Finger Tracking Methods Using EyesWeb

Anne-Marie Burns[1] and Barbara Mazzarino[2]

[1] Input Devices and Music Interaction Lab, Schulich School of Music,
McGill University, 555 Sherbrooke Street West,
Montréal, Québec, Canada, H3A 1E3
anne-marie.burns@mail.mcgill.ca
http://www.music.mcgill.ca/~amburns/
[2] InfoMus Lab, DIST - University of Genova,
Viale Causa 13, I-16145, Genova, Italy
barbara.mazzarino@unige.it
http://infomus.dist.unige.it

Abstract. This paper compares different algorithms for tracking the position of fingers in a two-dimensional environment. Four algorithms have been implemented in EyesWeb, developed by DIST-InfoMus laboratory. The three first algorithms use projection signatures, the circular Hough transform, and geometric properties, and rely only on hand characteristics to locate the finger. The fourth algorithm uses color markers and is employed as a reference system for the other three. All the algorithms have been evaluated using two-dimensional video images of a hand performing different finger movements on a flat surface. Results about the accuracy, precision, latency and computer resource usage of the different algorithms are provided. Applications of this research include human-computer interaction systems based on hand gesture, sign language recognition, hand posture recognition, and gestural control of music.

1 Introduction

The advances in technology and the widespread usage of computers in almost every field of human activity are necessitating new interaction methods between humans and machines. The traditional keyboard and mouse combination has proved its usefulness but also, and in a more extensive way, its weakness and limitations. In order to interact in an efficient and expressive way with the computer, humans need to be able to communicate with machines in a manner more similar to human-human communication.

In fact, throughout their evolution, human beings have used their hands, alone or with the support of other means and senses, to communicate with others, to receive feedback from the environment, and to manipulate things. It therefore seems important that technology makes it possible to interact with machines using some of these traditional skills.

The human-computer interaction (HCI) community has invented various tools to exploit humans' gestures, the first attempts resulting in mechanical devices.

S. Gibet, N. Courty, and J.-F. Kamp (Eds.): GW 2005, LNAI 3881, pp. 156–167, 2006.

Devices such as data gloves can prove especially interesting and useful in certain specific applications but have the disadvantage of often being onerous, complex to use, and somewhat obtrusive.

The use of computer vision can consequently be a possible alternative. Recent advances in computer vision techniques and availability of fast computing have made the real-time requirements for HCI feasible. Consequently, extensive research has been done in the field of computer vision to identify hand poses and static gestures, and also, more recently, to interpret the dynamic meaning of gestures [6][9]. Computer vision systems are less intrusive and impose lower constraints on the user since they use video cameras to capture movements and rely on software applications to perform the analysis.

In order to avoid the problem of complex and not reproducible high cost systems, this paper focuses on two-dimensional systems using a single simple video camera. Algorithms using projection signatures, the circular Hough transform, and geometric properties have been chosen and are compared to an algorithm using color markers. Color markers are used solely as a reference system to evaluate the accuracy and the precision of the other algorithms, the presence of markers being a non-desirable constraint on the user of such a system. All the algorithms have been implemented in EyesWeb using the Expressive Gesture Processing Library [1] together with newly developed blocks (available in EyesWeb 4). These algorithms are designed to track different joints of the hand and more particularly of the finger (finger intersections, fingertips). Knowledge about these points on a frame-by-frame basis can later be provided to other analysis algorithms that will use the information to identify hand poses (static) or hand gestures (dynamic). Finger tracking is therefore at the base of many HCI applications and it opens new possibilities for multimodal interfaces and gestural control of music.

The algorithms presented in this paper are inspired by the research on tabletop applications [7][8]. These kinds of applications are often limited to the use of one finger instead of using the information that can be provide by tracking all fingers. Furthermore, these applications often use specific and expensive hardware (infrared camera for example). In this paper we suggest alternative methods that can work with simple hardware, such as a low-cost webcam. We use methods that were traditionally used in static pose identification (e.g. contour, signature) to do dynamic tracking. The use of the Hough transform, on the other hand, was inspired by research in 3-dimensional tracking [4], but also by some of the previously mentioned tabletop applications. These applications use the specific geometric shape of the fingertip with various templates matching algorithms to locate fingers.

The first section of this article briefly describes and illustrates the EyesWeb implementation of the four algorithms. Next, the test procedures are explained. The third section presents the results obtained from each algorithm during the tests. Finally, the article concludes with a comparative discussion of the potential uses of the different algorithms.

2 Methods

All the algorithms were evaluated using two-dimensional images of a hand performing different finger movements on a flat surface. The videos were recorded by a single fixed camera with a frame rate of 25fps (frame per second), fixed gain and fixed shutter. The tests were run on a Pentium 4 3.06GHz with 1Gb of RAM under Windows XP operating system. In order to test the algorithms, the problems of finding the region of interest and of eliminating complex backgrounds were reduced by shooting only the hand region on a uniform dark background. The second line of figures 1, 2, and 3 illustrates the segmentation process. In this simplify case, it consists of converting the image to gray-scale, applying a threshold to segment the hand from the background (using the fact that the hand is light while the background is dark), and filtering with a median filter to reduce residual noise.

2.1 Projection Signatures

Projection signatures, are performed directly on the resulting threshold binary image of the hand. The core process of this algorithm is shown on line 3 of figure 1 and consists of adding the binary pixels row by row along a diagonal (the vertical in this case). Previous knowledge of the hand angle is therefore required. A low-pass filter is applied on the signature (row sums) in order to reduce low frequency variations that create many local maxima and cause the problem of multiple positives (more than one detection per fingertip). The five maxima thereby obtained correspond to the position of the five fingers.

2.2 Geometric Properties

The second algorithm is based on the geometric properties and, as shown on line 3 of figure 2, uses a contour image of the hand on which a reference point is set. This point can be determined either by finding the center of mass of the contour (barycenter or centroid) or by fixing a point on the wrist [11]. Euclidean distances from that point to every contour points are then computed, with the five resulting maxima assumed to correspond to the finger ends. The minima can be used to determine the intersections between fingers (finger valleys). The geometric algorithm also required filtering in order to reduce the problem of multiple positives.

2.3 Circular Hough Transform

The circular Hough transform is applied on the contour image of the hand but could as well be performed on an edge image with complex background if no elements of the image exhibit the circular shape of the fingertip radius. The circular Hough transform algorithm uses the fact that the finger ends and the

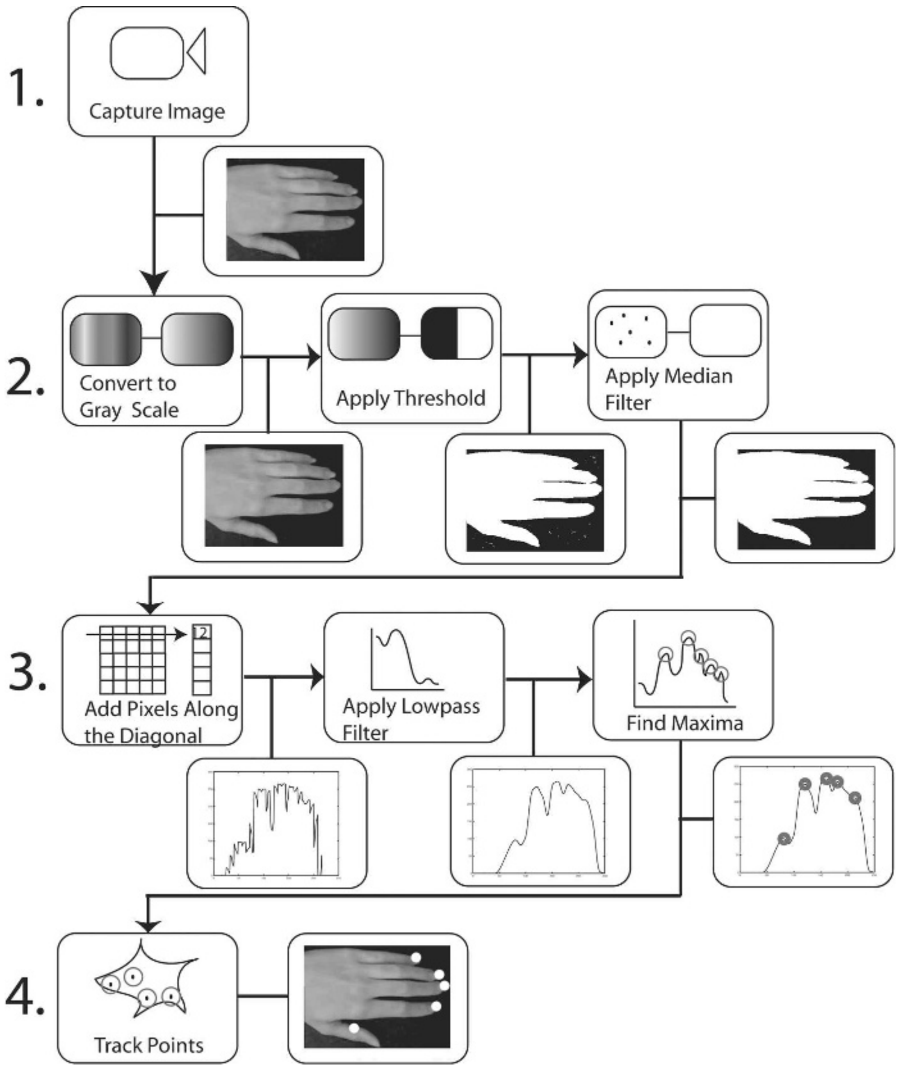

Fig. 1. Processing steps of the column signature algorithm

finger valleys 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 or contour image and regions with the highest match (many circles intersecting) are assumed to correspond to finger ends and valleys (this process is illustrated on line 3 of figure 3). Searched fingertips radius can be set manually or determined by an algorithm using the palm radius to fingertip radius proportion as an estimate [2] [11] [4]. The circular Hough transform can find both finger ends and valleys but, as opposed to the geometric algorithm, doesn't output them in

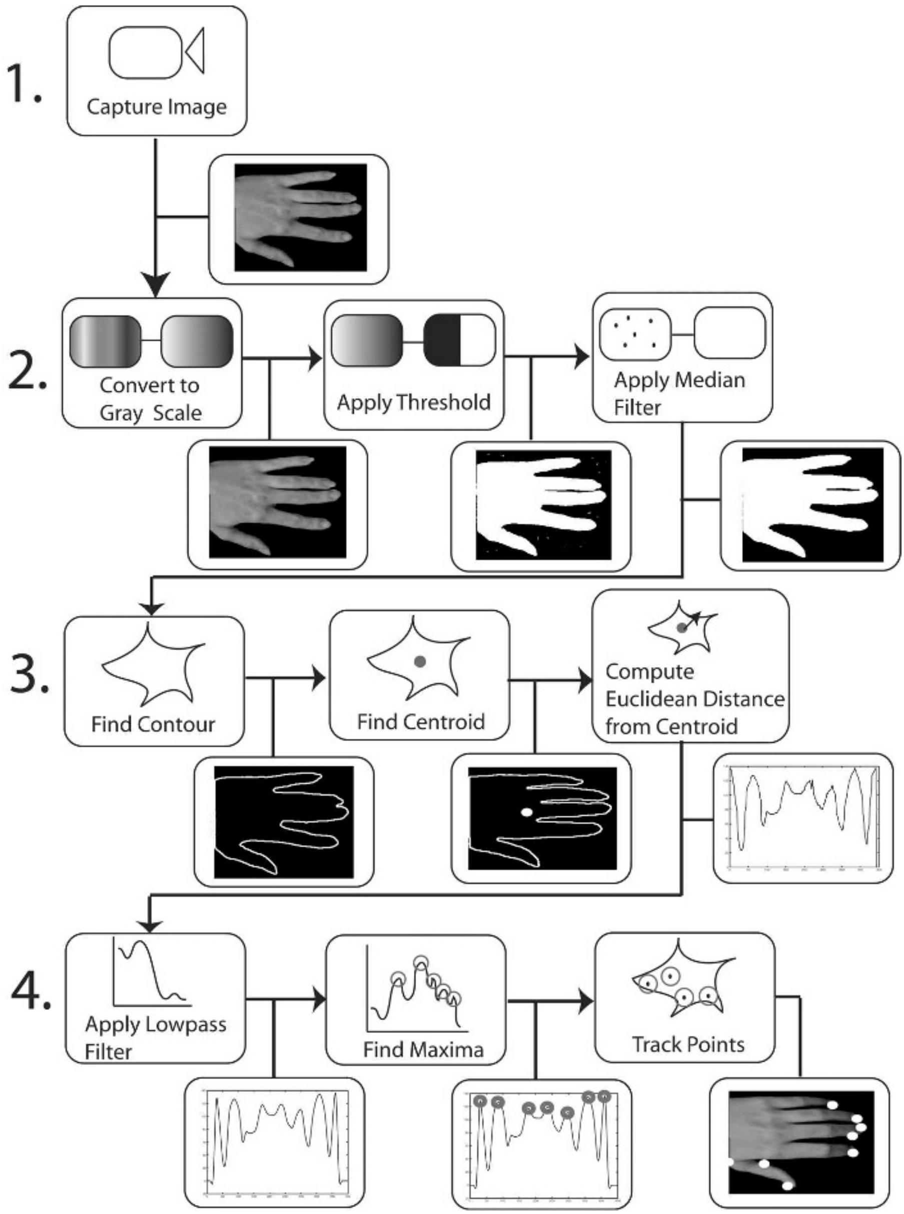

Fig. 2. Processing steps of geometric properties method

two distinct sets. Furthermore, the circular Hough transform requires filtering to eliminate false positives (detected regions that are not finger ends or valleys) that frequently appeared between fingers. As illustrated in line 4 of figure 3,

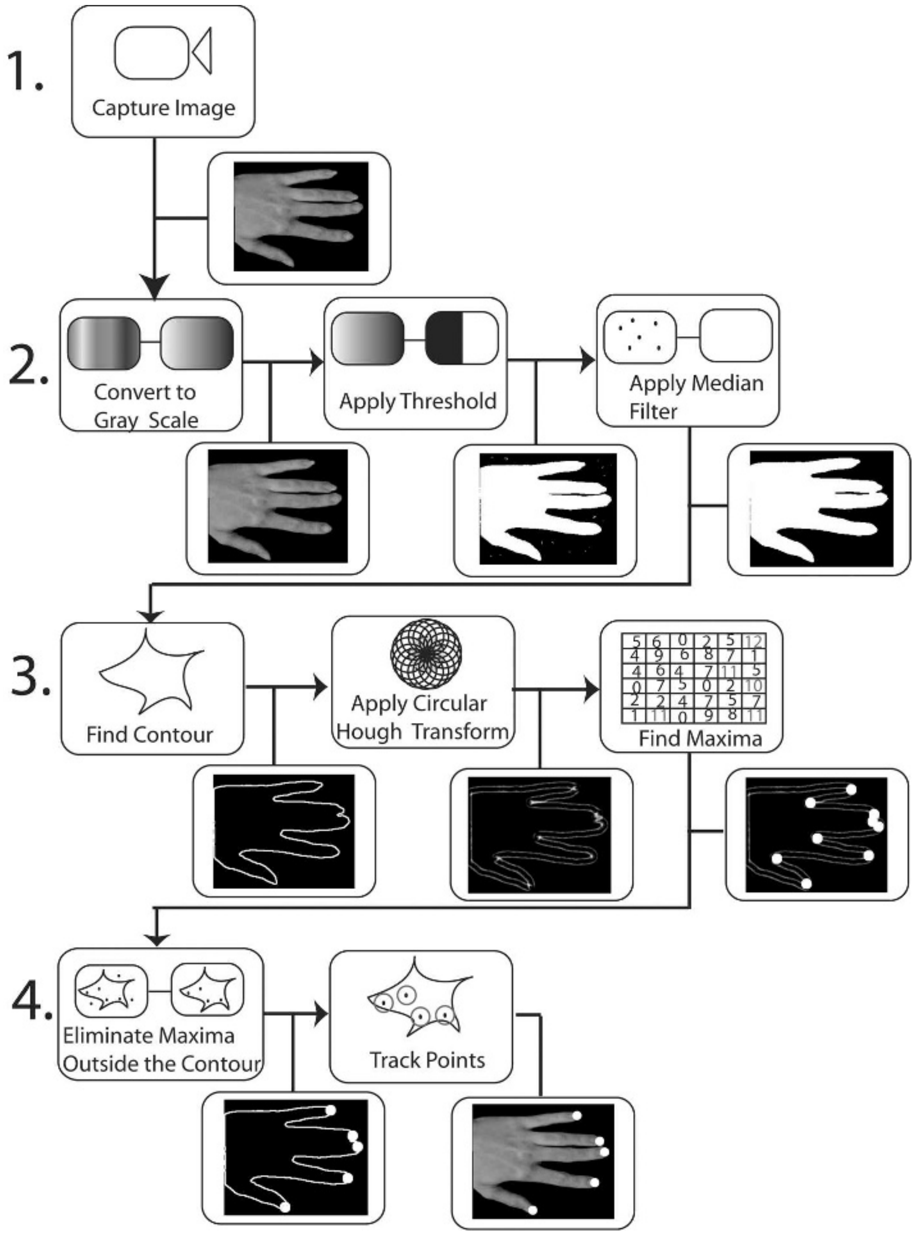

Fig. 3. Processing steps of circular Hough transform method

this can be done efficiently for finger ends by eliminating points that are found outside the contour image. The inconvenient is that the set of discard points contains a mix of finger valleys and false positive that cannot be sorted easily.

2.4 Color Markers

While the three previous algorithms rely only on the hand characteristics to find and track the fingers, the marker algorithm tracks color markers attached to the main joints of the fingers. Each color is tracked individually using color segmentation and filtering as illustrated in line 2 of figure 4. This permits the identification of the different hand segments. The marker colors should therefore be easy to track and should not affect the threshold, edge or contour image of the hand. Respecting these constraints makes it possible to apply all algorithms to the same video images and therefore to compare each algorithm degree of accuracy and precision with respect to the markers.

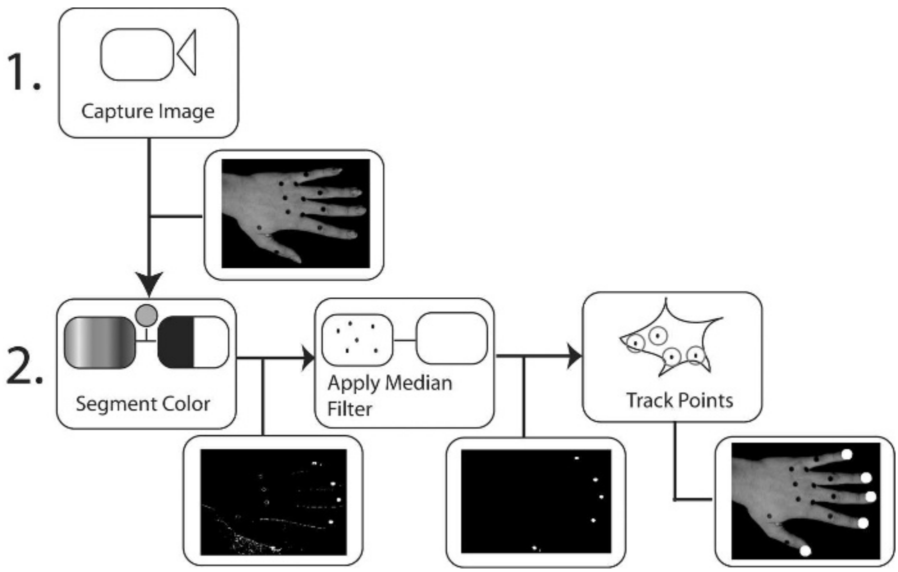

Fig. 4. Processing steps of color markers method

3 Tests

3.1 Accuracy and Precision

Accuracy and precision are important factors in the choice of a finger-tracking algorithm. The accuracy and precision of the different algorithms were determined with respect to the result obtained from the evaluation of the marker positions. To evaluate the accuracy and precision of the algorithms, the coordinates of 4 joints on each finger were tracked by applying the color tracking method (figure 4). Coordinates obtained with the three other algorithms were then related to the first set. The Euclidean distance between the marker and the closest point of each algorithm was computed. The accuracy of an algorithm

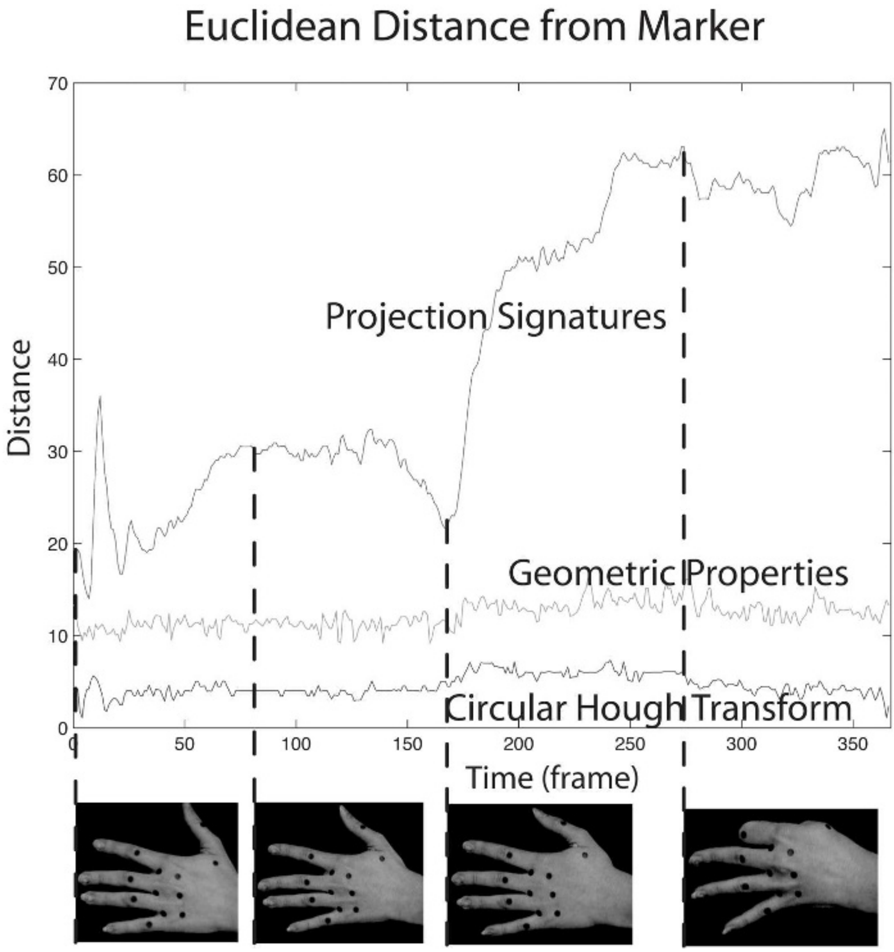

Fig. 5. Euclidean distance from the color marker for each of the three methods

can be determined by its distance from the marker. A curve close to zero denotes an accurate algorithm. The precision of an algorithm can be determined by observing the shape of the curve. A precise algorithm will exhibit an almost flat curve. Figure 5 presents the results obtained by tracking the tip of the small finger using each of the three algorithms. The values are compared to a marker placed at the center of the tip of the small finger. It can be observed that both the circular Hough transform and the geometric properties algorithm are precise algorithm since the distance between the marker and the point they return is almost constant. However, the circular Hough transform seems to be more accurate than the geometric properties. The average distance to the marker is really close to zero in the case of the circular Hough transform, but is approximately ten pixels in the case of the geometric properties.

The difference is mainly due to the fact that the geometric properties algorithm detects the extremity of the finger while the circular Hough transform finds the center and that where the markers are placed. In the case of the projection signatures, the detection of the fingers is robust but rough: the algorithm can only find the fingers and not a specific region of the finger like a tip or a valley. It can be observed in figure 5 that for an almost flat angle of the small finger, the accuracy is near twenty pixels (frame 0 and 160), for a small angle (between frame 50 and 160) it is approximately thirty pixels, and can go over a difference of sixty pixels for a large angle (after frame 160). This is due to the computation method, when the finger is angle, the end of the section that is in straight line with the palm will create a maximum and not the real finger end. This algorithm is consequently efficient only to find fingers or finger ends when the fingers are not angled. This algorithm is therefore neither accurate nor precise.

3.2 Latency and Resources Usage

The latency of each of the algorithms is determined by computing the delay between the evaluation of a frame and the output of its results. If the output rate is the same as the input rate (expressed in terms of the amount of time lapse between two input frames), no significant delay is generated by the evaluation part of the algorithm. In order to know the processing rate and the resource usage of the evaluation algorithm, all screen or file outputs were turned off. Table 1 displays the CPU (central processing unit) usage for each algorithm. The range is the observed minimum and maximum CPU usage percent throughout the duration of the test. The mode is the most frequently observed percentage. Table 1 shows that all the algorithms can be used in real time since no significant

Table 1. CPU usage of the three methods

Input Rate	**Algorithms**	**CPU Usage Range**	**CPU Usage Mode**	**Output Rate**
33 ms	Projection Signatures	10-18%	15%	33 ms
	Circular Hough Transform	38-77%	55%	
	Geometric Properties	16-45%	30%	

latency as been observed. Projection signature is extremely easy on computer resource with a mode of 15% of CPU usage and peaks ranging between 10 and 18%. Geometric properties is a bit more demanding with a mode of 30%. The poor performance of the circular Hough transform is probably due to the usage of the traditional algorithm [3] [10] that requires a lot of computation and storage for the accumulator cells, more modern implementations using probabilistic and heuristic approaches to optimize the algorithm performance exist [5] and are known to detect circles with the same degree of accuracy and precision.

4 Results and Discussion

We tested the previously presented algorithms with video recordings of the left and right hand of 5 users (3 females and 2 males, all adults). Results of these preliminary tests were coherent among all users and are qualitatively summarized in Table 2.

Table 2. Algorithms characteristics (+ → good to excellent, 0 → neutral to good, - → poor to neutral

	Projection Signatures	**Geometric Properties**	**Circular Hough Transform**	**Color Markers**
Locates fingers	+	+	+	+
Locates fingertips	-	0	0	+
Locates finger ends and valleys	-	+	+	+
Distinguishes between finger ends and valleys	-	+	0	+
Works with complex background	-	-	0	0
Works in real time (low latency)	+	+	+	+
Computer resources usage	+	+	-	+
Accuracy	-	+	+	+
Precision	-	+	+	+
Works with unknown hand orientation	-	+	+	+
Works with unknown fingertips radius	+	+	0	+

All the presented algorithms have succeeded, in various degrees, in detecting each finger. The projection signatures algorithm can only roughly identify a finger, but the circular Hough transform and geometric properties algorithms can find both finger intersections and finger end points, it is important to note that in the case where finger are folded, the end points dont correspond to the fingertips. The geometric properties algorithm outputs intersections and extremities in two distinct sets, but the circular Hough transform algorithm cannot make this distinction. The marker algorithm is the only one that can distinguish the various joints of the finger when different colors are used.

The projection signatures and geometric properties algorithms need a strong segmentation step prior to their application. The circular Hough transform, when combined with edge detection instead of contour, can work in complex environments, but some confusion can occur if other circular shapes of the size of the fingertip radius are present. Color markers can be used in complex backgrounds if the colors are properly chosen but are sensitive to light variation.

At 25fps all the algorithms output results without any significant delay; the input and output rate is the same. However, the circular Hough transform algorithm is much more demanding on CPU usage. This characteristic might limit its use when it is combined with pose and gesture recognition algorithms. The geometric properties and the circular Hough transform algorithms have similar and acceptable accuracy and precision values. The projection signatures algorithm cannot be used if these two characteristics are important.

The projection signatures algorithm can only be used in a controlled environment where the hand orientation is known and where finger angles dont vary to much from the straight line. The circular Hough transform algorithm needs previous knowledge of the fingertip radius or the palm radius. It can work in an environment where the distance from the video camera will change only if a method to estimate these radii is attached to it [2]. The geometric properties algorithm does not need any prior knowledge to be performed.

5 Conclusion

This article presented three algorithms to track fingers in two-dimensional video images. These algorithms have been compared to one another and evaluated with respect to a fourth algorithm that uses color markers to track the fingers. All the algorithms were implemented and tested in EyesWeb. Results relative to the precision, accuracy, latency and computer resource usage of each of the algorithms showed that geometric properties and circular Hough transform are the two algorithms with the more potential. The circular Hough transform should be preferred when a clean segmentation from the background is impossible while the geometric properties algorithm should be use when the fingertips radius is unknown and when information on both the finger ends and valley is required. Projection signature can be used as a fast algorithm to roughly obtained finger position. The choice of an algorithm should, therefore, depend on the application and on the setup environment. Future users should refer to the algorithms characteristics and constraints in table 2 to chose the appropriate one. It is also important to note that in this paper, the algorithms were tested alone and in a controlled environment. Consequently, the choice of an algorithm can also be influenced by the system in which it is supposed to work. As an example, the segmentation algorithm used in the pre-processing step and the pose or gesture algorithm used in the post-processing step can create constraints that will dictate the usage of a finger-tracking algorithm.

Acknowledgments

This work has been partially supported by funding from the Qubec Government (PBCSE) and the Italian Ministry of Foreign Affairs to Anne-Marie Burns and by the EU 6 FP IST ENACTIVE Network of Excellence to both authors. The authors would like to thank all students and employees of InfoMus lab who "gave their hands" for the realization of the tests. Special thanks go to Ginevra Castellano for her help in compiling the results, Gualtiero Volpe for his contribution to the development of the EyesWeb blocks, Pascal Bélanger for English proofreading, and Marcelo Wanderley for proofreading and constructively commenting on this article. Anne-Marie Burns would also like to thank Antonio Camurri for welcoming her as an internship student researcher and Marcelo Wanderley for initiating and making this collaboration project possible.

References

1. Camurri M., Mazzarino B., and Volpe G.: Analysis of Expressive Gesture: The EyesWeb Expressive Gesture processing Library, in Gesture-based Communication in Human-Computer Interaction. LNAI 2915, Springer Verlag (2004) 460-467
2. Chan, S. C.: Hand Gesture Recognition, [http://www.cim.mcgill.ca/~schan19/research/research.html], Center for Intelligent Machines, McGill University (2004).
3. Duda, S. R. D. and Hart, P. E.: Use of the Hough Transform to Detect Lines and Curves in Pictures in Communications of the Association of Computing Machinery, 15, 11-15, (1972).
4. Hemmi, K.: On the Detecting Method of Fingertip Positions Using the Circular Hough Transform in Proceeding of the 5th Asia-Pacific Conference on Control and Measurement, (2002).
5. Illingworth, J., and Kittler, J.: A Survey of the Hough Transform in Computer Vision, Graphics, and Image Processing, 44, 87-116, (1988).
6. Kohler M.:Vision Based Hand Gesture Recognition Systems, [http://ls7-www.cs.uni-dortmund.de/research/gesture/vbgr-table.html], Computer Graphics, University of Dortmund.
7. Koike, H., Sato, Y., and Kobayashi, Y.: Integrating Paper and Digital Information on EnhancedDesk: A Method for Realtime Finger Tracking on an Augmented Desk System, ACM Transaction on Computer-Human Interaction, vol. 8, no. 4, 307-322, (2001).
8. Letessier, J., and Brard F.: Visual Tracking of Bare Fingers for Interactive Surfaces, Seventeenth Annual ACM Symposium on User Interface Software and Technology, vol. 6, issue 2, 119-122, (2004).
9. Pavlovic V. I., Sharma R., and Huang T. S.: Visual Interpretation of Hand Gestures for Human-Computer Interaction: A Review, IEEE Transactions on pattern analysis and machine intelligence, vol. 19, no. 7, 677-695, (1997).
10. Schulze, M. A.: Cicular Hough Transform A Java Applet Demonstration, [http://www.markschulze.net/java/hough/], (2003).
11. Yörük E., Dutağaci H., and Sankur B.: Hand Biometrics, Electrical and Electronic Engineering Department, Boğaziçi University (to appear).