

11-22-2021

Analysis of Human Arm Movements Using Computer Vision System.

I. El-Nahry

Department of Electrical & Computer Engineering, Suez Canal University, Port-Said, Egypt.

H. El-Hendy

Department of Electrical & Computer Engineering, Suez Canal University, Port-Said, Egypt.

Follow this and additional works at: <https://mej.researchcommons.org/home>

Recommended Citation

El-Nahry, I. and El-Hendy, H. (2021) "Analysis of Human Arm Movements Using Computer Vision System.," *Mansoura Engineering Journal*: Vol. 29 : Iss. 3 , Article 11.

Available at: <https://doi.org/10.21608/bfemu.2021.206043>

This Original Study is brought to you for free and open access by Mansoura Engineering Journal. It has been accepted for inclusion in Mansoura Engineering Journal by an authorized editor of Mansoura Engineering Journal. For more information, please contact mej@mans.edu.eg.

ANALYSIS OF HUMAN ARM MOVEMENTS USING COMPUTER VISION SYSTEM

تحليل حركات ذراع الإنسان باستخدام نظام الرؤية بالحاسب

Dr. I.F. El-Nahry

Dr. H.M. El-Hendy

Department of Electrical & Computer Eng., Suez
Canal University, Port-Said, Egypt

الملخص

التحليل البصري لحركة الإنسان حالياً واحد من أهم المواضيع البحثية النشطة في مجال الرؤية بالكمبيوتر. هذا الاهتمام القوي يقوده مدى واسع من التطبيقات الواعدة في عدة مناطق مثل الواقعية التخيلية، التحليل البيوميكانيكي، المراقبة الذكية والواجهة الإدراكية الحسية. تحليل حركة الإنسان تهتم باكتشاف وتتبع وإدراك وفهم سلوكيات الإنسان من تتابع الصور التي تشمل الإنسان. في هذا البحث تم وصف خوارزم لتحليل حركة ذراع الإنسان من تتابع فيديو باستخدام الصورة الرفيعة. تجيء طرافة هذا الخوارزم من (i) تحديد ذراع الإنسان أوتوماتيكياً (ii) إزالة السمات المشوهة (iii) ترفيع وانتزاع سمات الصورة. تلميحات الخوارزم المقترح تستخدم لتقرير نشاطات ذراع الإنسان مثل سرعة الكتف و الكوع و الرسغ واليد. وعلى خلاف الخوارزميات الأخرى فهذا الخوارزم لا يتطلب نموذج استنتاجي للإنسان أو عدد كبير من النقاط للهدف (ذراع الإنسان). علاوة على انه رخيص بشكل حسابي مثالي لتطبيقات الفيديو مثل التحليل البيوميكانيكي.

Abstract

Visual analysis of human arm motion is currently one of the most active research topics in computer vision. This strong interest is driven by a wide spectrum of promising applications in many areas such as virtual reality, biomechanical analysis, smart surveillance and perceptual- interface. Human arm motion analysis concerns the detection, tracking, recognition and understanding of human behaviors, from image sequences involving human. In this paper, a process is described an algorithm for analyzing the arm motion of a human target in a video sequences by image skeletonization. The novelty of this algorithm comes from (i) automatic arm detection, (ii) removing spurious features, (iii) robust of skeleton and features extraction. The cues of the proposed algorithm are used to determine human arm activities such as velocity of shoulder, elbow, wrist and hand. Unlike other algorithms, this does not require a priori human model, or a large number of "pixels on target". Furthermore, it is computationally inexpensive and thus ideal for video applications such as biomechanical analysis.

Keywords: Computer Vision, Motion Segmentation, Image skeletonization, Feature extraction, Motion Analysis.

1- Introduction

Using video in machine understanding has recently become a

significant research topic. Human arm motion analysis has attracted great

interests from computer vision researches due to its promising applications in many areas such as robot learning, surveillance, biomechanical analysis, virtual reality, etc.

Recently, there have been several popular approaches to the recognition of human motion with much emphasis on real time computation [1-5]. Several survey papers review vision-based motion recognition, human motion capture and human motion analysis [6,7]. One of the most recent research areas is activity understanding from video imagery [8]. Understanding activities involves being able to detect and classify targets of interest and analyze what they are doing. There have been several good human detection schemes, such as [9] which use static imagery. Detecting and analyzing human motion in real time from video imagery have recently become feasible with algorithms like [10] and [11].

There are two main drawbacks of these systems in their present forms: they are completely human specific and they require a great deal of image-based information in order to work effectively. We propose a human arm motion algorithm that is able to analyse the motion from small amount of image data using skeletonization procedure. Once a skeleton is extracted motion analysis can be determined from it. Section 2 describes the human arm movements algorithm. Section 3 presents the detection results of this algorithm on several human arm databases. Conclusions are described in Section 4.

2. Human arm movements algorithms

An overview of the proposed arm movements algorithm is depicted in figure 1, which contains five operating stages.

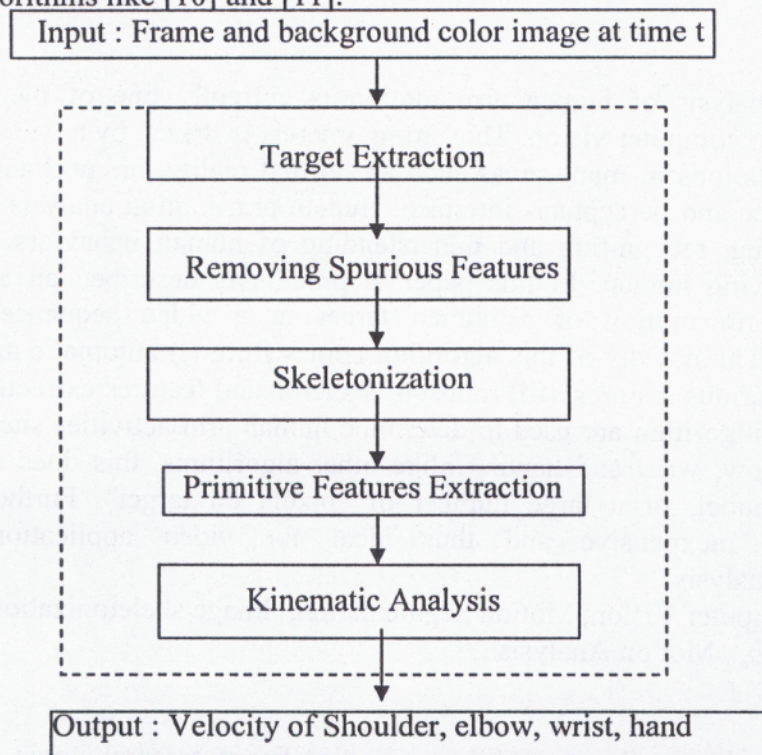


Figure (1): Outline of arm movements algorithm.

It begins by employing a low level process like target extraction in the first stage and then it uses higher level operations that involve some heuristic knowledge in the later stages. Thus each stage makes full use of result yielded by its preceding stage in order to refine the output result. Consequently, all the stages must be carried out progressively according to the given sequence. A detail description of each stage is presented below.

Stage one- target extraction

The initial stage of the human motion problem is the extraction of moving arm from a video stream. There are three conventional approaches to moving target detection: temporal differencing (two- frame or three-frame) [12], background subtraction [13], and optical flow [14]. Temporal differencing is very adaptive to dynamic environments, but generally does a poor job of extracting all relevant features pixels. Background subtraction provides the most complete feature data, but is extremely sensitive to dynamic scene changes due to lighting and extraneous events. Optical flow can be used to detect independently moving targets in the presence of camera motion, however most optical flow computation methods are very complex and are inapplicable to real time algorithms without specialized hardware.

The approach is inspired by a region- based features, presented by [15], and is extracted from preprocessed binary map. These binary maps are representations of the moving and non-moving areas of the sequences, as seen in figure (2) and figure(3) where moving areas are highlighted in black.

The motion is extracted by pixel - wise differencing of consecutive

frames given by :

$$O_1(t) = I_i(x, y) - I_b(x, y) \begin{cases} 1 & > \text{threshold} \\ 0 & < \text{threshold} \end{cases} \quad (1)$$

Where $O_1(t)$ is the Euclidean between image pixels in consecutive frames I_i and background frame I_b , and x, y represent the pixel location.

No motion detection algorithm is perfect. There will be spurious pixels detect, holes in moving features, "interlacing" effects from video digitization processes, and other anomalies. Foreground regions are initially filtered for size to remove spurious features, and then the remaining targets are pre-processed before motion analysis is performed



Figure 2. Input frame image.

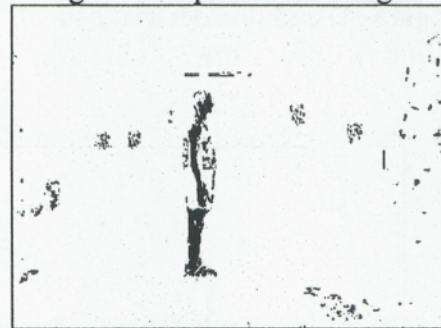


Figure 3. Bitmap produced by stage one.

Stage two- removing spurious features

This stage considers the bitmap produced by the previous stage to contain the moving target that is corrupted by noise. The noise may be appear as a small holes or it may be also appear as object in background scene.

Therefore this stage performs simple morphological operations such as dilation to fill in any small hole in the target area and erosion to remove any small object in the background area. The intention is not necessarily to remove entirely, but to reduce the amount and size of the noise.

It has been found that the brightness of the image is non-uniform throughout the moving targets, while the background region tends to have more even distribution of the brightness. Hence based on this characteristics, background region that was detected can be further eliminated. Standard deviation has been used as the statistically measure of the spatial distribution characteristic of the luminance values (Y) of the color model $YCrCb$ for every target pixels in input frame.

We demonstrated in [16] if the standard deviation $\delta(x, y)$ below a value of 2 then the pixel is considered unlikely to be part of the target. As a result, the output bitmap of the stage two, denoted as $O_2(x, y)$ is shown in figure (4) and it is derived as :

$$O_2(x,y) = \begin{cases} 1 & \text{if } O_1(x,y) = 1 \text{ and } \delta(x,y) \geq 2 \\ 0 & \text{otherwise} \end{cases} \quad (2)$$

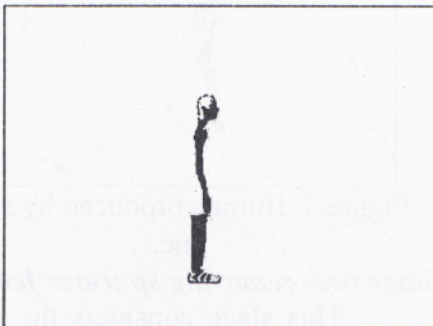


Figure 4. Binarization produced by stage two.

Stage three- skeletonization

A basic method of skeletonization is thinning. There are numerous definitions of a skeleton and hence the thinning algorithms from each other implementation and in performance. The algorithm used parallel thinning algorithm, because it gives skeletons with fewer spurious branches [17-18].

In parallel thinning algorithm, 3×3 window has been used. All kinds of relations (256) formed by 8-neighbors of object pixel have been considered. From these cases, a group of elimination rule can be obtained. All rules are given in figure (5) (where blank denotes 0). These rules were classified according to the amount of black pixels in 8- neighbors. The first column denotes the number black pixel in 8-neighbors of object black pixel and the second column shows the elimination rules. These elimination rules are used also to remove the loss of connectivity and the distortion of the target.

All the rules are applied simultaneously to each pixel. In some peculiar cases, these can not be performed very well. A two- pixel - width or even pixel width in horizontal or vertical direction may be deleted, which would cause the loss of connectivity of target. In order to keep up the connectivity, these pixels should not be deleted. On the other hand, if all two- pixels width are retained, the skeleton would not be one pixel width. The window of figure (6) has been used to resolve this problem [18]. If the target pixel matches one of the templates in figure (6), the pixel should be preserved.

Amount	Elimination Rules
0	Never
1	Never
2	
3	
4	
5	
6	
7	
8	Never

Figure 5. Thinning elimination rules, blank = white pixel and image pixel =1

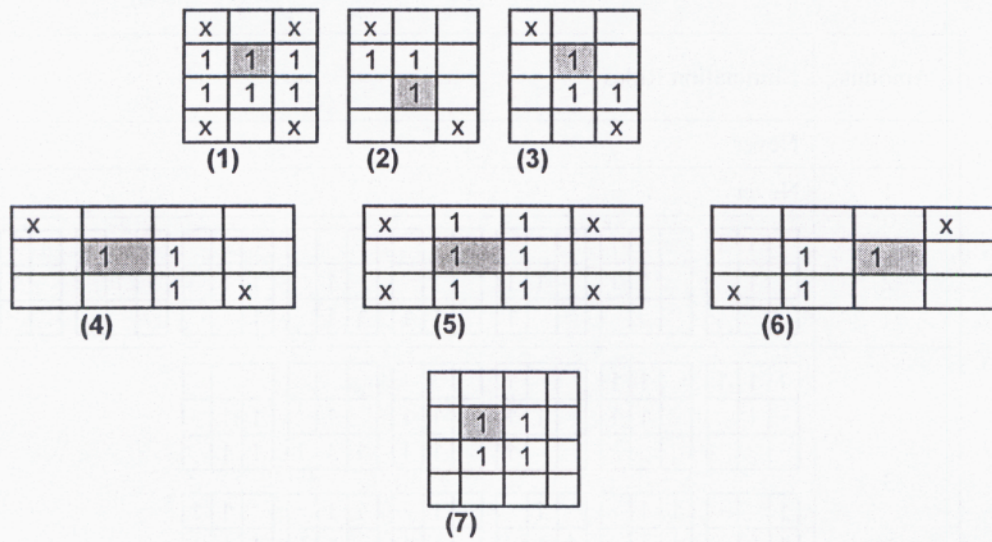


Figure 6. The preserved template used in thinning algorithm, blank denote 0, "x" are do not care .

For every binary image sequences in stage two, we use this algorithm of thinning. As depicted in figure (7), there are some human arm images and thinning results.

Stage four- Primitive feature extraction

An algorithm implementing a 3x3 window is used to trace along the

Binary image						
Thinning						

Figure 7. Skeleton of human and some moving arm images produced by stage three.

path of skeleton recording the structural information of the trace path . A path is described as a list of direction labels leading between a pair of junction or end points, where an end point has a single neighbor and a junction points has two neighbors. This path is stored in a node of a binary tree- where a choice of path to trace exists, a left and right node are formed beneath the current one and their respective paths traced out.

The starting point for tracing the skeleton is determined by dividing the image frame into three horizontal regions, and the top and bottom regions are searched for end points or junction points. This ensures that the starting point does not split a path into two subpaths the left most pixel of the image is used as starting point.

The structural information for each path traced is saved as follows:

- 1- Freeman code chain [19-20]: an 8-directional code describing the direction of each path segment. The codes are shown in figure (8).



Figure 8. Freeman chain code.

- 2- Positional : coordinate describing the start and end points of the path. these coordinates are used to determine positional relationship between loops and between loops and touching path.
- 3- Loop : pointer indicating path joining previously explored section. The completed tracing results in the segmentation of the target into paths or strokes which will be

formed into primitives.

The structure information in the binary tree allows the formation of pattern primitives, or sub-patterns, that are used to describe the target image features. There are two main primitive described in this system: straight lines and curves. A path may be described by a single primitives or by multiple primitives. The structural information in the tree is converted to these primitives using the following definition:

Break point (Separator): divided a path into subpaths more easily described by primitives. A break point satisfies at least one of the two possible conditions:

- Inflection point : a change in curvature, a positive (clockwise) curve followed by a negative (anti-clockwise) curve, or vice versa, as illustrated in figure (9).
- Cusp point: a sharp change in direction, two segments form an acute angle or right angle as illustrated in figure (9).

Straight lines : a straight line is a sequence of points with the same chain code.

Curves : two types of curves open and closed are distinguished . Table (1) illustrates the primitives used in this system.

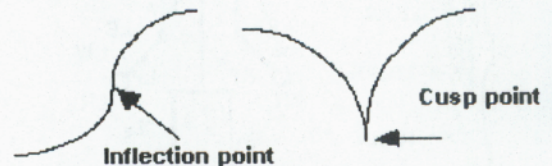


Figure 9. The inflection point and cusp point.

Table 1. Primitive features used in this stage.

Line	—	/		\
Open curve	∪	∩	∪	∩
Closed curve	○			

Then for each target frame, first its free ends have been detected. A moving window of fixed size used to traverse the overall target and the primitives found in different window positions are recorded using a stack. If one of the primitives described is present then the systems records its code and location in sequence as we move from one window position to another. In some window positions where there are no primitives present, the system records an empty code. These codes are used to describe the torso in Table (2) and figure (10).

Stage five- Kinematic analysis

Kinematics is that branch of dynamic which deals with displacement and velocity. The velocity of an objects is calculated by dividing the displacement by the change in time taken to move from the first position to final position as following :

$$V_a = \frac{d_2 - d_1}{t_2 - t_1} = \frac{\Delta d}{\Delta t} \tag{3}$$

where $\Delta d = \Delta r$ (4)

and $r = S_c(x, y)$ (5)

where $S_c(x,y)$ is position location of the primitive features with respect to a particular frame.

Table 2.
Feature points and position.

Symbol	Meaning	Symbol	Meaning
H	Tip of Head	RP	Right Pelvis
J	Jaw	RK	Right Knee
RS	Right Shoulder	SOR	Right Sole
RE	Right Elbow	LP	Left Pelvis
RW	Right Wrist	LK	Left Knee
RH	Tip of Right Hand	SOL	Left Sole
LS	Left shoulder	^w L _e	Length between Wrist to Elbow
LE	Left Elbow	^e L _s	Length between Elbow to Shoulder
LW	Left Wrist	^{sow} L _k	Length between Sole to Knee
LH	Tip of Left Hand	^k L _p	Length between Knee to Pelvis

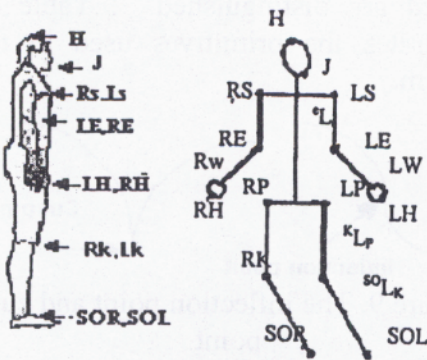
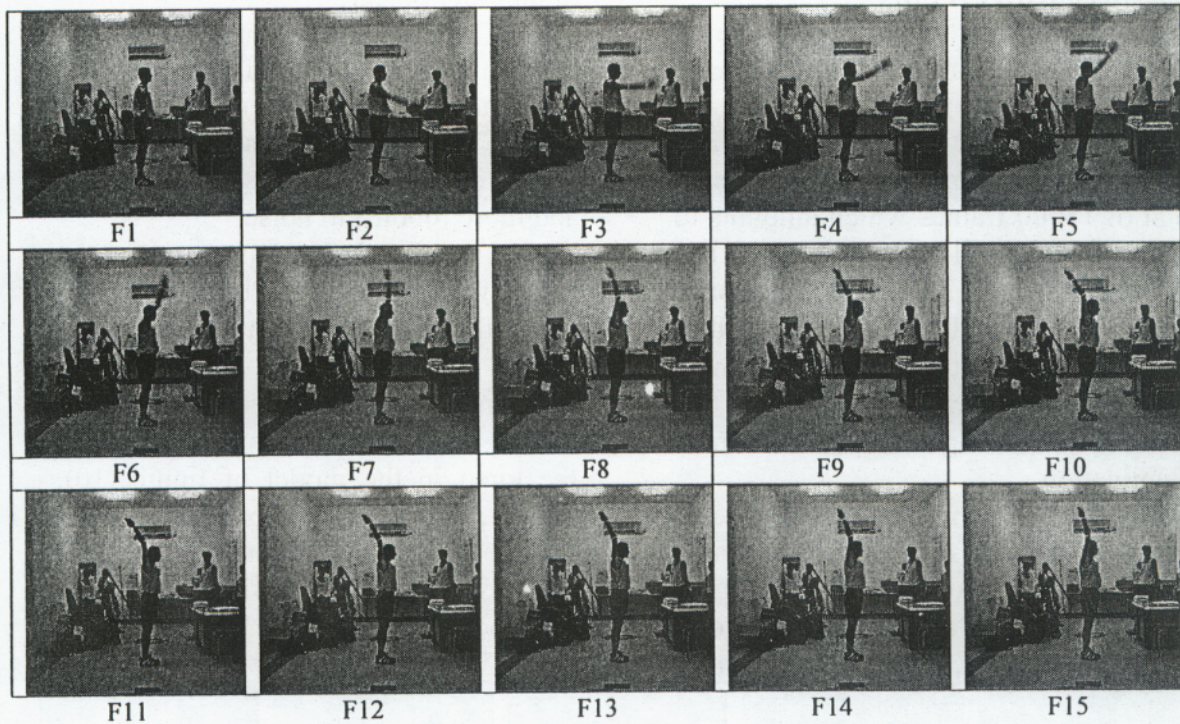
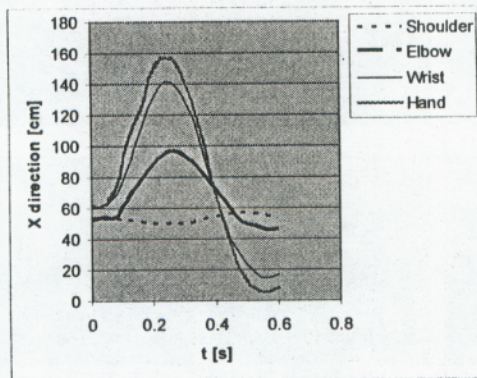


Figure 10. Skeleton model

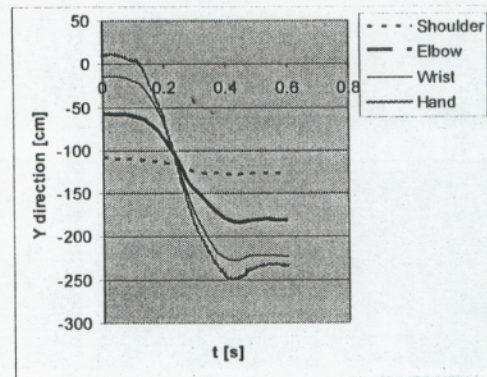
d is the displacement change in position and V_a is the velocity- change in position with respect to time. Figure 11 show the displacement and velocity for arm movement sequences.



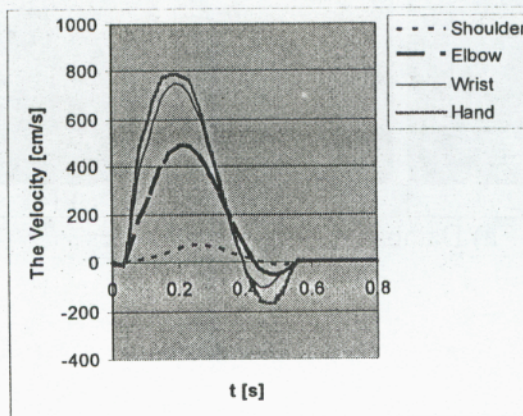
(a)



(b)



(c)



(d)

Figure 11. Human arm movements. a) Image frames from arm movement sequences., b) The displacement in x direction., c) The displacement in y direction., and d) The velocity of shoulder, elbow, wrist and hand.

3. Experimental results

The experiments were carried out on a database of 30 different image sequences. All the sequences are approximately 20 second long and captured at 25 frame per second given a total of 15000 frames which amounts to approximately 10 minutes. The sequences are full 24-bit color and have a resolution of 240 by 320 pixels. It is important to note that the video camera and all its adjustments are fixed in the period of capturing and also the whole illumination.

The moving arm was detected using the methods described in stage-one and stage- two. As shown in figure (4) the image contains a sharp image of the tracked object, and a blurred image

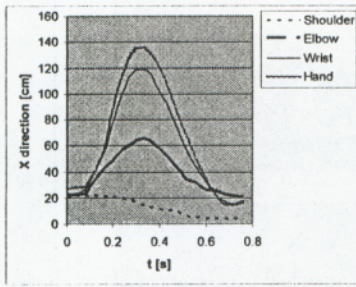
of all other object.

In stage-three there are some arm images and thinning results as illustrated in figure (7). The results prove that the parallel thinning algorithm in this stage is very efficient and more robust to noise.

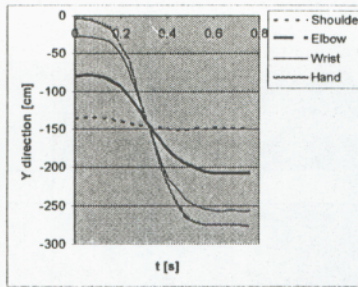
Once a skeleton is extracted the primitive features can be determined using the technique that prescribed in stage- four. These features are confirmed the confidence of the result values of the torso silhouette based on the torso in the model of figure (10). Finally, figure (12) shows 15 frames and the corresponding displacement and velocity for each arm movement sequences.



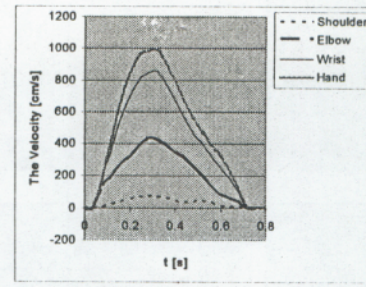
a) Databasel-2 of image frames



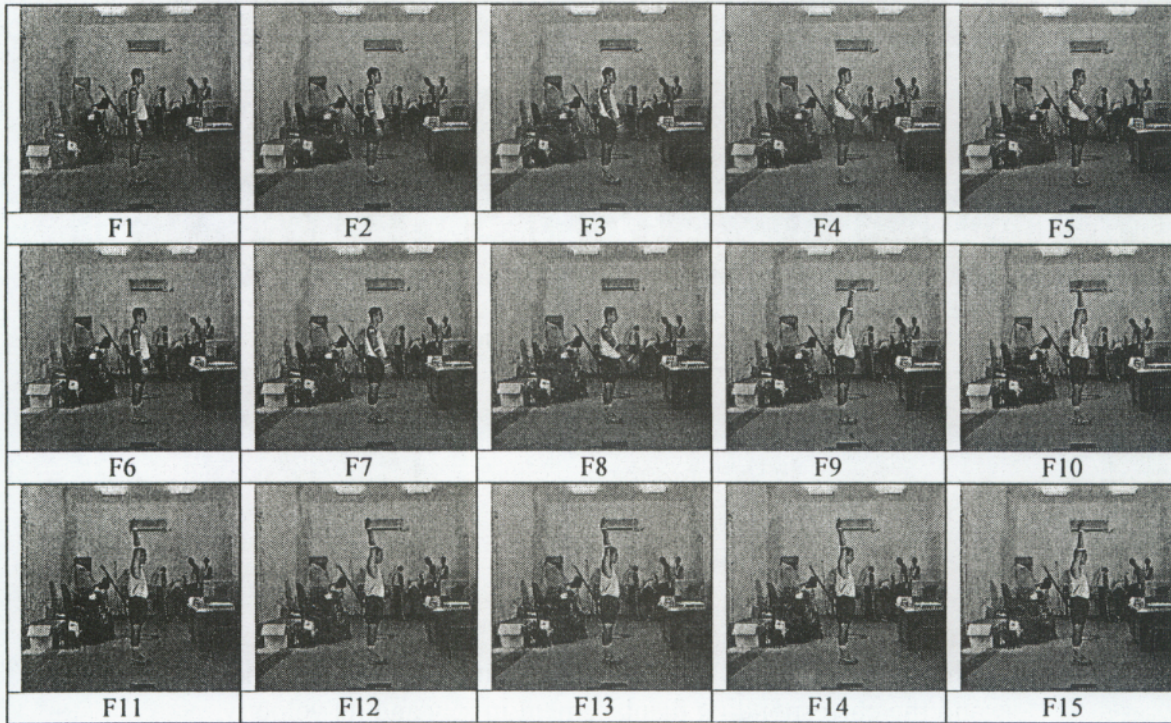
b) The displacement in x



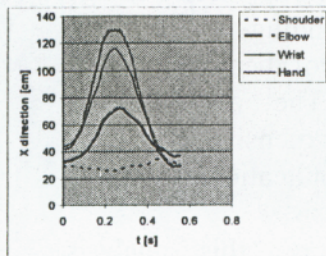
c) The displacement in y



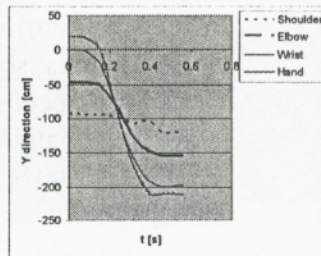
d) The Velocity



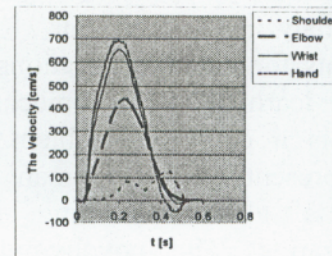
a) Database 1-3 of image frames



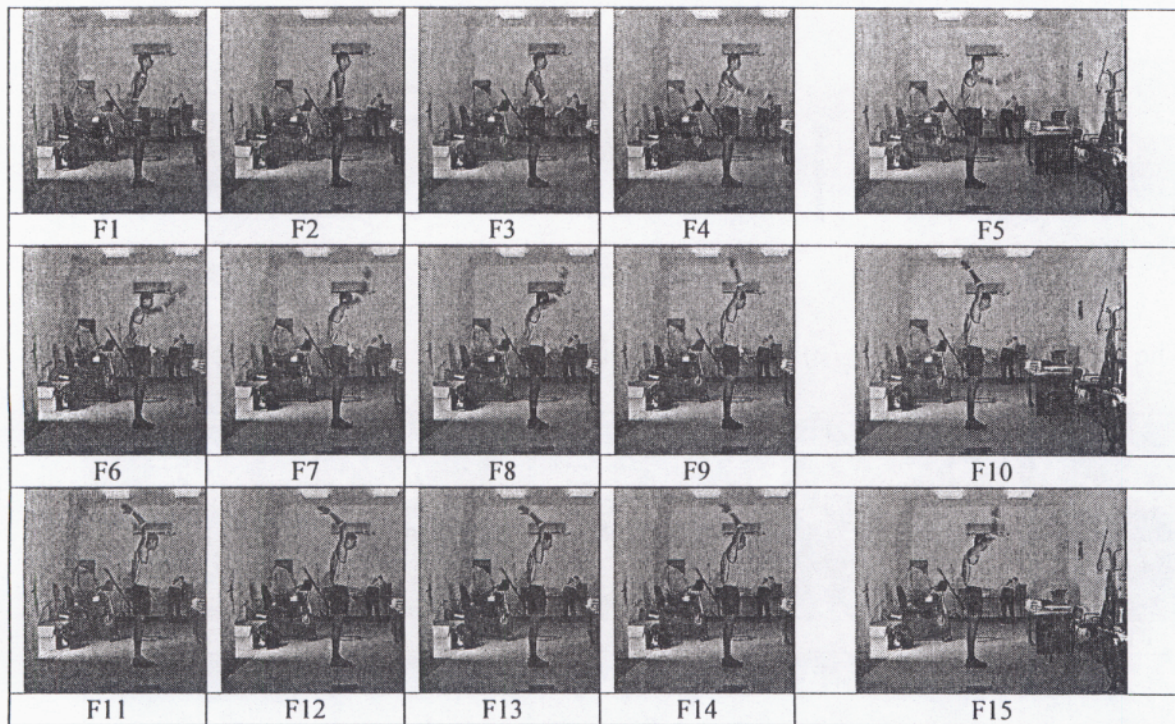
b) The displacement in x



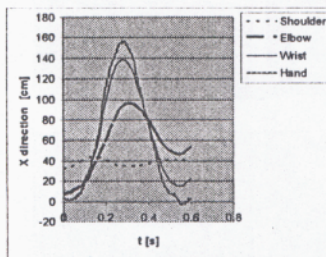
c) The displacement in y



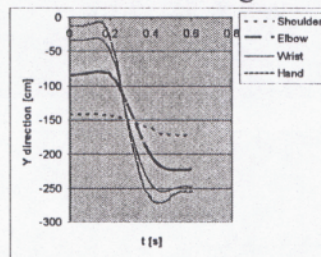
d) The Velocity



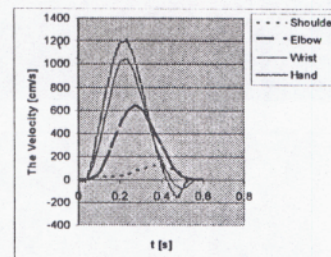
a) Database 1-4 if image frames



b) The displacement in x



c) The displacement in y



d) The Velocity

Figure 12. The database image frames from arm movements sequences and the corresponding displacement in x, y and its velocity.

4. Conclusion

Analyzing human arm movements for video applications such as robot learning and biomechanical analysis is a complex problem. This paper presents the human arm movements algorithm using image skeletonization. The novelty of this algorithm comes from (i) automatic arm detection, (ii) robust and fast features extraction.

The feasibility and robustness of the proposed algorithm has been shown

using long and inherently noisy image sequences corresponding to different arm activities, some of them recorded at different speeds. The algorithm is computationally inexpensive and thus ideal for video applications such as biomechanical analysis.

In the future, this analysis technique will be applied to more complex human motions such as crawling, jumping, and so on.

References

- [1] S. Maybank, T. Tan, "Introduction to special section on visual surveillance", *International Journal of Computer Vision*, 37(2), 2000.
- [2] J. Steffens, E. Elagin, H. Neven, "person spotter- fast and robust system for human detection tracking and recognition", *Proc. of IEEE Intl. Conf. on Automatic Face and Gesture Recognition*, PP. 516-521, 1998.
- [3] J.J. Little, J.E. Boyd, "Recognizing people by their gait: The shape of motion videre", *Journal of computer vision Res.*, The MIT Press, 1(2), 1998.
- [4] P.S. Huang, C.J. harris, M.S. Nixon, "Human gait recognition in canonical space using temporal templates", *Proc. of IEEE Vis. Image Signal Process*, 146(2), PP. 93-100, 1999.
- [5] J. Segen, S., Kamar, "Shadow gestures: 3D hand pose estimation using a single camera", *Proc. of IEEE CS Conf. on Computer Vision and Pattern Recognition*, PP. 479-485, 1999.
- [6] S. Arsenean, J.R. Cooperstok, "Real-time image segmentation for action recognition", *Proc. of IEEE Pacific Rim Conf. on Communications, Computer and Signal Processing*, PP. 86-89, 1999.
- [7] R.T. Collins et al., "System for video surveillance and monitoring: VSAM final report, CMU-RI- TR-00-12, Technical Report, Carnegie Mellon University, 2000.
- [8] T.B. Moeslund, E. granum, "A survey of computer vision- based human motion capture", *Computer Vision and Image Understanding*, 81(3), PP. 231-268, 2001.
- [9] M. Oven, C. papahegiou, P. Sinha, E. Osuna and T. Poggio, "Predestrian detection using wavelet templates", *Proc. IEEE CUPR 97*, PP. 193-199, 1997.
- [10] P. Tsai, M. Shah, K. Ketter, and T. Kasparis, "Cycling motion detection for motion based recognition", *Pattern Recognition*, Vol. 27, No. 12, PP. 1591-1603, 1994.
- [11] C. Wren, A. Azarbajejani, T. Darrell and A. Pentland, "Pfinder: Real time tracking of the human body", *IEEE Trans. Pattern Anal. Machine. Intell.*, Vol. 19, No. 7, PP. 780-785, July, 1997.
- [12] C. Anderson, P. Burt, and G. Van Derwal, "Change detection and tracking using pyramid transformation techniques", *SPIE Intelligent Robots and Computer Vision*, Vol. 579, PP. 72-78, 1985.
- [13] I. Haritaoglu, P., D. Harwood, and L.S. Davis, "W⁴ who? When? Where? What? A real time system for detecting and tracking people", *Face and Gesture Recognition 98*, PP. 222-227, 1998.
- [14] J. Barron, D. Fleet and S. Beanchemin, "Performance of Optical flow Techniques", *Int. J. Comput. Vis.*, Vol. 12, No. 1, PP. 42-77, Jan., 1994.
- [15] J. Yamato, J. Ohya, and K. Ishil, "Recognizing Human Action in Time- Sequential Images Using Hidden Markov", *IEEE Intl. Proc. on Computer Vision and Pattern Recognition*, PP. 379-385, 1992.
- [16] I.F. El-Nahry, H.M. El-Hendy, "Locating colorful signs features of moving gymnast", *Port Said Engineering Research Journal*, Vol. 8, No. 1, 2004.

- [17]Y. Xia, "Skeletonization via the realization of the five front's propagation and estimation in digital binary shapes", IEEE Trans Pattern Analysis and Machine Intell., PAMI 11, PP. 1076-1086, 1989.
- [18]L. Huang, G. Wan, C. Liu", An improved parallel thinning algorithm", IEEE Proceeding of 7th International Conference on Document Analysis and Recognition, 2003.
- [19]S. Singh and A.Amin, "Neural network recognition of hand printed characters", Neural Comput. & Applic.,8:67-76,1999.
- [20]A. Blake, M. Isard and D. Reynard", Learning to track curves in motion of contours", Artificial Intel. J, 78, PP. 101-133, 1995.
- [21]H. Fujiyoshi, A.J. Lipton, and T. Kanade, "Real-time human motion analysis by image skeletonization", IEICE Trans. Inf. & Sys., Vol. E 87, No. 1, 2004.
- [22]Tae-Ho Kim, Jo Kang-Hyun, "Extraction of skeleton features using human silhouette and skin", Proc. of the 7th Korea Russia International Symposium, 2003.