Available online at www.sciencedirect.com
ScienceDirect
Biomedical Signal Processing and Control 2 (2007) 59–67
Biomedical Signal Processing and Control
www.elsevier.com/locate/bspc

Technical note

Design of a marker-based human motion tracking system

A. Kolahi, M. Hoviattalab *, T. Rezaeian, M. Alizadeh, M. Bostan, H. Mokhtarzadeh

Department of Mechanical Engineering, Biomechanics Lab, Sharif University of Technology, Azadi Ave., Tehran, Iran

Received 29 July 2006; received in revised form 7 February 2007; accepted 7 February 2007
Available online 27 March 2007

Abstract

In this paper a complete design of a high speed optical motion analyzer system has been described. The main core of the image processing unit has been implemented by the differential algorithm procedure. Some intelligent and conservative procedures that facilitate the search algorithm have also been proposed and implemented for the processing of human motions. Moreover, an optimized *modified direct linear transformation* (MDLT) method has been used to reconstruct 3D markers positions which are used for deriving kinematic characteristics of the motion. Consequently, a set of complete tests using some simple mechanical devices were conducted to verify the system outputs. Considering the system verification for human motion analysis, we used the system for gait analysis and the results including joint angles showed good compatibility with other investigations. Furthermore, a sport application example of the system has been quantitatively presented and discussed for Iranian National Karate-kas. The low computational cost, the high precision in detecting and reconstructing marker position with 2.39 mm error, and the capability of capturing from any number of cameras to increase the domain of operation of the subject, has made the proposed method a reliable approach for real-time human motion analysis. No special environment limitation, portability, low cost hardware and built in units for simulations and kinematic analysis are the other significant specifications of this system.

Keywords: Image processing; Optical tracker system; Human motion tracking; Deinterlacing; Modified direct linear transformation; Karate

1. Introduction

Human motion capture consists of the recording of human body movements for immediate or delayed analysis and playback. The information captured can be as simple as the body position in space or as complex as the deformations of the face and muscle masses. Motion capture for computer character animation involves the mapping of human motion onto the motion of a desired computer character. This mapping can be direct, like the human arm motion controlling a character's arm motion, or indirect, in the case of a human hand and finger patterns controlling a character's skin color or emotional state [1].

Over the past years many methods have been developed to capture human motions [2,3]. The history of this work goes back to joystick and controllers that enabled the puppeteer to control many facial parameters of the computer character including the mouth, head, neck, and the eyes [1]. Later, sophisticated mechanical methods and sensor-based motion capturing devices including magnetic and data gloves were developed [4,5]. The major problem of these methods was the motion limitation of the subject. In recent years, real-time optical systems have been developed that have successfully overcome these limitations.

Optical tracker systems typically use small markers attached on the body of the subject and a set of two or more cameras focused on the subject to capture its motions. An image processing program detects the markers captured by the cameras, and by combining 2D data, calculates the three-dimensional positions of the markers throughout the capture time.

The main limitation of current optical trackers is in the image processing unit, which detects the actual location of the markers. In fact the trace of the marker may be lost in some instances of time where the image processing software does not detect the marker position or the marker is not captured in some frames. Each of these problems should be handled in an appropriate way; the solution of the first problem is done by using an appropriate algorithm for tracking, while the second one needs to use of multiple cameras to capture the motions and

* Corresponding author.
E-mail address: hoviat@sharif.edu (M. Hoviattalab).

1746-8094/$ – see front matter
doi:10.1016/j.bspc.2007.02.001

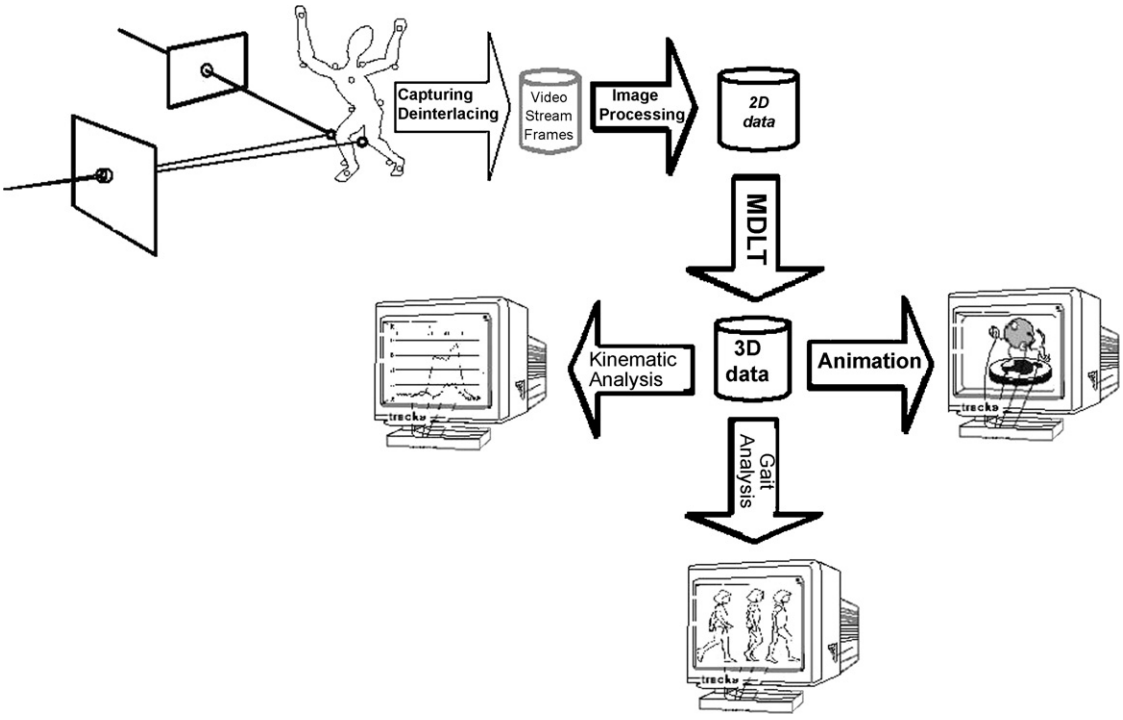

Fig. 1. A block diagram of the hardware and software units.

to prevent losing the track of the markers. The other method to handle such situations is to use model-based motion tracking systems [6].

The purpose of this study was the development of Sharif Motion Analyzer (SMA); a motion analyzer for Biomechanics Laboratory at Sharif University of Technology. Considering justification we quantitatively compared the kinematic parameters for some simple machines from direct measuring and SMA output. Moreover, the verification of SMA for human motion analysis has been presented in this work. We proposed a method named ''differential algorithm'' which is based on the simple concept of finding the average of *Mean Square Error* (MSE) between each Red–Green–Blue (RGB) values of the markers and the frame matrices. This method has been used as the core of image processing unit.

2. Methodology

This system includes both hardware and software components. A block diagram of the hardware and software units can be seen in Fig. 1. Hardware unites are composed of a Pentium IV computer with a 3.06 GHz Intel CPU (with 1 MBytes Cache), and 1 GBytes RAM together with an I-Link Digital Input/Output (IEEE 1394 compliant) video grabber card were used for the capturing and processing units. Two high speed JVC GR-DVL9800U cameras which are able to capture 200 frames per second (fps), a calibration frame of size 1.3 m × 1.2 m × 1.0 m with 26 spherical fiber markers of diameter 3.5 cm diameter placed in special positions (Fig. 2), two 800 watts professional projectors with 2 m legs were used for the recording. The markers used for motion tracking of the target object were light-weighted spherical fiber markers of diameter 1.8 and 3.5 cm in phosphoric red, green, and white colors. The MATLAB® software version 7.0.4.365(R14) was used as the main platform to implement SMA. The 3DS-MAX® software version 7.0 was also used in combination with MATLAB to animate the artificial character and the Microsoft® Excel software was used to save markers positions data.

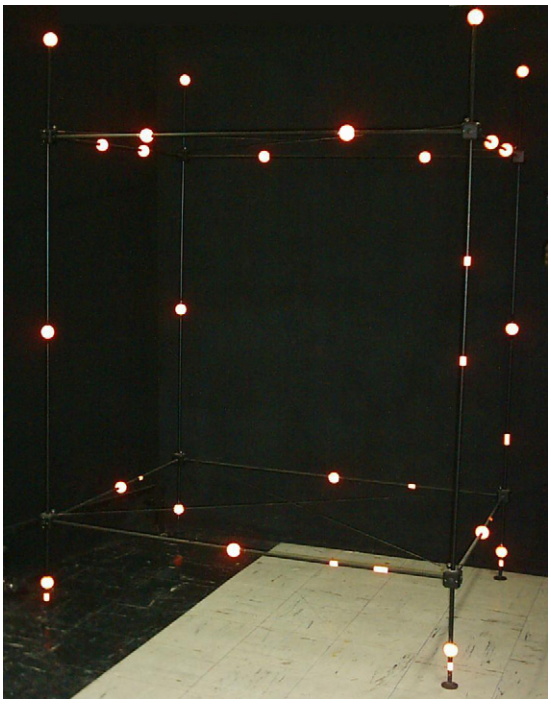

Fig. 2. The calibration frame and the Markers.

The SMA software units consist of *capturing*, *deinterlacing*, *image processing*, *synchronizing*, *reconstructing of the 3D coordinates*, and *animating the computer character subroutines*. Fig. 3 shows the interface of SMA. The arrows show the logic pattern of working with SMA from capturing process to kinematic analysis.

2.1. Capturing

As a first step the capturing program captures video streams from the cameras and provides matrices of RGB data. Throughout the capturing procedure, the subject wears a black suit with markers attached on special locations. Two cameras with different angles of view aligned in the way that they cover the whole motions of the subject and all of the markers (if possible). In general, the distance between a camera and a subject depends on the camera's resolution and markers' size. For example the JVC GR-DVL9800U camera when recording at 50 fps has a resolution of 576 × 720 pixels. For a reliable and smooth tracking and achieving a precision of millimeter the markers' size in captured frames should be at least 4 × 4 pixels. In this case for markers of 3.5 cm diameter the maximum distance between the cameras and markers will be 7.5 m and cameras would view a cross sectional area of 5.04 m × 6.30 m. Projectors are needed to provide necessary light for the cameras. The captured video stream is transferred to the computer through the grabber card by the capturing program. Subsequently an RGB image is stored in MATLAB as an $m \times n \times 3$ data array. Where m is the width and n is the height of the image and the number 3 indicates red, green, and blue components for each individual pixel [7]. This format prepares an appropriate framework to handle video data and to access RGB values of each pixel.

2.2. Deinterlacing

Interlacing is in fact a smart way to compress a movie when one cannot use digital compression methods. This is the way digital camcorders and digital VCRs record and digital broadcasting is done. One second of a movie consists of at least 25 frames equals to 50 interlaced pictures (fields) [8]. The Deinterlacing program reconstructs the original frames from mixed video frames. To do so, this program gets a frame of interlaced video stream and decomposes two images from the even and odd fields (lines). Then a cubic interpolation is performed on each image to estimate the lost fields. This process is done for all frames of the video data and the original frame rate is obtained. For higher capturing speeds, some cameras combine two or four interlaced frames with the cost of resolution reduction. For example one second of interlaced frames that have four sub-frames in each frame will reconstruct 100 interlaced frames that is equal to 200 deinterlaced frames.

2.3. Image processing

The image processing unit obtains 2D coordinates of all of the markers in each frame. It gets deinterlaced frames as the input and produces an Excel file of 2D coordinates of markers in an appropriate coordinate system defined by the user (Fig. 4).

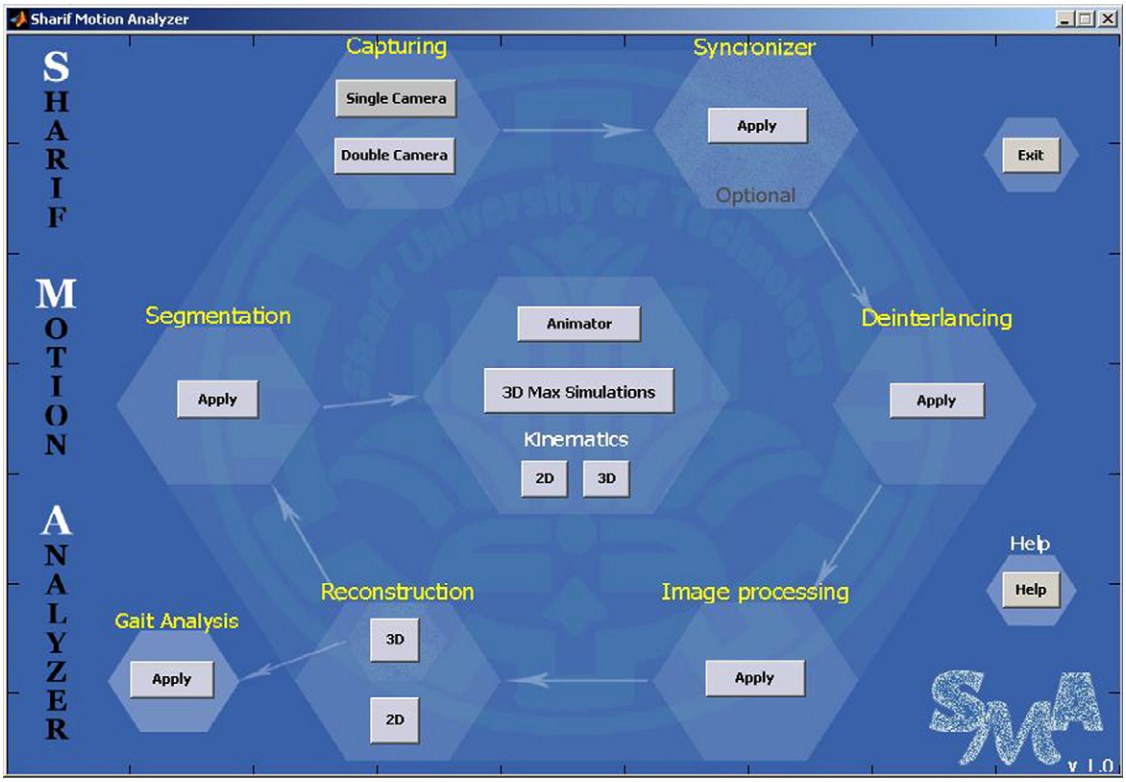

Fig. 3. The SMA user interface.

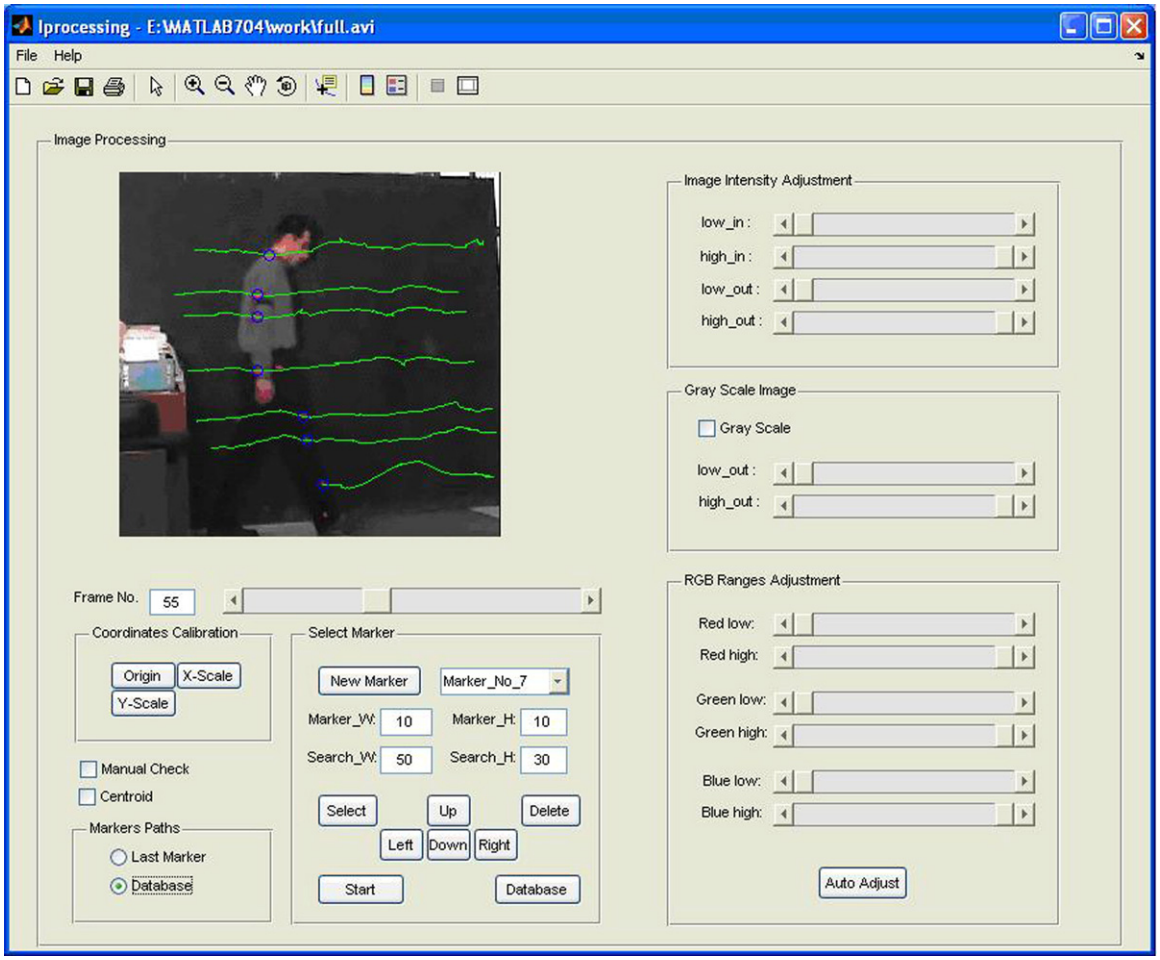

Fig. 4. Image processing unit and the paths of the markers after image processing.

It is known that according to the biomechanical kinematic analysis of the human motion, the acceleration of a moving body may not have any discontinuities [9]. This fact may be used to improve the image processing procedure, by performing data fitting on the consecutive frames to assure the continuity of the second order derivatives. The smooth fitted data has been used to predict the position of the markers in the next frame.

Note that due to the importance of the image processing unit and the necessity of precise marker position tracking, it is important to use fast and numerically stable algorithms in this unit.

2.4. Synchronization

This unit is used to synchronize the video streams of two cameras. It needs a flasher to produce an instantaneous light that could be captured by all cameras. The synchronization unit searches all of the frames of two video streams and finds the frame with flash lights. Using these synchronization frames, the rest of the frames of the two cameras are time synchronized.

2.5. 3D coordinates

By combining different 2D coordinates a 3D coordinate of the markers is obtained. This is done by the 3D coordinate reconstruction unit. This unit uses a *Direct Linear Transformation* (DLT) method to combine 2D marker positions of two cameras [10,11]. This method uses the calibration frame to determine some unknown constants of DLT equations. As mentioned before, the calibration frame (Fig. 2) is a frame with some markers in specified spatial positions. The cameras 2D data are a mapping of the 3D data obtained from DLT equations. These data should coincide with the actual spatial position of calibration frame markers. Knowing the calibration constants, it is possible to combine and construct 3D positions of all markers. Consequently, the extended data is used as an input for biomechanical kinematic analysis of human motions.

2.6. 3D animating

In the last step the 3D data obtained from the previous step were entered into the animation unit to show a stickman diagram with 3D marker data (Fig. 5). It also provides input data to animate computer character in 3DS-MAX's environment. 3DS-MAX's character studio toolbox produces a character skeleton with any number of links that are arranged in a tree structured hierarchy [12]. This hierarchy ensures that segments inherit the rotations applied to joints higher in the tree. For example a rotation applied at the shoulder joint, causes the entire arm to rotate, and not just the upper arm segment. To

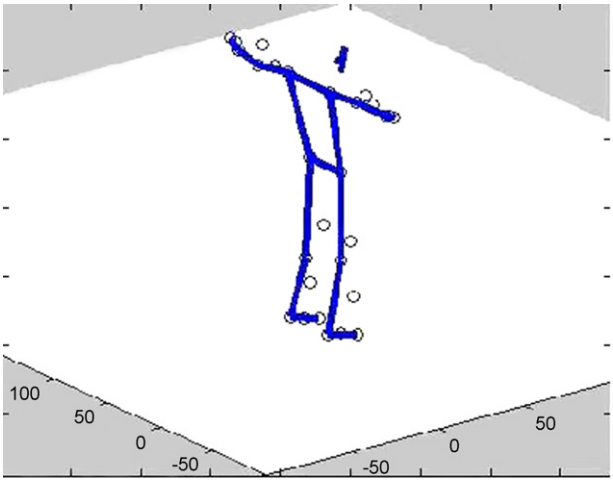

Fig. 5. Animation of captured data.

animate the character, 3DS-MAX can move the links from a specified 3D position defined by the operator. To this end, according to the proposed tree of structured hierarchy, the system converts 3D marker data to a specific format of *Character Studio Max* (CSM) files, to be used by *Character Studio*.

3. Differential algorithm and the image processing software

In the developed software, the operator specifies several regions as the marker locations. These regions are preprocessed to obtain a *marker matrix* template to be tracked in the next frames.

The main core of the image processing unit is the differential algorithm procedure. This procedure calculates the average of MSE between each RGB values of the marker matrix and all possible matrices with the same size inside the search area of the next frame matrix. The region having the least MSE is selected to be the marker position.

For example, suppose that the marker area is 10×10 pixels and the search area is 100×50 pixels. By considering the RGB values of each pixel, the marker matrix will be $3 \times 10 \times 10$ and similarly the search area matrix will be $3 \times 100 \times 50$. The differential algorithm aligns the first element of marker matrix with the first element of search area matrix and then calculates the average of MSE between elements of these matrices. In the next step the marker matrix slides one pixel forward and its first element is aligned with the second element of the search area matrix, and the MSE calculation is repeated. This procedure continues until the last element of markers matrix reaches to the last element of the search area matrix (in the mentioned example, 3731 MSE values will be calculated). As mentioned the region with the least MSE value is selected as the location of the marker in the current frame; which will be used as the reference marker matrix for the next step.

To increase the reliability of detection and tracing of markers, some conservative procedures facilitate the search procedure of the main core:

- Prediction of markers positions in consecutive frames. To this end it uses the velocity and acceleration of markers to predict its position in the next frame.
- Increasing the marker detection speed, by predicting the approximate marker search region using markers' velocity and acceleration and their directions.
- An intelligent procedure that measures the probability of error in detecting markers and produces warnings to get help from the operator to position the marker. It calculates the amount of differences between two regions in previous and current frames. Higher values of differences report a warning to get help from user.
- Automatic image adjustment, which bounds the color ranges and filters possible image noises, as a result it provides images that markers are brightened and enhanced for more efficient detection.

A significant characteristic of the image processing unit is its compatibility with various local conditions. The embedded features provide using different colors for subject suits, markers and background.

4. 3D reconstruction

One of the most widely used three-dimensional video analysis techniques has been developed by Abdel-Aziz and Karara [10]. Their method, which requires two or more cameras, involves the DLT method to transform image coordinates into object space coordinates. It establishes a relationship between digitized coordinates from the two or more camera views and the corresponding coordinates in three-dimensional space.

The DLT algorithm can be reduced to a set of two equations, expressed as:

$$u + \Delta u = \frac{L_1 X + L_2 Y + L_3 Z + L_4}{L_9 X + L_{10} Y + L_{11} Z + 1},$$
$$v + \Delta v = \frac{L_5 X + L_6 Y + L_7 Z + L_8}{L_9 X + L_{10} Y + L_{11} Z + 1} \quad (1)$$

where u and v are the image coordinates and Δu and Δv are the image coordinate corrections for lens distortion. The object point coordinates are X, Y, and Z, and the constants L_1–L_{11} are the DLT parameters. The DLT parameters define calibration, position, and orientation of each camera. Optical errors can be expressed as:

$$\Delta u = \xi(L_{12} r^2 + L_{13} r^4 + L_{14} r^6) + L_{15}(r^2 + 2\xi^2) + L_{16}\xi\eta,$$
$$\Delta v = \eta(L_{12} r^2 + L_{13} r^4 + L_{14} r^6) + L_{16}(r^2 + 2\eta^2) + L_{15}\xi\eta \quad (2)$$

where

$$[\xi, \eta] = [u - u_0, v - v_0], \quad r^2 = \xi^2 + \eta^2 \quad (3)$$

Parameters L_{12}–L_{14} in above equations are due to lens distortion and L_{15}, L_{16} are due to the fact that the object points and the corresponding image points do not converge to a single point called projection point. The projection point's coordinate is shown by $[u_0, v_0]$. Eq. (1) indicates that at least 11 parameters are needed to calculate the 3D coordinates of markers, and if

more precision is desired, more parameters should be used, however the optical errors are not considerable and mostly ignored in DLT method.

Matrix Eq. (1) implies that:

$$X_iL_1 + Y_iL_2 + Z_iL_3 + L_4 - u_iX_iL_9 - u_iY_iL_{10} - u_iZ_iL_{11} = u_i,$$
$$X_iL_5 + Y_iL_6 + Z_iL_7 + L_8 - v_iX_iL_9 - v_iY_iL_{10} - v_iZ_iL_{11} = v_i \tag{4}$$

So by given a set of N spatial points at their image coordinate, we obtain the following matrix equation:

$$\begin{bmatrix} X_1 & Y_1 & Z_1 & 1 & 0 & 0 & 0 & 0 & -u_1X_1 & -u_1Y_1 & -u_1Z_1 \\ 0 & 0 & 0 & 0 & X_1 & Y_1 & Z_1 & 1 & -v_1X_1 & -v_1Y_1 & -v_1Y_1 \\ \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot \\ \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot \\ \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot \\ \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot \\ \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot \\ \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot \\ \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot \\ X_N & Y_N & Z_N & 1 & 0 & 0 & 0 & 0 & -u_NX_N & -u_NY_N & -u_NZ_N \\ 0 & 0 & 0 & 0 & X_N & Y_N & Z_N & 1 & -v_NX_N & -v_NY_N & -v_NZ_N \end{bmatrix} \begin{bmatrix} L_1 \\ L_2 \\ L_3 \\ L_4 \\ L_5 \\ L_6 \\ L_7 \\ L_8 \\ L_9 \\ L_{10} \\ L_{11} \end{bmatrix} = \begin{bmatrix} u_1 \\ v_1 \\ u_2 \\ v_2 \\ \cdot \\ \cdot \\ \cdot \\ \cdot \\ \cdot \\ u_N \\ v_N \end{bmatrix} \tag{5}$$

This equation has 11 unknowns and each point providing 2 constraint equations, we need at least six points to solve the equation.

The best least squares estimate of the L_i is obtained using the pseudo-inverse as derived below.We can write (Eq. (5)) in matrix form of:

$$AL = b \tag{6}$$

where A is the $2N \times 11$ matrix, $L = (L_1, \ldots, L_{11})^{\mathrm{T}}$ is an 11-vector, and b is the $2N$-vector in (Eq. (5)). To estimate the unknown elements embodied in the vector L, we formulate the problem as one of minimizing $||AL - b||^2$, where $||\ ||$ denotes the 2 norm of the vector.

Let

$$Q = ||AL - b||^2 = (AL - b)^{\mathrm{T}}(AL - b) \tag{7}$$

Then our objective is

$$\min_L Q = \min_L (AL - b)^{\mathrm{T}}(AL - b) \tag{8}$$

Differentiating Q with respect to L and setting to 0 gives

$$A^{\mathrm{T}}(Aq - b) = 0 \Rightarrow A^{\mathrm{T}}Aq = A^{\mathrm{T}}b \Rightarrow q = (A^{\mathrm{T}}A)^{-1}A^{\mathrm{T}}b \tag{9}$$

In general, the term is commonly known as the pseudo-inverse of the matrix A and is often denoted by $A+$.

In order to determine the DLT parameters, a calibration frame (Fig. 1) with no coplanar points of known coordinates relative to an arbitrary reference frame is filmed first. Each calibration point provides two equations, and, a minimum of six calibration points are required. But it should be noticed that as pseudo-inverse method is used to obtain the DLT parameters, using more than 6 points may decrease the construction errors. Once the DLT parameters have been determined, the position of any marker within the object space can be obtained. This is accomplished by transforming the image coordinates from each camera into the global coordinates relative to the arbitrary reference frame mentioned earlier. However, if the cameras are altered in any way, the calibration process must be redone.

The DLT equations consist of 11 DLT parameters that just 10 parameters are independent and one of them can be expressed in terms of the other 10 parameters, even though the system has only 10 independent unknown factors. In other words, one of the DLT parameters must be redundant and we need to add a non-linear constraint to the system to solve this problem. In the standard 11-parameter DLT, one computes 11 parameters independently using the least square method. In this process the dependency among the 10 independent factors is impaired, resulting in a non-orthogonal transformation from the object–space reference frame to the image–plane reference frame.

To consider the affects of orthogonal transformation, the Modified DLT (MDLT) was developed. In MDLT method, one of the parameter is expressed in terms of the other 10 parameters [11].

$$(L_1L_9 + L_2L_{10} + L_3L_{11})(L_5L_9 + L_6L_{10} + L_7L_{11})$$
$$= (L_1L_5 + L_2L_6 + L_3L_7)(L_9^2 + L_{10}^2 + L_{11}^2) \tag{10}$$

Eq. (10) clearly shows the non-linear relationship among the 11 DLT parameters. Now, the question is how to add this non-linear constraint to the linear system of the DLT. An iterative approach may be used:

1. Compute the 11 DLT parameters using the conventional DLT in the first iteration.
2. From the second iteration, solve Eq. (10) for one parameter (for example L_1) by using the values of L_2–L_{11} obtain from the previous step. So we can express image coordinates as functions of L_2–L_{11}:

   $$u = f(L_2, \ldots, L_{11}), \qquad v = g(L_2, \ldots, L_{11}) \tag{11}$$

   Using Taylor's first order approximation we can change nonlinear functions of Eq. (11) to linear ones:

   $$u = f_0 + \frac{\partial f_0}{\partial L_i}\mathrm{d}(L_i), \qquad v = g_0 + \frac{\partial g_0}{\partial L_i}\mathrm{d}(L_i) \tag{12}$$

where f_0 and g_0 are:

$$f_0 = f[L_i(\text{PreviousStep})], \qquad g_0 = g[L_i(\text{PreviousStep})] \tag{13}$$

Using above equations and least square method, values of $d(L_i)$ are calculated. DLT parameters for present step are expressed as:

$$L_i(\text{Present}) = d(L_i) + L_i(\text{Previous}) \tag{14}$$

3. Repeat Step 2 until parameters of two consecutive steps differ less than desired precision.

The DLT method is suitable and convenient for a number of reasons. Firstly, only two non-metric (where the internal camera parameters such as lens' focus distance, lens distortion are not previously known) cameras are required, which reduces the initial capital expense and digitizing time during analysis. Secondly, the orientation of the image reference frame with respect to the object reference frame, and the distance from the camera to the object do not need to be known. Thirdly, this technique is extremely flexible in that the cameras are able to be positioned almost anywhere, as long as the object of interest is in the field of view of both cameras. Finally, the ability of the DLT method to locate points in space has been shown to meet the accuracy requirements associated with human movements [13]. An investigation of the accuracy of the DLT reconstruction found errors up to 6–7 mm for the resultant three-dimensional coordinates within the calibration space [14]. However, when points outside the calibration space are analyzed, the error increases significantly [15]. The reconstruction error for the designed system is 2.39 mm. For evaluating the reconstruction and kinematic errors, we did different verification tests. In these test markers were attached to simple devices (bifilar pendulum, and gyroscope) with known motion. After capturing markers' trajectory and reconstructing the 3D coordinates of them, desired parameters such as length of cords or angular velocity of motion were calculated [16] and compared with direct measuring. Table 1 shows the results of verification tests.

In 3D reconstruction unit, 2D coordinates of markers can be loaded as an Excel input files from any number of cameras. Then data for each marker can be filtered using different smoothing methods like Discrete Fourier Transformation (DFT). Data processing by direct linear transformation or modified DLT will provide 3D coordinates of the measured marker.

The obtained marker's 3D coordinates is consequently used to derive the linear and angular displacement, velocity or other kinematic characteristics based on dynamic modeling equations of rigid body [17] using the developed software units following step by step user friendly interface illustrated in Fig. 3.

Table 1
Errors calculated on the kinematic verification tests

S.D. (%)	Mean error	Desired parameter	Device name
1.01	1.5	Cord length	Bifilar pendulum
1.19	1.03	Bar length	Bifilar pendulum
0.90	1.00	Angular velocity	Gyroscope

5. Results and discussions

Considering 2.39 mm for resultant 3D coordinates and mean error up to 1–1.5% for linear displacement and angular velocity illustrated in Table 1, has made SMA a reliable system for motion analysis. According to the importance of gait in human motion analysis, the relevant option as a gait oriented kinematic output has been performed in SMA software package as illustrated in Fig. 3. The result for joint flexion angels during a gait cycle were extracted for sample subjects. Fig. 6 shows the comparison between knee and ankle flexions for gait cycle obtained by SMA and the other investigators. In this figure tick lines shows the results of SMA and square-point curve *A* is the results of Winter [18] for knee flexions and circle-point curve *B* shows the results of Whittle for ankle flexions [19].

Although the gait patterns are the same, SMA kinematic outputs show lower values for knee angle in initial phase of stance and lower plantar flexion for ankle angle. The depicted differences are due to different subjects, their speed and pattern of walking. Furthermore, as it is mentioned in some literatures, the range of motion of a knee joint during a walking cycle is between 0° and 60° [20] and the obtained SMA results completely placed in this range.

Through the development of the SMA wide range of verification tests has been taken for different markers' color, subject suit color, light intensity and background color and appropriate features are included in the interface of the SMA that communicate with the user through the process to handle the wide variety of testing conditions. It also enables the user to monitor the process and interferes in order to make essential corrections.

Another important application of SMA is in sport biomechanics. We used the system to compare some Kata techniques

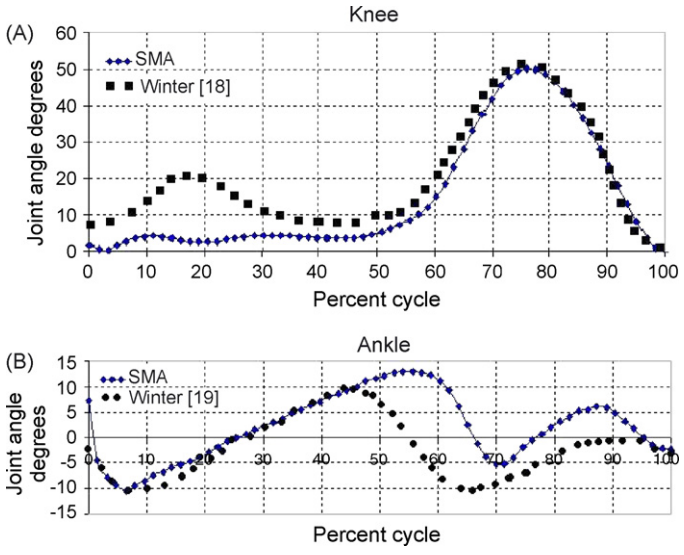

Fig. 6. Comparison of (A) knee and (B) ankle flexions for gait cycle with other invasigators [18,19].

of a group of Iranian national Karate team members and non-expert Karate Kas in order to have a good reference for instructors and new Karate Kas. Here the results of comparison of expert and non-expert Karate Ka have been presented.

One of the fundamental techniques of Karate (Zenkoutso Dachi) was considered to get insight into kinematic discrepancies between the groups mentioned above. Zenkoutso Dachi is one the most important techniques in Karate and there are many other techniques that are performed while standing on it.

To perform it the front shank is right to horizon and the other shank makes an angle that undergoes almost 40% of the body weight. When Karate Ka moves forward two shanks change their positions (Fig. 7).

There are many factors to perform this technique correctly that most of them have mechanical backgrounds. First of all, the foot and ankle should be studied so that they would not rise too much. Shoulder's vertical displacement should be taken into account to observe the displacement of the center of mass. Shank angle of the moving leg with the motion direction (Y) should be studied to demonstrate the pattern of the shank movement. Finally, variation of knee angle during performing the technique to know more about how knee joint moves should be considered. Finally, hip and trunk angles were measured to know if the Karate Ka is using his hip and trunk movement or just fixing it.

It can be seen that this technique is using mostly one side of the body to be performed. Due to this fact we considered the right side of the Karate Ka. To get the kinematic results four segments were taken. These were trunk, thigh, shank and foot segments. Also arms were free to move and quite folded due to their negligible effects on this performance and our experiment.

Anatomical land markers were used to position markers on these segments: right acromion process (shoulder joint), greater trochanter (hip joint), lateral femoral epicondyle (knee joint), lateral malleolus (ankle joint), the second metatarsal head. In addition, 4 other markers were positioned on segments between two markers on every segment but not in a same plane.

Fig. 8 shows vertical displacement of metatarsal in Zekoutso Dachi for an expert and a non-expert Karate Ka. The difference

Fig. 7. Markers positions on each segment and different stages of Zenkoutso Dachi during experiment.

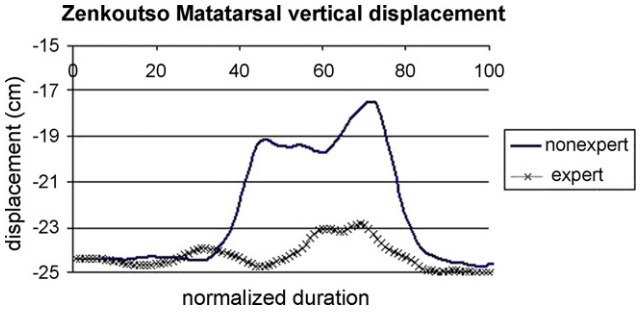

Fig. 8. Metatarsal vertical displacement vs. normalized duration in Zenkoutso Dachi for an expert and a non-expert Karate Kas.

starts from about 40% of the normalized duration till near the end of the movement and we can see that this range is almost the same for both Karate Ka. Vertical displacement of metatarsal for the expert is considerably lower than that of the non-expert one. This can be explained by the difference between their strength. Non-expert karate ka's weaker muscles in the left leg cause him to raise his moving leg more than the expert. So the expert's metatarsal tries to move nearer to the ground than the non-expert karate ka does. Although there are similarities between the pattern of their performance.

Vertical displacement of ankle's marker versus normalized duration for both people is compared in Fig. 9. At the beginning of their movement it is seen that both have raised their ankles. This is 10 cm for the expert and 15 cm for the non-expert karate ka. Finally, they end their technique at almost the same level. Comparing these displacements during whole time shows that the expert can move his ankle nearer to the ground than the non-expert one. It is because of their last differences between their metatarsal displacements. It seems that ankle angle for the expert is less that than of the non-expert one so it makes his ankle move nearer to the ground. Again it is observed that their curves have quite similar pattern.

In Fig. 10, trunk's mean angle of two groups of experts (9 people) and non-expert Karate Kas (7 people) are compared. This angle is trunk's angle with respect to horizon in the direction of movement. When both groups start their techniques, their trunk is somehow right to the floor. For all experts this mean angle is almost 87°, but for non-expert ones it is near 85° which shows that experts have tried to be more erected than the other group. Also in the beginning, non-expert Karate Kas inclined more than experts. Experts incline about 2° while non-expert ones inclined about 4°. The statistical

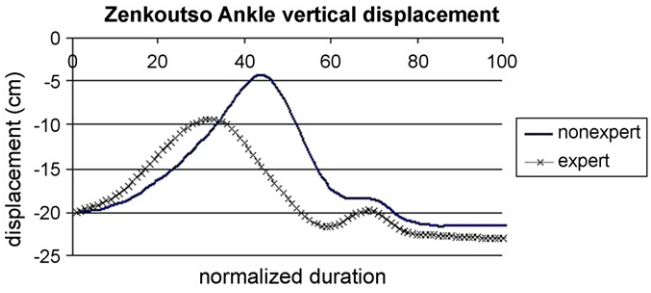

Fig. 9. Ankle vertical displacement vs. normalized duration in Zenkoutso Dachi for an expert and a non-expert Karate Kas.

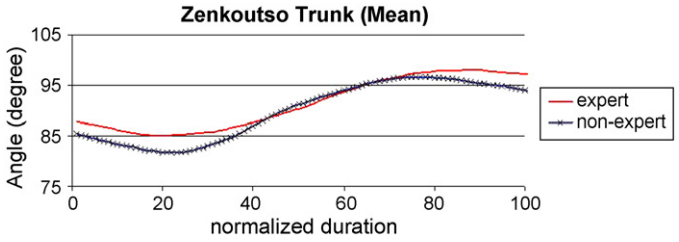

Fig. 10. Trunk angle (mean) vs. normalized duration in Zenkoutso Dachi for a group of experts and non-expert Karate Kas.

experiment of these two groups showed that there is a meaningful difference between the way Zenkoutso Dachi is performed by the experts and non-expert ones. ($\alpha = 0.025$).

In conclusion, it seems that our developed software is able to show the discrepancies between groups of non-expert and expert Karate kas that was our aim to be satisfied. These results and some other validation results that were performed to verify our software lead the SMA to be acceptable software to analyze any other human body's movements.

6. Conclusion

A general framework for a marker-based human motion tracking system has been developed. The proposed differential algorithm for image processing and an optimized MDLT method for 3D coordinates reconstruction are presented. The results showed that, in spite of its simplicity, the developed system works in a wide range of practical situations, such as different marker colors, subject's suit color, light intensity and even the background colors.

This work has mainly focused on using a low cost hardware and precise built-in units for motion analysis applications. Beside that, its precision and capability of capturing from any number of cameras to increase the domain of operation has made the proposed method a reliable approach for real-time industrial applications, such as animation production and special effects. Currently, the developed system is widely used for biomechanic analysis of human motions in Biomechanics Laboratory of Sharif University of Technology [21].

Based on the presented experimental analysis for Karate-Kas, we quantifiably discussed the motion analysis for a small number of subjects. However further works are required to perform a more precise and exact motion analysis for a larger number of subjects to extent the system application for training of the athletes.

Acknowledgements

This study was financially supported by a grant devoted to the development of Sharif Motion Analyzer (SMA) by Industrial Development and Innovation Organization. The authors would like to thank Dr. S. Kassaee and Dr. E. Fatemizadeh from Computer and Electrical Engineering Departments of Sharif University of Technology, for the fruitful discussions and valuable comments. We would also like to thank the Mathworks Ltd. for accepting to publish a demo version of this software on their website.

References

[1] D.J. Sturman, A brief history of motion capture for computer character animation, Character Motion Systems, SIGGRAPH 94: Course 9.
[2] J.A. DeLisa, Gait Analysis in the Science of Rehabilitation, Veterans Health Administration. Scientific and Technical Publications Section, Washington, DC, United States, 1998.
[3] J. Rose, J.G. Gamble, Human Walking, second ed., Williams & Wilkins, 1994.
[4] M. Furniss, Motion Capture, MIT Communication Forum, 1999.
[5] C. Grow, I. Gordon, R.D. Stuart, A. Adalja, Motion Capture as a Means for Data Acquisition, ARA Focus Group Spring, 1998.
[6] C.-L. Huang, C.-Y. Chung, A real-time model-based human motion tracking and analysis for human-computer interface systems, EURASIP J. Appl. Signal Process. 2004 (11) (2004) 1648–1662.
[7] MATAB Documentation, Mathworks Co., Version 7.0.4, 2005.
[8] Sugiyama, Nakamura, A method of de-interlacing with motion compensated interpolation, IEEE Trans. Consumer Electron. 45 (3) (1999) 611–616.
[9] C. Vaughan, B.L. Daris, in: J.C. O'Connor (Ed.), Dynamics of Human Gait, Kiboho Publishers, University of Cape Town, South Africa, 1999.
[10] Y.I. Abdel-Aziz, H.M. Karara, Direct linear transformation from comparator coordinates into object space coordinates in close-range photogrammetry, in: Proceedings of the Symposium on Close-Range Photogrammetry, American Society of Photogrammetry, (1971), pp. 1–18.
[11] H. Hatze, High-precision three-dimensional photogrammetric calibration and object space reconstruction using a modified DLT-approach, J. Biomech. 21 (1988) 533–538.
[12] 3dS MAX L&T CD., MAX User Reference, Auto Desk Co., Version 6.0, 2003.
[13] R. Shapiro, Direct linear transformation method for three-dimensional cinematography, Res. Q. 49 (2) (1978) 197–205.
[14] L. Chen, C.W. Armstrong, D.D. Raftopoulos, An investigation on the accuracy of three-dimensional space reconstruction using the direct linear transformation technique, J. Biomech. 27 (1994) 493–500.
[15] G.A. Wood, R.N. Marshall, The accuracy of DLT extrapolation in three-dimensional film analysis, J. Biomech. 19 (9) (1986) 781–785.
[16] H.H. Mabie, H. Reinholts, Mechanisms and Dynamics of Machinery, fourth ed., John Wiley and Sons, C.F., 1987.
[17] A.F. Dsoza, V.K. Garg, Advanced Dynamics Modeling and Analysis, Prentice Hall Inc., Englewood Cliffs, NJ 07632, 1984.
[18] D.A. Winter, The Biomechanics and Motor Control of Human Gait: Normal, Elderly and Pathological, second ed., Waterloo Biomechanics, CA, 1991.
[19] M.W. Whittle, Gait Analysis: An Introduction, second ed., Butterworth-Heinemann Ltd., Oxford/Boston, USA, 1993.
[20] M. Nordin, V.H. Frankel, Basic Biomechanics of the Musculoskeletal System, third ed., Lippincott Williams & Wilkins, 2001.
[21] A. Kolahi, M. Hoviattalab, T. Rezaeian, M. Alizadeh, M. Bostan, Implementation of optical tracker system for marker-based human motion tracking, in: International Conference of Applied Simulation and Modeling, IASTED, Greece, June, (2006), pp. 26–28.