A Computer Vision Based Prototype for Automatic Guitarist Fingering Retrieval

Anne-Marie Burns and Marcelo M. Wanderley

anne-marie.burns@mail.mcgill.ca
marcelo.wanderley@mcgill.ca

Input Devices and Music Interaction Lab,
Schulich School of Music, McGill University,
Montréal, Québec, Canada
http://www.music.mcgill.ca/musictech/idmil/

Abstract. This article presents a method to visually detect and recognize fingering gestures of the left hand of a guitarist. The choice of computer vision to perform that task is motivated by the absence of a satisfying method for realtime guitarist fingering detection. The development of this computer vision method follows preliminary manual and automated analyses of video recordings of a guitarist. These first analyses led to some important findings about the design methodology of such a system, namely the focus on the effective gesture, the consideration of the action of each individual finger, and a recognition system not relying on comparison against a knowledge-base of previously learned fingering positions. Motivated by these results, studies on three important aspects of a complete fingering system were conducted. One study was on realtime finger-localization, another on string and fret detection, and the last on movement segmentation. Finally, these concepts were integrated into a prototype and a system for left-hand fingering detection was developed. Such a data acquisition system for fingering retrieval has uses in music theory, music education, automatic music and accompaniment generation and physical modeling.

1 Introduction

The sound produced by an instrument is influenced by its physical characteristics but also by the way the musician interacts with it. On an instrument like the guitar, both hands perform a distinct but complementary set of actions. The left-hand fingering gesture is the action performed by the musician to modify the string vibration length and, therefore, determines its pitch. The point where the musician presses the string against a fret is called the fingering point.

Generally, many different fingering points can be used to produce the same pitch. In fact, each pitch can be fingered at one to four fret positions, and theoretically each fingered position could be played by any of the four fingers. Consequently, for a score containing n notes, there can exist a maximum of

16^n combinations of $(string, fret, finger)$. However, a professional musician will only consider a few of these possibilities. The choice of the appropriate fingering will therefore be determined by many factors, including philological analysis (interpretation of a sequence of notes), physical constraints due to the musical instrument, and biomechanical constraints in the musician-instrument interaction [16]. Although the appropriate fingering might be obvious and intuitive for the experienced musician, beginners will often need external guidance because fingering indications are not always included in scores [10, 11].

Realtime guitarist gestures have been used to control sound synthesis for approximately thirty years either by capturing these with devices added to traditional guitars or by using guitars-shaped controllers. In the 1970's, when the popularity and availability of keyboard synthesizers were growing, "it came to pass that guitar players would have the same performance potential as keyboard players" (ARP instruments advertisement cited by [18]). The desire to capture guitarist gestures in realtime to control sound effects and sound synthesis was born and is still an active research topic today. Commercial solutions exist for the acquisition of the left-hand fingering gesture of the guitarist but up to now, these solutions only solve the $(string, fret)$ component of the problem.

Fingering retrieval is an important topic in music theory and performance. Guitarist fingering has being studied for educational purpose, to help beginners or non-musically trained amateurs [19, 13] and as a compositional help for non-guitarists [17]. It has also been studied for producing more realistic sounds in guitar physical models [6, 12] and in image modeling of a guitarist playing [8]. Also, it has impact in automatic music generation [4], and in score and tablature generation.

This articles presents an alternative method for real-time retrieval of the fingering information from a guitarist playing a musical excerpt. It relies on computer analysis of a video recording of the left hand of the guitarist and is inspired by the work of the composer Roberto Doati and InfoMus laboratory on the use of computer vision combine with color markers to detect guitarist fingering [7]. The first part of this article discusses the preliminary manual and automated analyses of multiple-view video recordings of a guitarist playing a variety of musical excerpts. The subsequent sections present studies on three aspects of visual analysis of a guitarist fingering: finger localization, string and fret detection, and movement segmentation. Finally, a system integrating these three components is presented and evaluated.

2 Preliminary Analysis

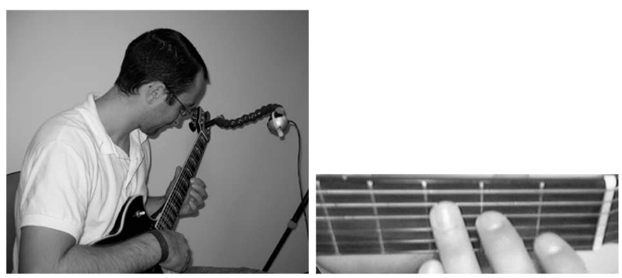
During the preliminary analysis, different camera views were evaluated (global view, front view, and top view). The aim was to find a viewpoint that allows the retrieval of the most information possible with the desired degree of accuracy and precision.

The top view (figure 1(b)) was retained for its interesting characteristics with respect to the problem, namely a detailed view of the fingers, the possibility

(a) Global view with zooms (b) Top view of the left hand

Fig. 1. Two different views of a guitarist playing captured from a camera on a tripod placed in front of the musician: (a) Global view with zoom on different zones for gesture analysis: facial expression and front view of right and left hands. (b) Top view of the left hand.

for string and fret detection, and the ability to observe finger-string proximity. However, slow motion observations of the video recording showed that the neck is subject to many ancillary movements. Preliminary automated tests have shown that this type of movement can influence the computer's capacity to correctly identify fingering. Consequently, the tripod was replaced by a camera mount on the guitar neck (figure 2). The preliminary automated fingering recognition



(a) Camera mount (b) Camera view

Fig. 2. Depiction of the guitar camera mount that was used to eliminate the ancillary gesture problem: (a) The camera mount installed on an electric guitar. (b) The camera on a classical guitar. In this example, the camera is placed to capture the first five frets.

tests were performed by comparing two top view recordings of a musician playing musical excerpts against top view images of previously recorded chords played by the same performer stored in the form of Hu moments vectors [14]. These tests allowed to identify three main issues:

1. Using an appearance based method limits the system to previously learned material.
2. Using the global shape of the hand limits the system to the recognition of chords.
3. Using a knowledge base makes the recognition time grow with the knowledge base size.

From the above issues, the main specifications for a fingering recognition system are:

1. Focus on effective gestures by further reducing the presence of ancillary movements and background elements.
2. Use of a representation that considers the action of individual fingers.
3. Use of a recognition mechanism that eliminates the burden of a knowledge base and that is therefore not limited to previously learned material.

The first specification can be achieved using the guitar mount as presented in figure 2. In order to fulfill the other specifications, three studies were conducted. In a first study, the circular Hough transform was chosen to perform finger localization. The second study examined the use of the linear Hough transform for string and fret detection, and a third one explored movement segmentation.

3 Finger-Localization

The circular Hough transform algorithm used in this paper was developed and implemented in EyesWeb [3]. It presents the following interesting characteristics:

1. It demonstrated to have a high degree of precision and accuracy;
2. It can be applied in complex environments and with partial view of the hand;
3. It can work on edge versions of the images.

3.1 Circular Hough Transform

As illustrated in figure 3, the circular Hough transform [2] is applied on the binary silhouette image of the hand. The edge-image is obtained by applying the Canny edge detection algorithm [5] on the silhouette images. The circular Hough transform algorithm makes use of the fact that finger ends have a quasi-circular shape while the rest of the hand is more linearly shaped. In this algorithm, circles of a given radius are traced on the edge-images and regions with the highest match (many circles intersecting) are assumed to correspond to the center of fingertips.

4 String and Fret Detection

The localization of the fingertips allows to determine where each finger is in space. In the case of guitar fingering, this space can be defined in terms of string and fret coordinates. Prior to the detection stage, the region of interest (in that

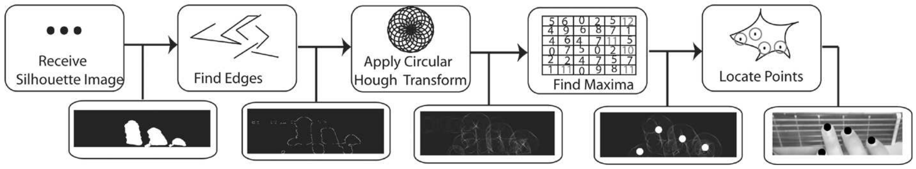


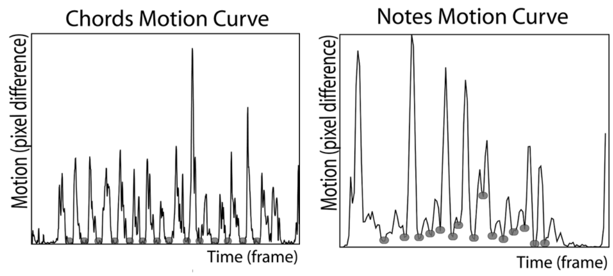
Fig. 3. Fingertip detection using the circular Hough transform algorithm

case the guitar neck) must be located in the image. Once the neck has been located, the strings and frets are segmented from the grayscale neck image by applying a threshold. A vertical and a horizontal Sobel filter are applied on the threshold image to accentuate the vertical and horizontal gradients. A Linear Hough Transform [2] is then computed on the two Sobel images. The linear Hough transform allows detection of linearity in a group of pixels, creating lines. These lines are then grouped by proximity to determine the position of the six strings and of the frets. Once this is done, it is possible to create a grid of coordinates to which fingertip positions will be matched.

5 Movement Segmentation

Movement segmentation is essential to detect fingering positions during significant portions of the playing process (i.e. when notes are played and not during transitions between two notes or chords). Furthermore, in order to save computer resources, this segmentation must be done early in the global process so that subsequent analysis steps are not performed unnecessarily. Movement segmentation is used to separate the nucleus phase of the gesture from the preparation and retraction phase [15].

In the preliminary analysis, movement segmentation was done by applying a threshold on the motion curve (figure 4(a)) generated by the computation of the pixel difference between each frame. The characteristic lower velocity phase of the nucleus was easily detected between each chord. However, in other playing situations, such as when playing a series of notes, the separation between the transition phases and the nucleus is not clear (figure 4(b)). This is due to a phenomenon called *anticipatory placements of action-fingers* that has been studied in violin [1] and piano [9]. In these cases, the preparation phase of other fingers occur during the nucleus of the action-finger. Thus the motion is not serial and consequently, the global motion curve does not exhibit clear global minima like in the case for chords. However, local minima can still be observed and detected and they can be assumed to correspond to the moment the note is trigged by the right hand. Local minima are found by computing the second derivative of the motion curve.



(a) Motion curve of a guitarist playing chords

(b) Motion curve of a guitarist playing notes

Fig. 4. Motion curve of the left hand of a musician playing musical excerpts: (a) Motion curve of a guitarist playing chords (b) Motion curve of a guitarist playing notes

6 Prototype

The prototype was designed to fulfill the requirements for a fingering recognition system highlighted by the preliminary analysis. The focus on effective gestures is partially realized at the hardware level by affixing the camera to the guitar neck, thereby eliminating the motion of the neck caused by ancillary gestures. Elimination of background elements is done by selecting a strict ROI (Region of Interest) around the neck and by applying a background subtraction algorithm on the image. Movement segmentation is performed by finding minima in the motion curve, obtained by computing the pixel difference between each frame. The action of each individual finger is considered using the finger-localization algorithm described above.

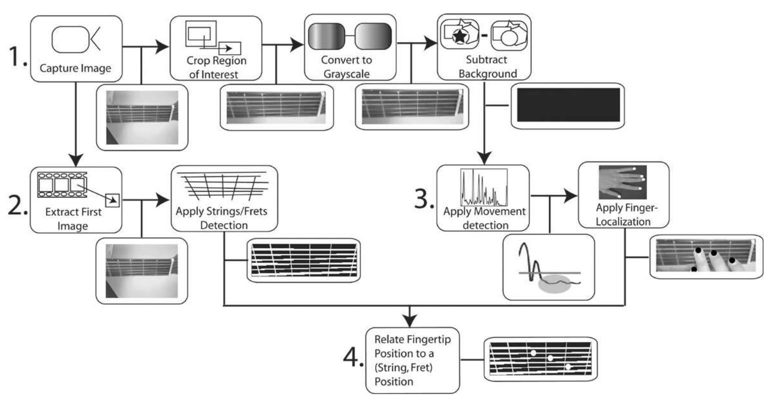


Fig. 5. Prototype - algorithm

7 Test Methods

The prototype was tested on three different types of excerpts: the C major scale, the C major chords progression, and a short excerpt of the melody of Beethoven's *Ode an die Freude*. These excerpts cover the six strings, the three first frets, and are played with the index, middle, and ring fingers. Further tests will be performed in the future to cover the whole camera view fret range and the four fingers. During the test session, the camera was fastened to the neck of a classical guitar. The ROI around the neck for the finger-localization algorithm and for the string and fret detection algorithm was manually selected. The threshold for the finger-localization algorithm and for the string and fret detection components of the string and fret detection algorithm were also manually selected. Finally, the circular Hough transform radius was selected to match the guitarist's fingertip radius. The musician was then asked to play the three chosen excerpts using the fingering displayed on the scores (see [2]). The video images of the playing session were recorded by the camera attached to the neck and by a camera on a tripod in front of the musician. The videos taken with the camera on the guitar mount were then processed in realtime (i.e., without altering the playback speed) in the Eyesweb patch. The videos were also processed manually with the assistance of the musician in order to identify transition phases, note beginnings, and note ends.

8 Results

The system and musician's output were compiled in a table (available on the project website: http://www.music.mcgill.ca/∼amburns/masterproject/). Analysis of the results for the three excerpts was automated in order to compare the musician and the system output. Results were compiled in the following way:

– Fingering positions are defined by the musician for the duration of notes and chords. System output during transition phases is consequently not considered (see table 1 lines 69-70 for an example). It is the movement segmentation algorithm's task to eliminate system output during these phases. The results are compiled using the assumption that this task would have been accomplished successfully. The aim of the compiled results is to evaluate the recognition algorithm only. The movement segmentation algorithm is evaluated separately and will be discussed in he discussion section.
– Fingering positions triplets *(s#: string number x, f#: fret number x, d#: finger number x)* for open string notes and for unplayed strings are left empty by the musician. In these cases, the positions of the fingertips are considered to be undefined. In a real playing situation fingers will probably be placed somewhere over a $(string, fret)$ position in preparation for the next note but this position will not be pressed. The actual prototype can evaluate a fingertip position with respect to the $(string, fret)$ grid but cannot determine if the position is pressed or not. Consequently, the system output is

not considered during open string positions (see table 1 lines 160-161 and 64-65 respectively for examples).

In short, all the fingering positions left empty by the musician were not considered. All the other positions are considered. A match (displayed in bold) can be partial, if for example, the system correctly identifies only the string or fret, or complete, if the $(string, fret, finger)$ triplets of the musician and system output are identical (see table 1 lines 171-172).

			Output			
Frame	Phase	Output type	(s1, f1, d1)	(s2, f2, d2)	(s3, f3, d3)	(s4, f4, d4)
64	D	Musician				
		System	(2, 1, 1)	(2, 2, 2)	(4, 3, 3)	(1, 4, 4)
65	D	Musician				
		System	(2, 1, 1)	(2, 2, 2)	(3, 3, 3)	
69	Transition	Musician				
		System	(3, 1, 1)	(1, 0, 2)	(4, 2, 3)	(4, 3, 4)
70	E	Musician		(**4**, **2**, 2)		
		System	(4, 1, 1)	**4**, **2**, 2)	(4, 3, 3)	(1, 3, 4)
160	G7	Musician	(**1**, **1**, 1)	(**5**, **2**, 2)	(**6**, **3**, 3)	
		System	(**1**, **1**, 1)	(**5**, **2**, 2)	(**6**, **3**, 3)	
161	G7	Musician	(**1**, **1**, 1)	(**5**, **2**, 2)	(**6**, **3**, 3)	
		System	(**1**, **1**, 1)	(**5**, **2**, 2)	(**6**, **3**, 3)	(4, 3, 4)
162	G7	Musician	(**1**, **1**, 1)	(**5**, **2**, 2)	(**6**, **3**, 3)	
		System	(**1**, **1**, 1)	(1, 1, 2)	(**5**, **2**, 3)	(**6**, **3**, 4)
171	G7	Musician	(**1**, **1**, 1)	(**5**, **2**, 2)	(**6**, **3**, 3)	
		System	(**1**, **1**, 1)	(**5**, **2**, 2)	(**6**, **3**, 3)	(4, 3, 4)
172	G7	Musician	(**1**, **1**, 1)	(5, **2**, 2)	(**6**, **3**, 3)	
		System	(**1**, **1**, 1)	(6, **2**, 2)	(**6**, **3**, 3)	(4, 3, 4)

Table 1. (64-65) Example of an open string "fingering". Since this D is played on the open 4th string, all finger positions are undefined and cannot be identified by this prototype. (69-70) Example of the output during a transition to E. On frame 70, E fingering is correctly recognized by the prototype. (160-161) Example of an undefined fingering position. In the G7 chord only the three first finger positions are defined, the little finger does not participate in the chord, consequently its position is undefined and is not considered for a match. Both frames 160 and 161 are perfect matches. (162) Example of the detection of a false fingertip. The system is detecting two fingertips on the first string and fret. This causes the detection of the fifth string, second fret and sixth string, third fret to be shifted to the third and fourth fingers. (171-172) Example of a complete and partial match. On frame 171, the G7 chord is completely recognized while on frame 172 the string of the second finger is not correctly identified.

Table 2 presents the results for each excerpt classified by fret and string. This allows observation of variations in the degree of recognition between the different regions of the guitar neck. A recognition rate *NA* means that this string

	Fret			String						Complete
	1 (%)	2 (%)	3 (%)	1 (%)	2 (%)	3 (%)	4 (%)	5 (%)	6 (%)	fingering (%)
Chords progression	100.0	76.9	56.0	92.7	62.5	24.4	27.3	24.2	84.2	**14.0**
Scale	100.0	79.0	63.8	NA	100.0	87.5	64.7	19.2	NA	**34.2**
Ode An die Freude	100.0	85.4	75.9	NA	79.2	70.3	24.3	33.3	NA	**51.6**
Total	**100.0**	**80.3**	**67.4**	**92.7**	**66.7**	**59.2**	**28.9**	**22.5**	**84.2**	**40.2**
	76.2			**48.2**						

Table 2. Recognition rate per string and fret

or fret was not played in this musical excerpt. The last line outlines the total recognition rate for each fret and string and the total recognition rate for all frets and all strings. Finally, the complete fingering column presents the recognition rate of a complete fingering (all the $(string, fret, finger)$ triplets composing a chord or a note).

9 Discussion

9.1 Missed or Extra Fingertips Detection

More than eighty percent of the errors are caused by the detection of false fingertips or by the non-detection of a fingertip. The first type of error is caused by phenomena such as string vibration, shadow or reflection due to lighting variation, and variation in the guitar neck color, due, for instance, to aging of the instrument. These phenomena create noise that can be misinterpret as a circular shape by the Hough transform. These errors are difficult to solve but in some cases – like the one illustrated in table 1 line 162– they could be solved by applying rules like forbidding the detection of two fingers on the same $(string, fret)$ position. Problems due to lighting variations could potentially be solved using an infrared camera together with a ring of infrared LEDs providing a constant lighting. The second type of errors is mostly due to partial or total occlusion of a finger by another finger or by deformation of the quasi-circular shape of the fingertips due to the camera angle. These errors also cause the $(string, fret)$ position to be attributed to the wrong finger and in the worst case – for instance, when two fingers play the same fret on two different strings– a fingering position will completely be omitted. These errors are difficult to solve since they require more than small modifications of the setup. They could potentially be solved algorithmically by estimating fingertip trajectories from the previous non-occluded images or by locating the non-occluded part of the finger and estimating the tip position from it.

9.2 Strings Spacing

Due to the placement of the camera, the space between the strings is smaller for the upper strings (E, A, D) than for the lower strings (G, B, E), affecting the accuracy of the recognition system. The angle of the camera also affects the quasi-circular shape of the fingertips making these appear flatter and consequently more likely to be missed by the Hough transform as explained previously. In fact, it is possible to observe a decrease in the recognition rate from string 1 (high pitch) to string 5 (low pitch). The sixth string seems to be recognized better. This might be due to the fact that it is the last string, consequently fingertips that are found above it will also be quantized to the sixth string. As explained in [3, 2], the circular Hough transform has an accuracy of 5 +/- 2 pixels with respect to color marker references placed at the center of the fingertip. The resolution of the camera used in this prototype is 640x480 pixels, giving a 610x170 pixels neck region in our tests. The distance between the E (low pitch) and A strings is of 12 pixels at the first fret and 17 at the fifth fret. Between the B and E (high pitch) strings, the distance is 16 and 20 pixels for the first and fifth fret, respectively. In the worst case the finger-localization algorithm error exceeds half the space between the upper strings and the fingertip center is detected above the string, resulting in the fingertip being quantized one string above its real position. However, since this problem happens less frequently with high-pitched strings, where the distance between two strings is larger, the problem could have been solved using a higher-resolution camera. The higher recognition rate for the fret positions where the space between two frets is much larger also tends to confirm this hypothesis.

9.3 Guitar Neck Image Deformation

From table 2 it can also be observed that there is a small decrease in the fret recognition rate from left to right. This problem might be due to the camera angle that creates a deformation of the neck image and of the fingertips' shapes or to the angle at which the musician attacks the different frets. The neck image deformation or some attack angles can cause the fingertip center to appear slightly to the right of the fret. The chosen quantization method will therefore quantize the fingertip to the neck fret position. This problem could potentially be solved by applying a perspective correction algorithm to straighten the image. Perspective correction might also help to reduce the ''missing fingertips" type of error.

9.4 Movement Segmentation Error

The method of thresholding the motion curve presented in section 5 works for chords, and the assumption was that it would have been possible to detect minima in the motion curve of sequences of notes, but this assumption failed. It is either because the assumption is wrong, and consequently it might not be possible to rely on the left-hand image only for movement segmentation, or because the motion curve would need further high-pass filtering to remove small variations that cause minima unrelated to the note nucleus and generate false segmentation. The second hypothesis is the preferred one since it was possible to located minima at the note nucleus by visual inspection of the motion curve as seen in figure 4(b). Further tests are required to draw a definitive conclusion on this matter.

10 Conclusion

This prototype meets most of the requirements for a realtime guitarist fingering retrieval system, namely:

- The system outputs the musician's solution and consequently accounts for all aspects of the fingering choice.
- The system does not require any preliminary information or analyses of the musical excerpt, it reads the fingering solution directly from the musician execution of the excerpt.
- The system is non-obtrusive, the musician does not need to adapt his playing style or to wear special devices. Only the weight of the guitar mount can be disturbing but this could be solved by using a lighter camera-mount setup.
- The system is composed of a regular webcam on a mount and is easy to affix to the guitar. The software requires only a few manual settings that could be possible to automate in the future version. The system is therefore accessible in terms of cost and ease of use. However, further testing is still required to conclude on the reproducibility of the results for a variety of guitars.

Although the recognition rate for chords is low, this algorithm demonstrated the potential for the use of computer vision to solve the fingering problem. In fact, by detecting individual fingers, it is possible to obtain partial fingering information. For instance two notes of a three note chord are solved, or ($string$, $fret$) coordinates are correctly recognized but are attributed to the wrong finger. In some cases, it is possible that this partial information could be used to make an educated guess on the complete fingering. Also, as the discussion section highlighted, many of the problems could be solved by small modifications of the hardware and software settings.

Acknowledgments

The study on finger-localization was realized at InfoMus laboratory, D.I.S.T., Universitá degli studi di Genova and has been partially supported by funding from the Québec Government (PBCSE), the Italian Ministry of Foreign Affairs, and by the EU 6 FP IST ENACTIVE Network of Excellence. Anne-Marie Burns would like to thank Antonio Camurri for welcoming her as an internship student researcher.

The authors would like to thank the guitarists who participated in the tests: Jean-Marc Juneau, Riccardo Casazza, and Bertrand Scherrer. Thanks also to Mr. Donald Pavlasek, for the conception and implementation of the camera guitar mount.

References

1. A. P. Baader, O. Kazennikov, and M. Wiesendanger. Coordination of bowing and fingering in violin playing. *Cognitive Brain Research*, 23:436–443, 2005.
2. A.-M. Burns. *Computer Vision Methods for Guitarist Left-Hand Fingering Recognition.* Master thesis, McGill University, Montréal, Québec, Canada, 2006.
3. A.-M. Burns and B. Mazzarino. Finger tracking methods using EyesWeb. In S. Gibet, N. Courty, and J.-F. Kamp, editors, *Gesture Workshop 2005 Proceedings*, volume LNAI 3881, pages 156–167, 2006.
4. G. Cabral, I. Zanforlin, R. Lima, H. Santana, and G. Ramalho. Playing along with d'Accord guitar. In *Proceedings of the 8th Brazilian Symposium on Computer Music*, 2001.
5. J. A. Canny. Computational approach to edge detection. *IEEE Transaction on Pattern Analysis and Machine Intelligence*, 8(6):679–698, 1986.
6. G. Cuzzucoli and V. Lombardo. A physical model of the classical guitar, including the player's touch. *Computer Music Journal*, 23(2):52–69, 1999.
7. R. Doati. My Eyesweb experience. In *EyesWeb Week 2006*, 2006.
8. G. Elkoura and K. Singh. Handrix: Animating the human hand. In D. Breen and M. Lin, editors, *Proceedings of eurographics/SIGGRAPH Symposium on Computer Animation*, 2003.
9. K. C. Engel, M. Flanders, and J. F. Soechting. Anitcipatory and sequential motor control in piano playing. *Experimental Brain Research*, 113:189–199, 1997.
10. A. Gilardino. Il problema della diteggiatura nelle musiche per chitarra. *Il "Fronimo"*, 10:5–12, 1975.
11. A. Gilardino. Il problema della diteggiatura nelle musiche per chitarra. *Il "Fronimo"*, 13:11–14, 1975.
12. M. Laurson, C. Erkut, V. Välimäki, and M. Kuushankare. Methods for modeling realistic playing in acoustic guitar synthesis. *Computer Music journal*, 25(3):38–49, 2001.
13. M. Miura, I. Hirota, N. Hama, and M. Yanagida. Constructing a system for finger-position determination and tablature generation for playing melodies on guitars. *Systems and Computers in Japan*, 35(6):10–19, 2004.
14. S. Paschalakis and P. Lee. Pattern recognition in grey level images using moment based invariant features image processing and its applications. *IEE Conference Publication*, 465:245–249, 1999.
15. V. I. Pavlovic, R. Sharma, and T. S. Huang. Visual interpretation of hand gestures for human-computer interaction: A review. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 19(7):677–695, 1997.
16. D. Radicioni, L. Anselma, and V. Lombardo. A segmentation-based prototype to compute string instruments fingering. In *Proceedings of the Conference on Interdisciplinary Musicology*, Graz, 2004.
17. C. Truchet. *Contraintes, recherche locale et composition assistée par ordinateur.* PhD thesis, Université Paris 7 - Denis Diderot, UFR Informatique, 2004.
18. J. A. Verner. MIDI guitar synthesis yesterday, today and tomorrow. an overview of the whole fingerpicking thing. *Recording Magazine*, 8(9):52–57, 1995.
19. J.-F. Wang and T.-Y. Li. Generating guitar scores from a MIDI source. In *Proceedings of 1997 International Symposium on Multimedia Information Processing*, 1997.