THREE METHODS FOR PIANIST HAND ASSIGNMENT

Aristotelis Hadjakos
TU Darmstadt
telis@tk.informatik.tu-darmstadt.de

François Lefebvre-Albaret
IRIT Toulouse
lefebvre@irit.fr

ABSTRACT

Hand assignment is the task to determine which hand of the pianist has played a note. We propose three methods for hand assignment: The first method uses Computer Vision and analyzes video images that are provided by a camera mounted over the keyboard. The second and third methods use Kalman filtering to track the hands using MIDI data only or a combination of MIDI and inertial sensing data. These methods have applications in musical practice, new piano pedagogy applications, and notation.

1 INTRODUCTION

This paper presents three methods for real-time hand assignment that are able to determine which hand of the player has played a note. The first method uses Computer Vision to detect the hands in video images provided by a camera mounted over the keyboard. The second method uses MIDI data only while the third method combines MIDI and movement data from inertial sensors worn on the player's wrist. Our methods can be used to improve existing notation applications. Furthermore, we see applications for musical practice and upcoming piano pedagogy applications. In the following, we provide descriptions of applications that would benefit by using our methods:

1) Hand-instrument mapping: Current electronic keyboards, e.g., the Korg X5 [16], typically allow to separate the claviature into two areas, one for the left hand and one for the right, so that the player can play a different instrumental sound with the left hand than with the right. However, to do so, each hand is confined to a fixed area, which is contrary to normal piano practice. Hand assignment methods could eliminate the need for this static boundary and enable a more natural playing experience.

2) New piano pedagogy applications: Sonification of playing movements has a potential to help piano students to become more aware of their playing movements, which can help to improve their technique. One way to perform the sonification is to modify the timbre of a played note according to the movement that leads to it. In order to do so, the computer has to know which hand has played a note so that the movement signal of the corresponding arm is evaluated. We are currently developing such an application based on data from inertial sensors, which are attached to the user's arm.

3) Notation: To notate a MIDI-recording of a piano performance, it is necessary to perform hand assignment to assign the notes to the correct note system. Current notation software typically assigns notes to hands based on their position relative to a split point. Hand assignment methods could minimize the amount of post-editing by the user.

When comparing the three methods within oneanother each method exposes individual advantages and disadvantages. The camera-based method provides the best accuracy rate, followed by the sensor-based method and the MIDI-based method. However, when using the MIDI-based method no additional hardware is needed. This makes the MIDI-based solution ideal for improving existing notation applications. In order to use the sensor-based method, the user has to attach two clock-like sensors to his wrists, which can be done in an instant. However, to use the camera-based method, the user has to mount the camera over the keyboard. Therefore, the sensor-based method has an advantage over the camera-based method if mobility is important for the user, e.g., if the player frequently has to bring her equipment to rehearsals or concerts. Adverse lighting conditions on a concert stage can be problematic for camera-based hand assignment. The sensor- and MIDI-based methods on the other hand are not influenced by stage conditions. Finally, the sensor-based and the MIDI-based methods are computationally cheaper than the camera-based method and can therefore run on an ordinary microcontroller. Finally, the sensor-based and the MIDI-based methods are computationally cheaper than the camera-based method, making it possible to run them on an ordinary microcontroller. This makes it possible to build a self-contained pedagogical sonification application, which could This makes it possible to build self-contained pedagogical sonification applications that do not rely on an additional laptop or desktop computer.

The remaining paper is structured as follows. Section 2 discusses related work. The three hand assignment methods are presented in the sections 3 to 5. We provide an evaluation in section 6. Conclusion are presented in section 7.

SMC 2009, July 23-25, Porto, Portugal

2 RELATED WORK

In this section we report methods for hand assignment and hand tracking. We will discuss voice-separation techniques based on MIDI data and camera-based approaches for hand tracking. Methods for hand assignment using data from inertial sensors are not discussed in literature.

2.1 Methods based on MIDI

Kilian and Hoos proposed a method that finds a separation of a piece into different voices for notation [9]. Chords are allowed to occur in one voice. The method allows the user to select the number of present voices. Therefore, it can be used to find a left hand and right hand part of a MIDI performance (see [9] for notated examples). To separate voices, Kilan's and Hoos' method splits the piece into a sequence of slices with overlapping notes and finds the voice separation by minimizing an elaborate cost function using a stochastic local search algorithm. This approach, while reasonable for notation, cannot be used for real-time hand assignment of a live performance because the slice of overlapping notes cannot be immediately determined when a note is received. Furthermore, the stochastic local search algorithm operates on the entire piece. Other voice separation methods [1, 7, 11] do not allow chords inside a voice and can therefore not be used for hand assignment.

2.2 Methods based on Computer Vision

To detect the two hands in the video, most of the studies make use of a skin color model. To be more robust to illumination changes, other color spaces than RGB, like YUC or HSV, are often used for hand tracking. The hand color distribution can then be modeled as a histogram, a mixture of gaussian, or any other parametric model [14]. Recent studies propose to combine the color information with a displacement information between two consecutive frames [4]. Hands are then identified by their motion and color. Edge detection is often used to refine the estimation of hand shape. After the hand pixels are detected, several algorithms can be applied for hand tracking (an overview is provided in [12]). Algorithms like CamShift, CONDENSATION, etc. give very robust and accurate results as long as there is no hand occlusion. However, they often fail at labeling the right and left hand correctly after a big occlusion. However, overlappings and occlusions frequently occur in piano playing.

Gorodnichy and Yogeswaran developed a system for hand assignment that relies on visual tracking [3]. The system finds the position of the keyboard in the video and identifies the Middle C key. Background subtraction is used to find the hands in the image. Through the identification of cervices in the hand image, fingers are detected. The system annotates MIDI recordings with hand and finger labels. In contrast to our camera-based method, crossing over of the hands is not considered.

3 HAND ASSIGNMENT WITH COMPUTER VISION

In this section, we describe our Computer Vision based tracking algorithm. The pixels belonging to the hands are detected by their color. The hand tracking is performed with a particle filter. The disambiguation of the two hands is achieved by taking into account the principal direction of the hand shape and motion continuity. Finally, the bounding box of the hand is refined to provide accurate positions for the following hand assignment.

3.1 Hand tracking

The detection of skin pixels is made using a Bayesian approach. The skin model is learned form the HSV space of a skin picture that can be changed according to the pianist's skin color. The back-projection of the skin color provides a map of skin color probabilities. This method gives acceptable data for hand-tracking but is not sufficiently accurate to find the hand shape because of the spectral reflexion on the keyboard (see Figure 1). The hand-tracking is made using an annealed particle filter inspired by [2], where each hand is tracked by one cloud of particles. During the tracking process, the cloud pixels of each hand are alternatively subtracted from the skin detection map so that each cloud converges to a different hand. Hand positions are located at the centers of gravity of the particle clouds.

3.2 Hand disambiguation

The problem of hand disambiguation is hard to solve, especially when the two hands overlap frequently. To overcome this problem, we adapted a method originally designed for French Sign Language video processing [10] for pianist hand detection. Hand shape orientation is used as disambiguation criterion. The principal axis of each hand shape is computed after a bounding box has been determined for each hand (see Figure 1) .

We model the joint distribution of the pairs (α_r, α_l) as a gaussian probability density function $f(\alpha_r, \alpha_l, \theta_\alpha)$. Given a pair $(\alpha_1(t), \alpha_2(t))$ of hand shape orientation at the time t, the confidence measure that the hand 1 is the right hand can be computed with the following log-likelihood ratio.

$$llr(t) = \frac{f(\alpha_1(t), \alpha_2(t), \theta_\alpha)}{f(\alpha_2(t), \alpha_1(t), \theta_\alpha)} \quad (1)$$

Continuity of the movement is used as an additional criterion since the hand shape orientation is not sufficient to provide a robust distinction between the hands. While the log-likelihood ratio typically detects the hands correctly, it

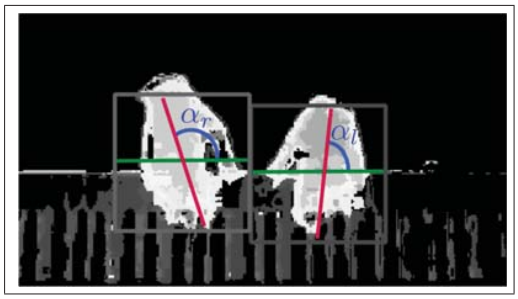

Figure 1. Hand shape orientations

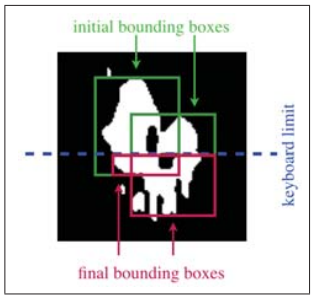

Figure 2. Bounding box refinement

fails if the hands adopt an unusual hand posture. In these occasions the continuity criterion ensures that the hands are not confused.

The two criteria are embedded in an optimization function that has to be maximized between the beginning of the video and the current frame The optimization is achieved with the Viterbi algorithm. Let $dist(t, t-1)$ be the sum of displacements of the right and the left hand between the frame $t-1$ and t then the global optimization function can be written as:

$$argmin \sum_{t} dist(t, t-1) - c \cdot llr(t) \qquad (2)$$

The value of the weight c was empirically determined.

3.3 Bounding box refinement

Once the two hands have been localized by the particle filter, it is important to localize the part of the hand that is located over the keyboard more precisely by making a bounding box refinement.

Once the two hands have been localized by the particle filter, the intersection of the hand with the keyboard is determined since a large part of the hand may be localized behind the keyboard. To this end, we use bounding box refinement. First, the background is subtracted from the hand image. Outliers are removed using thresholds and morphological operators. The actual hand shape is then considered as being the intersection of a large bounding box surrounding the hand centroid and the keyboard area. As visible in Figure 2, this approach finds the bounding boxes of the hands reliably, even if the two hands overlap.

3.4 Hand assignment

Hand assignment is performed by comparing the horizontal position of the played key in the video with the boundaries of the hands. The decision procedure takes into account whether the played key is inside the span of one hand, both hands, or outside both hands. If the key is inside the span of both hands, it is assigned to the hand where the key more inside the hand span. If the key is outside the span of both hands, it is assigned to hands based on distance. If one hand is outside the keyboard area, no notes will be assigned to it. To calculate the horizontal position of the played key, the procedure uses information about the keyboard position in the video, which is provided once by the user in a visual configuration dialogue.

4 HAND ASSIGNMENT WITH MIDI

The methods for MIDI-based hand assignment (described in this section), and hand assignment based on sensor and MIDI data (described in the next section) are closely related. Both methods are composed of a series of two steps. In the first step a received note-on event is assigned to the left or right hand. In the second step the note-on event is used to modify the estimated position of the corresponding hand.

4.1 Hand assignment

Hand assignment of a note is done with two mechanisms: the identification of unique notes and the examination of the distances of the played note to the estimated hand positions. The method does not allow crossing over of the hands so that the left hand has to be located left of the right hand. It is possible to find simultaneously pressed keys that are located too far from each other to be played by one hand, a condition that will be called a *unique note*. As the hands are not allowed to cross over unique notes can be directly assigned to the left or right hand. Unique notes are identified as notes with an interval of more than an eleventh to the highest or lowest currently pressed key as most players cannot grasp such intervals. If a note is not an unique note, it is assigned to a hand based on the distance of the note to the estimated hand positions..

The position of the hands are estimated with a Kalman filter for each hand. The received note-on events are handed over to the Kalman filter of the assigned hand.

4.2 Position estimation

For each hand, a Kalman filter is used to estimate the position of the hand. The state p of the filter is the position of the center of the hand. The position p is expressed in MIDI units. For example, let the center of the hand lie between the keys corresponding to MIDI pitch values of 60 and 61. Then the position p would be 60.5. The uncertainty of the position is expressed by the variance σ_p^2. The uncertainty of the position decreases when a measurement of hand position is obtained and increases otherwise.

A received note-on event is interpreted as an approximate measurement of hand position. The variance σ_m^2 expresses the uncertainty involved in the measurement. When a note-on message is received, the variance expressing the uncertainty in the position prior to incorporating the measurement $\sigma_p^2(t_2^-)$ is computed. Let t_1 be point in time when the last note was assigned to the Kalman filter and t_2 be the point in time when the new note was received. The uncertainty of the position before incorporating the new measurement $\sigma_p^2(t_2^-)$ is then updated based on the time difference between the two notes $t_2 - t_1$, a constant term σ_s^2, and the previous uncertainty after incorporating the measurement $\sigma_p^2(t_1^+)$.

$$\sigma_p^2(t_2^-) = \sigma_p^2(t_1^+) + (t_2 - t_1) \cdot \sigma_s^2 \tag{3}$$

The uncertainty of the position after incorporating the measurement $\sigma_p^2(t_2^+)$ is updated based on the uncertainty of the position before incorporating the measurement $\sigma_p^2(t_2^-)$ and the constant term σ_m^2 that expresses measurement uncertainty.

$$\sigma_p^2(t_2^+) = \sigma_p^2(t_2^-) - \frac{\sigma_p^2(t_2^-)}{\sigma_p^2(t_2^-) + \sigma_m^2}\sigma_p^2(t_2^-) \tag{4}$$

Let n be the pitch of the note received at t_2. Then the new position $p(t_2)$ is estimated based on the old position $p(t_1)$, the uncertainty of the position before incorporating the measurement $\sigma_p^2(t_2^-)$, and the pitch of the received note n.

$$p(t_2) = p(t_1) + \frac{\sigma_p^2(t_2^-)}{\sigma_p^2(t_2^-) + \sigma_m^2}(n - p(t_1)) \tag{5}$$

Values for σ_s^2 and σ_m^2 were empirically determined.

4.3 Discussion

This section illustrates the method with an example. Say, a user repeatedly plays two notes that are one octave apart with one hand. The first note is played after the hand has been inactive for some time. Therefore, the uncertainty of the hand position is high according to equation 3. Because of the high uncertainty, the new measurement has great influence on the estimated hand position according to equation 5 and the new estimated position will be near the pressed key. The uncertainty of the position reduces because of the new measurement according to equation 4. Because the position uncertainty has been reduced, the next note, which is played one octave apart, receives less weight so that the new position is between the first and second note, slightly towards the second. After several touches, the position uncertainty levels off at a low value controlled by equations 3 and 4 and execution speed. Therefore, new measurements do not drastically change the estimated position. The estimated hand position lies between the two alternating notes and only slightly oscillates when new measurements are made. If the user changes the position of the hand, the estimated position will adapt as older measurements loose influence over time according to equation 3.

5 HAND TRACKING WITH INERTIAL MEASUREMENT AND MIDI

The method described in the previous section can be improved by using measurement of arm movement. This section details on the method based on inertial measurement and MIDI.

To re-position the hand, a player can use various movements of the arm and the body. Despite the many possibilities to move the hand to a given position, players usually reach a position with consistent body and arm posture. Therefore, the angle between the player's forearm and the keyboard can be interpreted as an indication for the position of the hand. The rate of change of this angle can be obtained from an inertial sensor attached to the wrist of the player. However, this measurement provides only information of position change. To obtain absolute hand position, the inertial measurement is combined with the MIDI through Kalman filtering [8].

Similar to the MIDI-based method, unique notes are assigned to the corresponding hand; non-unique notes are assigned to the hands based on the distances of the played note to the positions of the hands.

5.1 Arm movement measurement

To determine the rate of change of the angle between the forearm and the keyboard, which will be called the rate of sideways movement for simplicity, it is necessary to obtain the orientation of the sensor toward gravity. It would be possible to calculate pitch and roll angles directly from the accelerometer signal. However, the playing movements create additional sources of acceleration, which would adversely affect the accuracy. To improve the accuracy of the calculated pitch and roll angles, Kalman filtering is used to fuse accelerometer and gyroscope signals. Given the pitch and roll angle, the rate of sideways movement is calculated from the gyroscope signals.

5.2 Posture measurement

It is necessary to be able to convert a given angle between forearm and keyboard to a hand position (in MIDI pitch units) and vice versa. Because of different movement habits, the relation between playing position and angle has to be measured for each player individually. To this end, the player executes several touches with the same finger in a distance of, for example an octave, over the entire playing range of the keyboard. The change of the angle between two played notes is measured by summation of the rate of sideways movement. The measurement has to be performed for both hands. To convert from hand position to angle between forearm and keyboard and vice versa, linear interpolation between the measured values is used.

5.3 Signal fusion

For each arm, a Kalman filter is used to fuse MIDI and inertial measurement data. The state of the filter is the angle θ between the forearm and the keyboard.

When a new inertial measurement sample is received, the angle is updated. The new angle θ_{i+1} is computed based on the previous angle θ_i, the rate of sideways movement s, and the sample time dt.

$$\theta_{i+1} = \theta_i + s_i \cdot dt \tag{6}$$

Crossing over of the hands is not supported and is avoided by setting s to zero if it would lead to a crossing over condition.

The uncertainty of the angle θ is expressed by the variance σ_θ^2. The uncertainty of the angle θ increases based on σ_s^2, which is the variance of the rate of sideways movement, and the sample time dt.

$$\sigma_{\theta,i+1}^2 = \sigma_{\theta,i}^2 + \sigma_s^2 \cdot dt \tag{7}$$

When a note is assigned to the Kalman filter, the corresponding angle has to be calculated (see section 5.2). The measurement has an effect on the estimated angle and reduces the uncertainty of the angle. Let ϕ be the angle that corresponds to the pressed key that is assigned to the Kalman filter. The new estimate of the angle θ_{i+1} is calculated based on the previous angle θ_i, the previous uncertainty of the angle $\sigma_{\theta,i}^2$, and the angle ϕ.

$$\theta_{i+1} = \theta_i + \frac{\sigma_{\theta,i}^2}{\sigma_{\theta,i}^2 + \sigma_m^2}(\phi - \theta_i) \tag{8}$$

The uncertainty of the position is calculated based on the previous uncertainty and the measurement accuracy which is represented by the constant σ_m^2.

$$\sigma_{\theta,i+1}^2 = \sigma_{\theta,i}^2 - \frac{\sigma_{\theta,i}^2}{\sigma_{\theta,i}^2 + \sigma_m^2}\sigma_{\theta,i}^2 \tag{9}$$

Values for σ_s^2 and σ_m^2 were empirically determined.

6 EVALUATION

To evaluate hand assignment accuracy, performances of different piano pieces were analyzed with our methods. To this end, video, MIDI, and inertial measurement signals were recorded with one pianist playing different pieces. The recordings were conducted with the inertial sensors that we previously developed [5], which provide six degrees of freedom measurements of acceleration and angular rates at an update frequency of 100 Hz.

Simple approaches to automatically evaluate the hand assignment results, for example by using score-following to match the obtained separation with a given correct separation, are problematic because of playing errors and differences because of ornamentation. Therefore, the results were manually examined. To this end it was necessary to present the result in a human-readably way. We used the text-based GUIDO format [6, 13] to create graphical musical scores for the left and right hand part. The human reader can then identify correct and wrong assignments.

The recorded pieces were the Sinfonias 1–5 by J. S.Bach (BWV 787–791) and the "Six Dances in Bulgarian Rhythm" (No. 148–153) from Bartok's Mikrokosmos vol. 6. Bartok's dances contain many instances where the hands overlap. Furthermore the dances contain frequent changes of hand position on the keyboard, which are performed very quick re-positioning movements. Also the hands are crossing over several times. Therefore, the dances are especially challenging for hand assignment.

The accuracies of the obtained hand assignments are shown in Table 1. For comparison with a baseline, we include the results of hand assignment with the split point method (split point is the Middle C). For all examined pieces, the our methods achieves better results than the split point method. The sensor-based method typically achieves better results than MIDI-based method. The camera-based method typically achieves better results than the MIDI-based and sensor-based methods.

The Bulgarian dance No. 152 shows a limitation of our methods. The hands often completely overlap in this piece, i.e., one hand is positioned above the other hand while both hands play notes in the same range. This is contrary to the assumptions of our methods, as they implicitly split the keyboard at a (time-variable) split-point. Therefore, our methods are not able to perform hand-assignment correctly if, e.g., the right hand plays a note that lies between two notes that are played with the left hand.

The camera-based hand disambiguation allows hands to be identified during crossing over. The disambiguation is successful when the hands cross over distinctly. To improve the disambiguation, the probability model (see equation 1) could be extended to additionally include the positions of the hands as the distribution of hand orientation changes according to its position on the keyboard.

Table 1. Hand assignment accuracy

Piece	Split	MIDI	Inert.	CV
Sinfonia 1	86.6%	97.6%	98.6%	97.8%
Sinfonia 2	86.4%	94.3%	97.0%	98.6%
Sinfonia 3	94.2%	97.3%	98.3%	99.7%
Sinfonia 4	90.9%	97.5%	98.6%	98.9%
Sinfonia 5	97.2%	99.4%	99.3%	99.6%
No. 148	81.8%	88.7%	91.2%	94.2%
No. 149	79.4%	82.5%	89.6%	88.1%
No. 150	81.9%	86.4%	83.6%	90.8%
No. 151	69.8%	83.9%	87.8%	93.6%
No. 152	65.2%	66.6%	68.4%	70.6%
No. 153	78.2%	85.6%	91.2%	92.5%

The and sensor- and MIDI-based methods could be improved by using a hand model that filters out impossible hand-note configurations, which could be used instead of the unique-note mechanism (see section 4.1).

7 CONCLUSION

The main contributions of this paper are three methods for hand assignment: The first method is based on video images from a camera mounted over the keyboard, the second method is based on MIDI, and the third method combines inertial measurement and MIDI. The methods are real-time capable and can therefore be used for hand assignment in interactive scenarios like hand-instrument mapping and for new piano pedagogy applications. The methods were evaluated by running them on performances of pieces by Bach and Bartok. Applications of our methods are instrument-hand mapping, new piano pedagogy applications, and notation applications.

8 REFERENCES

[1] Chew, E. and Wu, X. "Separating voices in polyphonic music: A contig mapping approach", *Computer Music Modeling and Retrieval: Second International Symposium*, 2004.

[2] Gianni F., Collet C., Dalle P. "Robust tracking for processing of videos of communications gestures." *GW 2007*, 2007.

[3] Gorodnichy, D. O. and Yogeswaran, A. "Detection and tracking of pianist hands and fingers", *CRV '06: Proceedings of the The 3rd Canadian Conference on Computer and Robot Vision (CRV'06)*, 2006.

[4] Habili, N., Lim, C. C., and Moini, A. "Segmentation of the face and hands in sign language video sequences using color and motion cues", *IEEE Trans. Circuits Syst. Video Techn.*, 14(8), 2004.

[5] "SYSSOMO: A Pedagogical Tool for Analyzing Movement Variants Between Different Pianists", *Enactive08 Proceedings*, 2008.

[6] Hoos, H. H., Hamel, K. A., Renz, K., and Kilian, J. "The GUIDO Music Notation Format - A Novel Approach for Adequately Representing Score-level Music", *ICMC'98 Proceedings*, 1998.

[7] Jordanous, A. "Voice Separation in Polyphonic Music: A Data-Driven Approach", *ICMC 2008*, 2008.

[8] R. E. Kalman. "A New Approach to Linear Filtering and Prediction Problems", *J. Basic Eng.*, 1960.

[9] Kilian, J. and Hoos, H.H. "Voice separation—a local optimisation approach", *ISMIR 2002*, 2002.

[10] Lefebvre-Albaret F., Dalle P. "Body posture estimation in a sign language video", *GW 2009*, 2009.

[11] Madsen, S.T. and Widmer, G. "Separating voices in MIDI", *Proceedings of the 9th International Conference in Music Perception and Cognition (ICMPC2006)*, 2006.

[12] Mahmoudi, F. and Parviz, M. "Visual Hand Tracking Algorithms", *Geometric Modeling and Imaging–New Trends*, 2006.

[13] Renz, K., "An Improved Algorithm for Spacing a Line of Music", *Proceedings of the ICMC 2002*, 2002.

[14] Vezhnevets, V., Sazonov, V., Andreeva, A. "A Survey on Pixel-Based Skin Color Detection Techniques". *Proc. Graphicon-2003*, 2003.

[15] Zieren, J., Unger, N., Akyol, S., "Hands Tracking from Frontal View for Vision-Based Gesture Recognition" LNCS 2449, Springer, 2002.

[16] Korg X5 Music Synthesizer AI2 Synthesis System Bedienungshandbuch, 1994.