than the changing of bowing speed. Most of the time the bow angle keeps the same and when there is a sudden change of angle, a new note is likely produced, which means we could use the speed of bow angle change as the indication that a new note is produced and the new onset should be detected.

To model the above observation, we derive the first derivative Θ'_h of hand motion Θ_h to describe the bow angle changing speed. After normalization into [0,1], we use it as the detection function $D_{h\theta}$ for onset detection. As illustrated in Figure 4(b), when there is a high value in $D_{h\theta}$ close to 1, we would have more confidence that this timing corresponds to an onset produced.

5.3 Left Hand Fingers Motion Analysis

In addition to bowing, another critical technique of violin playing is fingering. As discussed in Section 5.1, four fingers move independently towards and away from the origin of finger polar coordinate system. Further, when the fingers try

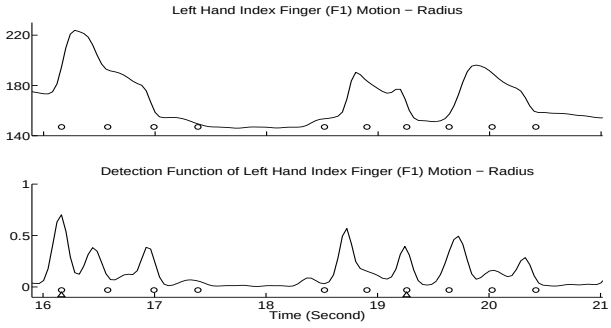


Figure 5: Fingers motion relationship with human annotated onsets (drawn as circles). The triangles point out the onsets which can be correctly detected by the corresponding detection function.

to release or press the strings, fast motions will be revealed in finger movements. Therefore, we conclude the fingers motion along radius direction of the finger polar system is of most interest. The fast speed of fingers motion could also be used to indicate the onset events.

5.3.1 Finger markers tracking

As different color markers are used to mark the four fingers, we use color marker tracking Algorithm 1 to track each finger individually to get the fingers motion trajectories (X_{fj}, Y_{fj}), $1 \leq j \leq 4$.

In the first frame of the input video sequence, we manually label the lower crossing point of the fingerboard and upper bout of violin as the initial origin $(x_{f0,1}, y_{f0,1})$ of the polar coordinate system for fingers motion decomposition. However, we cannot assume the origin does not move, because the camera is much closer (close-capturing) to the violin compared with the right hand situation and the movement of violin during playing must be considered. Therefore, we set a 50 by 50 square area in pixels around the initial origin, and use a motion estimation technique to find the best match positions $(x_{f0,i}, y_{f0,i})$ of this region in the following frames, and obtain the trajectory of the origin (X_{f0}, Y_{f0}).

The finger markers tracking and origin update are done simultaneously. After that, the origin trajectory (X_{f0}, Y_{f0}) and four finger trajectories (X_{fj}, Y_{fj}), $1 \leq j \leq 4$, are obtained for fingers motion decomposition.

5.3.2 Fingers motion decomposition

We follow the same approach to decompose the four finger trajectories (X_{fj}, Y_{fj}), $1 \leq j \leq 4$, in the original Cartesian coordinate system of the video frame into (R_{fj}, Θ_{fj}), $1 \leq j \leq 4$, of finger polar coordinate system. The difference we use the updated origin $(x_{f0,i}, y_{f0,i})$ for the current frame instead of using the initial origin of the first frame all the time. From the observation of violin fingering, the radius of fingers motion R_{fj}, $1 \leq j \leq 4$, is highly related with note production, but not the angle Θ_{fj}, $1 \leq j \leq 4$, of fingers motion. Therefore, we only calculate R_{fj}, $1 \leq j \leq 4$, during fingers motion decomposition.

5.3.3 Detection function of fingers motion

As discussed at the beginning of this subsection, we could use the fast speed of fingers motion to indicate onset timing. Therefore, we derive the first derivative R'_{fj} of R_{fj}, $1 \leq j \leq$

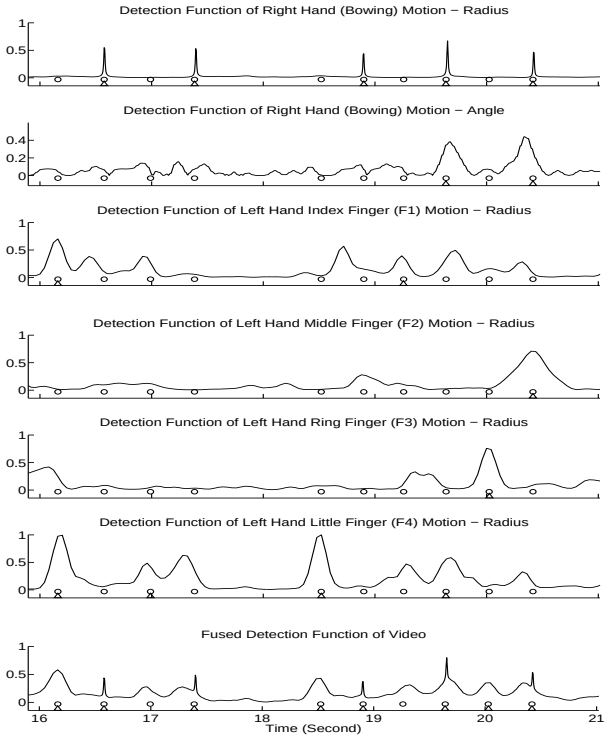


Figure 6: The intra-model data fusion of video. Human annotated onsets are drawn as circles. The triangles point out the onsets which can be correctly detected by the corresponding detection function.

4, as the speed functions of fingers, and further redefine it as the detection function for onset detection after normalizing into [0,1]. As illustrated in Figure 5, when the speed of one fingers motion is fast at a certain time, it is very likely that an onset is produced.

As each finger moves independently, we construct four independent detection functions D_{fj}, $1 \leq j \leq 4$, for onset detection.

6. SYSTEM INTEGRATION

In this section, we discuss how to fuse detection functions from both video and audio to create an audio-visual violin transcription system.

6.1 Data Fusion of Detection Functions

In order to clarify the relationship between audio and visual data, we divide the multimodal data fusion stage into an intra-model data fusion part addressing the data fusion of the six video detection functions and an inter-model data fusion part for the combination of the audio and video detection functions. That allows us to separately explore the relationship between the video detection functions first and then fuse audio and video detection functions at a higher abstraction level. In addition, this way of multimodal fusion renders system implementation and tuning less complex.

In order to synchronize different detection functions, we use linear interpolation to interpolate the six video detection functions and audio detection functions from their own sampling frequencies to 200 Hz.

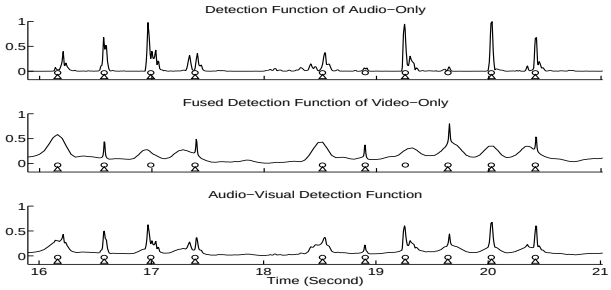


Figure 7: The inter-model data fusion.

6.1.1 *Intra-model data fusion of video model*

As can be seen from Figure 4 and Figure 5, not all onset timing points can be indicated by any of the six video detection functions; hand detection function D_{hr}, D_h and finger detection functions D_{fj}, $1 \leq j \leq 4$. Therefore, it is necessary to utilize information from all six detection functions to enable the extraction of more onset events. A combined video detection function Dv, is calculated, which we expect to correlate to the annotated note onsets.

Because of the clear correlation of each video detection function with onset events, we employ a simple data fusion method, weighting fusion, to fuse the D_{hr}, D_h and D_{fj}, $1 \leq j \leq 4$ to produce D_v. From the experimental results in Section 7, we see that this method of data fusion works well.

$$\begin{cases} D_v = Nor(w_{hr} \cdot D_{hr} + w_{h\theta} \cdot D_{h\theta} + \sum_{i=1}^{4} w_{fi} \cdot D_{fi}) \\ w_{hr} + w_{h\theta} + \sum_{i=1}^{4} w_{fi} = 1 \end{cases} \quad (2)$$

The fusion is formulated as Equation (2), in which w is the corresponding weight of each detection function D, the summation of the six weights equals 1, and Nor is the normalization function. The six original detection functions, D_{hr}, $D_{h\theta}$ and D_{fj}, $1 \leq j \leq 4$, obtained in Section 5 and the overall video detection function, D_v, are illustrated in Figure 6, from which we can clearly see that intra-model data fusion of video model makes the video detection D_v much more correlated with onset events than any single video detection function.

6.1.2 *Inter-model data fusion of audio and video*

After obtaining the overall video detection function D_v, we fuse it with the audio detection function by the same strategy, weighting fusion, used in intra-model data fusion of video. From the experimental results in next section, the simple fusion method is proven to work well.

$$\begin{cases} D_{av} = Nor(w_a \cdot D_a + w_v \cdot D_v) \\ w_a + w_v = 1 \end{cases} \quad (3)$$

The fusion is formulated as Equation (3). With two weights, w_a and w_v, specified for audio detection D_a and video detection function D_v the final audio-visual detection D_{av} is produced as the input of the next processing module, peak picking, of our system.

Figure 7 shows the fused audio-visual detection D_{av} compared with audio-only and fused video detection function.

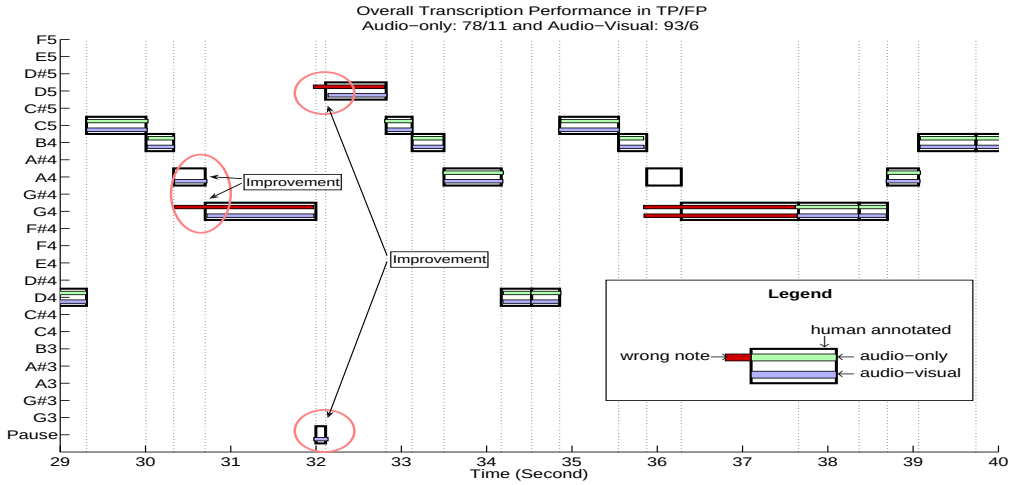


Figure 8: Example of a complete transcription result with audio-only and audio-visual methods.

As can be seen, D_{av} is more correlated with onset events than D_a and D_v. Interestingly, several onset events not revealed in the audio detection function are picked up in the combined audio-visual detection function, because of the aid of visual information.

This promising point of multimodal data fusion of audio and video models, to improve the onset detection performance and overall transcription system performance of violin transcription system, is verified by our experimentation discussed in Section 7.

6.2 Audio-Visual Violin Transcription

With the audio-visual detection function from the multimodal data fusion module, we describe in the following part the creation of an audio-visual violin transcription system.

From Figure 1, we can see that peak picking module picks the onset and offset timing points from the audio-visual detection function. As described in Section 4, we integrate two peak picking methods into our system: a fixed threshold based method and a median filter adaptive threshold based method. After peak picking, we get a series of timing points, T, representing the onsets and offsets in violin music. However, with the timing points, there is no information that indicates whether a certain timing point is an onset or offset.

In the pitch estimation module, we use onsets and offsets timing points T and audio wave signal X as input. Then an energy based approach is used to find active (not silent) note segments N_t to separate onsets and offsets. Then pitch estimation is performed based on each active note segment of audio signal.

After pitch estimation the transcribed notes, N, described by onset timing, offset timing and pitches are outputted as the final violin transcription results of our audio-visual violin transcription system.

Figure 8 shows an example of the transcription results, in which the human annotated notes are displayed as large and bright boxes, which our transcription system is compared against. For both methods, audio-only and audio-visual, the transcription results are drawn on top of the human annotations. Audio-only is represented by the upper thin boxes; the audio-visual method is displayed with the help of the lower thin boxes. A bright filling indicates that the onset and offset of the respective note as well as its pitch are correctly detected, whereas a dark red color represents errors in either or both criteria. In the example in this figure the audio-visual method is able to detect two onset events correctly, which are not obtainable with the audio-only method. One onset event is missed by both methods.

7. EVALUATION

This section provides a description of the setup of our system and discusses the results that our approach has yielded.

7.1 Violin Audio-Visual Database

All evaluations and tests were carried out against our audio-visual database of 16 violin recordings, half of which contain vibrato; performed by a professional violin player from the Conservatory of Music at our university. The recording took place in an indoor environment with regular lighting conditions. Overall the compositions contain 2157 onsets and notes, where pauses (silence) also are considered and to be recognized by our system as individual notes. Human annotation of the material was carried out and cross checked by different educated musicians. The procedure of locating the onsets in the audio was inspired by a work presented in [21]. Thus we are assured to have a reliable ground truth for the evaluation of our system.

7.2 Evaluation Procedure and Metric

The evaluation process undergoes several phases. During phase 1, in order to find an optimal parameter set for each method and promising candidates for a subsequent pitch estimation phase we assess different audio onset detection functions: an inverse correlation based method (Inverse Correlation), a pitch based method (Pitch based) and an equal loudness based method (Equal Loudness) (see Section 4.1). In particular, we create detection functions from all audio files using DFTs with all combinations of hop sizes and window sizes ranging from 256 samples to 4096 samples and 512 samples to 8192 samples respectively while the sampling frequency of the recorded pieces is at 44.1 kHz. We find for all detection functions that using a window size twice as large as the hop size works best, and finally the combination of hop

size 512 and window size 1024 yields the best performance, of which the results are shown in the following parts.

The suitability of all three detection functions was evaluated by measuring the onset detection performance using the two peak picking methods described earlier: the fixed threshold based method (F) and the median filter based adaptive threshold method (A). The tolerance for correctly detected onsets is 100 ms. The measure for the accuracy of each method is given in true positives (TP) and false positives (FP) calculated as:

$$TP = \frac{numberOfCorrectlyDetectedOnsets}{numberOfAnnotatedOnsets}$$

$$FP = \frac{numberOfWronglyDetectedOnsets}{numberOfAllDetectedOnsets}$$

The Receiver Operating Characteristics (ROC) curve of TP and FP is also plotted for further comparison of the performance of different methods over a range of thresholds. The definitions of TP, FP and ROC curve apply to the evaluation of pitch estimation and overall performance of the audio-visual violin transcription system.

For all detection functions combined with either of the two peak picking methods, the inter-model fusion weight of audio and video is evaluated by varying the weights for the audio and video detection functions from 1.0 weight for audio and 0.0 for video up to the inversed ratio of 0.0 weight for audio and 1.0 weight for video using a step size of 0.1 (see Figure 9 for illustration). The weights for intra-model data fusion of video streams are fixed based on the observation of violin playing as: $w_{hr} = 0.3$, $w_{h\theta} = 0.3$, $w_{f1} = 0.1$, $w_{f2} = 0.1$, $w_{f3} = 0.1$, $w_{f4} = 0.1$. In addition to the weights of audio and video, also the base threshold for both peak picking methods is assessed. The thresholds range from 0.001 to 0.5 with a granularity of 0.001 for values between 0.001 and 0.02 and a granularity of 0.01 for values between 0.03 and 0.5.

For pitch estimation assessment alone, the audio signal is split into audio segments according to the human annotated onsets. Klapuri's methods ([14, 15]) are specified in detail in his papers, so no further parameter evaluation is carried out. For the method based on the semitone spectrum the parameters α_1 and α_2 (see [16]) are evaluated in a range from 1 to 10 with a step size of 1 in order to find the most suited values for amplification of the fundamental frequency and the octave error correction. The overall transcription performance is evaluated by using the parameters that work best for each sub task and combining each method together to form a complete transcription system, consisting of onset detection and pitch estimation with silence detection.

7.3 Experimental Results

This part illustrates the evaluation results for onset detection, pitch estimation and the overall transcription performance of our system.

7.3.1 Onset detection performance comparison

Figure 9 illustrates the effect of the video detection function on the onset detection performance. As clearly can be seen when increasing the video fusion weight w_v from 0.0 (which corresponds to audio-only) towards a value of 0.5, the TP rises whereas the FP decreases. Progressing further towards a weight for the video detection function of 1.0 the performance is dropping. This illustration is an example of how we evaluate the optimum values for the audio-visual weights with fixed threshold for each detection function in combination with each peak picking method.

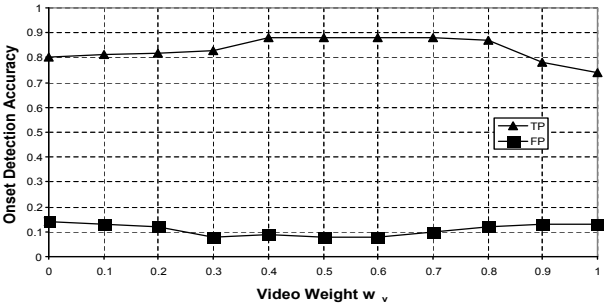


Figure 9: Effect of the video detection with different fusion weights on onset detection performance.

Table 1: Onset detection results.

Method	TP %	FP %	Best δ	w_v
Inverse Correlation F	**81**	**16**	**0.33**	–
Inverse Correlation & Video F	**88**	**16**	**0.09**	**0.8**
Inverse Correlation A	81	19	0.26	–
Inverse Correlation & Video A	88	23	0.18	0.5
Pitch based F	66	14	0.01	–
Pitch based & Video F	86	14	0.03	0.2
Pitch based A	**75**	**20**	**0.006**	–
Pitch based & Video A	**91**	**20**	**0.03**	**0.5**
Equal Loudness F	80	14	0.33	–
Equal Loudness & Video F	88	8	0.24	0.5
Equal Loudness A	**90**	**15**	**0.15**	–
Equal Loudness & Video A	**94**	**15**	**0.17**	**0.4**

Table 1 shows the experimental results of the six combinations with their own TP, FP, best threshold δ and video fusion weight w_v. As can be seen from Table 1, combined with the inverse correlation based detection function, the fixed threshold based peak picking method works better than the adaptive threshold peak picking method. Conversely, the adaptive threshold peak picking works better than the fixed threshold method if combined with pitch based and equal loudness based detection functions. Further, for the comparison of three detection function construction methods, in audio-only case, the equal loudness based method with adaptive threshold peak picking method performs the best with TP 90% and FP 15%. The second best method is the one with the inverse correlation based method and the fixed peak picking (81% TP, 16% FP). The pitch based method performs worst with only TP 75% and FP 20%. More interestingly, after fusing video information into audio, the performance of each method gets improved with an increase from 4% to 20% TP, which makes the onset detection more accurate to further improve the overall transcription performance.

The ROC curves of the six combinations (in bold in Table 1) are plotted in Figure 10. As can be seen, for each audio-only method the curve segment with reasonably high TP and low FP is shifted towards the left-top by a significant distance in the corresponding audio-visual curve, which clearly shows the advantage of multimodal data fusion of audio and video streams.

Finally, with multimodal data fusion of audio and video streams, for onset detection, we obtain 94% TP and 15% FP from equal loudness combined with adaptive thresh-

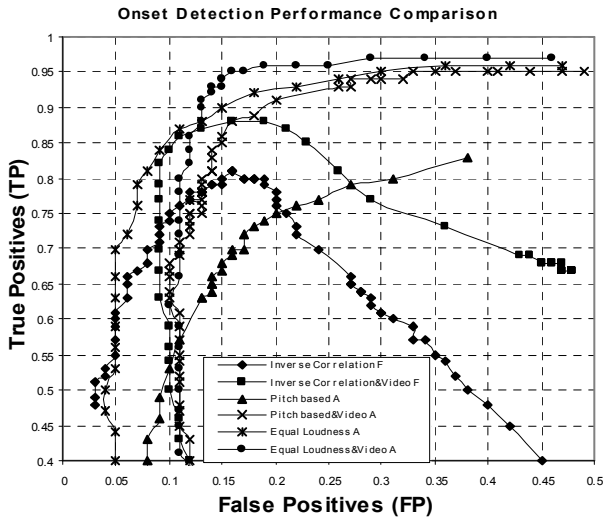


Figure 10: ROC curves of onset detection results.

old method (audio-only: 90% TP, 15% FP) and 88% TP and 16% FP from inverse correlation combined with fixed threshold method (audio-only: 81% TP, 16% FP). Because the audio-only performance of pitch based method is much worse than the other two methods, we do not include it for further system overall performance evaluation.

7.3.2 Pitch estimation performance comparison

Three pitch estimation performance methods are evaluated in this part in terms of accuracy and time efficiency.

Table 2: Pitch estimation results in terms of accuracy and time efficiency.

Method	TP %	FP %	Time/Note
Loscos 06 [16]	95	6	15.5 ms
Klapuri 05 [14]	93	7	1217.9 ms
Klapuri 06 [15]	94	7	719.5 ms

Table 2 summarizes the pitch estimation performance of Loscos 06 [16], Klapuri 05 [14], and Klapuri 06 [15] in terms of TP and FP of pitch estimation and average time spent for pitch estimation of one note. As can be seen, the pitch estimation method of Loscos 06 performs the best both in accuracy and time efficiency. For the whole database, it performs 95% TP and 6% FP with spending 15.5 milliseconds on average for pitch estimation of each note. Klapuri's two methods perform well in terms of accuracy. However, they are too time consuming for our intended e-learning applications.

7.3.3 Overall transcription performance comparison

Integrating Loscos 06 pitch estimation method into our violin transcription system, we further evaluate the overall transcription performance for inverse correlation with fixed threshold peak picking and equal loudness with adaptive threshold peak picking in both audio-only and audio-visual cases.

Table 3 illustrates the overall performance of the violin transcription system. Clearly we can see with audio-visual

Table 3: Overall transcription results.

Method	TP %	FP %	Best δ	w_v
Inverse Correlation F	62	27	0.19	–
Inverse Correlation & Video F	73	20	0.10	0.8
Equal Loudness A	74	36	0.14	–
Equal Loudness & Video A	83	28	0.16	0.5

data fusion, the overall performance improves with about a 10% TP increase and about an 8% FP reduction. The overall transcription performance results prove the multimodal data fusion of audio and video cues is very promising in application oriented violin transcription system.

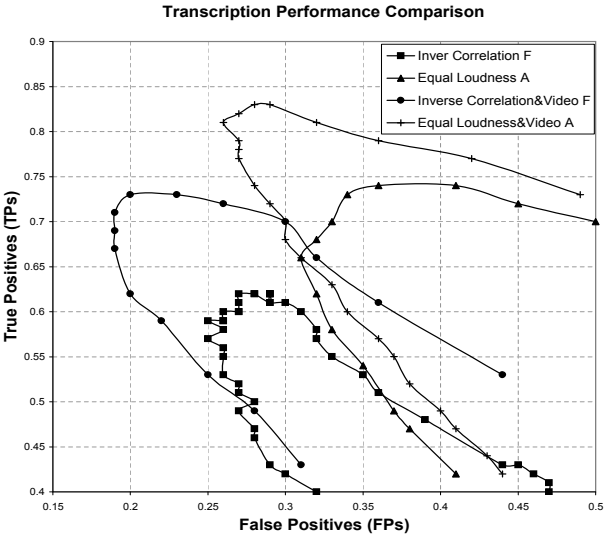


Figure 11: ROC curves of overall performance.

The ROC curves over a range of thresholds are plotted in Figure 11, where we can see that the ROC curve of audio-only transcription system is significantly shifted towards the left-top direction. That means by fusing video information into an audio-only transcription system, we can significantly improve the overall transcription accuracy.

8. DISCUSSION

Experimental results in this pilot study clearly verify our initial hypothesis that violin music transcription can be improved significantly by fusing audio and visual cues. However, there are many aspects that deserve in depth investigations in order to further enhance transcription performance.

We have developed this new approach with a clear application scenario; personalized violin education at home. That is, the system should be implemented with off-the-shelf hardware and be practical in home environments. The recording environment, student and teacher's home is clearly different in comparison with a professional recording studio. We can expect much less professional lighting and much higher noise level at home. Robustness of audio and video processing methods becomes a critical issue. We expect that multimedia fusion based methods will play a larger role in such applications.

Although we have chosen the violin as the instrument for

our method, we believe that the main design considerations generalize to other instruments.

We note that visual information not only is helpful for violin transcription, but also serves as a direct visual feedback which helps students to rectify bad playing habits even if correct notes are produced based on audio analysis.

9. CONCLUSIONS

We have presented the first attempt and experience with a violin music transcription system fusing audio and visual cues. It incorporated state-of-the-art audio-only music transcription methods as the baseline. We used a simple multimedia fusion technique for the proof of concept. Our experimental results demonstrate that multimodalities are superior to single modality in note segmentation.

Our project has lead to several innovations in combining audio and video processing. In audio processing, we have demonstrated that instrument-specific methods performed better than generic methods in terms of accuracy and complexity. In our video processing module, we have proposed novel methods to track bowing and fingering trajectories effectively for the purpose of enhancing violin music transcription. To integrate the system, we have explored intra-model and inter-model feature integrations.

The proposed method can be improved in many ways. The transcription performance of our method is not yet good enough for real life applications. We have taken an early data fusion approach in our system, but could investigate data-fusion in different stages, for example, late data-fusion in combination with machine learning methods. These are important areas for future work. Furthermore, our observation shows that an intelligent application of haptic sensors could improve music transcription.

To broaden its applicability, we have started to investigate various methods of finger and bow tracking with and without markers. Music transcription has many applications such as music education, which requires very high transcription speed and accuracy. We have taken the first step to enhance music transcription of string instruments by fusing multimedia streams in the hope to enhance the system performance which can satisfy end-users' requirements.

10. ACKOWNLEDGEMENT

Hugh Anderson is acknowledged for proofreading an earlier version of this paper. The three anonymous reviewers are acknowledged for their critical comments and constructive suggestions.

11. REFERENCES

[1] Yin J., Wang Y. and Hsu D., Digital Violin Tutor: An Integrated System for Beginning Violin Learners, *ACM Multimedia Conf.*, 2005.

[2] Perkins, D., *Smart Schools: Better Thinking and Learning for Every Child*, The Free Press, New York, 1992.

[3] Collins, N., A Comparison of Sound Onset Detection Algorithms with Emphasis on Psycho-Acoustically Motivated Detection Functions, *Journal of the Audio Engineering Society,* 2005.

[4] Bello, J. B., Daudet, L., Samer, A., Duxbury, C., Davies, M. and Sandler, M. B., A Tutorial on Onset Detection in Music Signals, *IEEE Trans. on Speech and Audio Processing,* pp: 1035-1047, 2005.

[5] Gillet O. and Richard G., Automatic Transcription of Drum Sequences using Audiovisual Features. *Proc. IEEE Int. Conf. on Acoustics, Speech, and Signal Processing,* 2005.

[6] Baader A. P., Kazennikov O., and Wiesendanger M., Coordination of bowing and fingering in violin playing, *Cognitive Brain Research,* pp:436ÍC443, 2005.

[7] Nakamura, S., Statistcal Multimodal Integration for Audio-Visual Speech Processing, *IEEE Trans. on Neural Networks,* pp:854-866, 2002.

[8] Garg, A., Potamianos, G., Neti, C., and Huang, T. S., Frame-dependent multi-stream reliability indicators for audio-visual speech recognition, *Proc. IEEE Int. Conf. Multimedia and Expo (ICME03),* pp:605-608, 2003.

[9] Kaynak, M.N., Qi Z., Cheok, A.D., Sengupta, K., Zhang J., Ko C.C., Analysis of lip geometric features for audio-visual speech recognition, *IEEE Trans. Systems, Man, and Cybernetics,* pp:564- 570, 2004.

[10] Fragopanagos, N. and Taylor, J. G., Emotion recognition in human-computer interaction, *Neural Network,* pp:389-405, 2005.

[11] Foote, J., Automatic Audio Segmentation Using a Measure of Audio Novelty, *Proc. IEEE Int. Conf. Multimedia and Expo (ICME00),* pp:452-455, 2000.

[12] Collins, N., Using a Pitch Detector for Onset Detection, *Proc. of ISMIR2005,* 2005.

[13] Boo W., Wang Y., Loscos A., A Violin Music Transcriber for Personalized Learning, *IEEE Inter. Conf. on Multimedia Expo*, 2006.

[14] Klapuri, A., A perceptually motivated multiple-f0 estimation method, *Proc. IEEE Workshop on Applications of Audio Signal Processing to Audio and Acoustics,* pp: 291- 294, 2005.

[15] Klapuri, A., Multiple Fundamental Frequency Estimation by Summing Harmonic Amplitudes, *Proc. of ISMIR2006,* pp:216-221, 2006.

[16] Loscos A., Wang Y., Boo W., Low Level Descriptors for Automatic Violin Transcription, *Proc. of ISMIR2006*, 2006.

[17] Flesch C., *Art of Violin Playing: Book One,* Carl Fischer Music Dist, 2000.

[18] Letessier J. and Brard F., Visual tracking of bare fingers for interactive surfaces, *Seventeenth Annual ACM Symposium on User Interface Software and Technology,* pp:119-122, 2004.

[19] Burns, A. and Wanderley, Visual methods for the retrieval of guitarist fingering, *Proc. of the 2006 Conf. on New interfaces For Musical Expression,* pp:196-199, 2006.

[20] Wu Y., Lin J., Huang T.S., Analyzing and Capturing Articulated Hand Motion in Image Sequences, *IEEE Trans. Pattern Anal. Mach. Intell,* pp:1910-1922, 2005.

[21] Leveau, P., Daudet, L., Richard G., Methodology and Tools for the Evaluation of Automatic Onset Detection Algorithms in Music, *Proc. of ISMIR2004,* pp:72-75, 2004.