Hand Pose Tacking for 3-D Mouse Emulation

P. Nesi and A. Del Bimbo
Department of Systems and Informatics, Faculty of Engineering
University of Florence, Via S. Marta 3, 50139 Florence, Italy
Tel.: +39-55-4796265, Fax.:+39-55-4796363, NESI@INGFI1.ING.UNIFI.IT

Abstract

Recently, several devices for improving the human-computer interaction quality and efficiency have been proposed. The greatest part of the most diffuses devices must be worn by the user – e.g., gloves for hand-tracking, complete or partial tracksuits. The adoption of such devices is unsuitable for many applications since they are too much constrictive – e.g., teleconference. For this reason, other simpler devices such as 3-D mice have been suggested. Most of the 3-D mice proposed in the literature compel the user to be connected to the device and their behavior is much closer to a Joystick rather than to a true 3-D mouse. More recently, vision-based hand tracking systems have been presented, but most of them constrain the user to wear gloves, bracelets or to mark hands in some points. In this paper, a vision-based, glove- and mark-free hand tracking system is proposed. This is based on stereo vision and can be implemented with low-cost architectures owing to its low computational complexity.

1 Introduction

In the last years, a great improvement in the human-computer interaction quality was given by the virtual reality techniques. In realistic virtual environments, the user has to move his/her body in the real environment according to the movements that should be performed in the virtual environment in order to move her/his-self copy in the virtual space. In typical applications of virtual reality gloves endowed of sensors are used for communicating hand movements to the computer – e.g., VPL Research's DataGlove, Exos's Dexterous Hand Master, Z-Glove, Mattel's Power Glove [19], [7]. By using such devices, it is possible to receive the hand *absolute position* and *orientation* as well as to get information about the hand and finger posture. The hand orientation and absolute position can be estimated by using coil-based (e.g., Polhemus System) or ultrasonic-based mechanisms, while optical fibres or tensors are used for measuring the bending of fingers. By analyzing the trend of signals it is possible to recognize: (i) the hand position/orientation, and (ii) the bending of finger, hand gestures and postures – e.g., [15], [17], [13]. Specific semantics can be associated with hand gestures and/or postures – for example for commands: *go-ahead*, *stop*, *zoom-in*, etc.

In general, the adoption of a hand-tracking device is very useful in several different fields: telerobotics [9], manipulation of 3-D objects for image query [6], sign language interpretation [17], teleconferences, etc. In some of the above-mentioned applications, the presence of the synthetic reproduction of the hand in the virtual space is mandatory (e.g., for manipulating virtual objects, or telerobotics), while in other applications this is completely unuseful (e.g., for navigating in a virtual environment, for drawing in a 3-D space). In these latter cases, a glove endowed of sensors is too complex and expensive to be used. In fact, for these cases, a sort of a 3-D mouse endowed of 1-2 buttons and capable of measuring hand motions can be more suitable and less binding – e.g., based on Polhemus transducer [18]. In order to address this problem the so-called Spaceball, consisting of a ball connected to a support has been presented. For the same purpose, a device consisting of a cube frame with a sphere or another object (connected to the vertices by links and tensors) in the core has been proposed. In both these devices, the user has to force the motion of such an object in order to transmit the corresponding directions of the hand motion to the computer. On the other hand, the behavior of these two devices is much more similar to a 3-D joystick rather than to a 3-D mouse. It should be noted that, with the above-mentioned 3-D mice, the user is constrained to be connected with the computer, through a cable or to grasp a ball or an object in a cube.

From the point of view of user movements, the vision-based approaches are less constrictive. In the literature, several vision-based systems for hand pose tracking have been presented, where most of them are based on stereo vision – e.g., [12], [4], [10], [5], [16]. In order to make the hand pose tracking easier some simplifications have been introduced: in [12] the user has to wear a special bracelet marked with specific drawings, in [4] 4 colour markers must be placed on the hand for hand-tracking; in [8] the user must use only his/her hand-index for indicating the position of the locator on the screen; in [5] each finger is marked with strips usually placed on a glove that the user has to wear. Most of the above-mentioned vision-based hand pose tracking systems are also computationally very expensive to be adopted for defining a low-cost 3-D mouse.

In this paper, a vision-based system for hand pose tracking is presented. The system proposed can be profitably used for implementing a low-cost vision-based 3-D mouse device, since it will be demonstrated that the approach can be used even by using images with low resolution. In addition, it doesn't constrain the user to wear a glove or something of equivalent, or to mark hand fingers and/or particular points of the hand.

2 Modeling a Vision-Based 3-D Mouse

The mouse movements are used by the computer for moving a locator on the screen (e.g., an arrow, a cross...), while program actions are associated with button transitions. It should be noted that the mouse is a relative transducer of hand pose, since its position is only relative and not absolutely measured with respect to the position of the hand on the table. This feature improves the mouse capabilities, since allows the extension of the ranges of mouse motions. In fact, the user can simply take off the mouse for replacing it in a more comfortable position without producing a change in the locator position on the screen. For example, a button can be used for switching off the motion tracking, thus implementing a 3-D mouse which is fully consistent with the mechanisms and concepts of classical 2-D mice. In order to solve the hand pose tracking problem, a vision-based technique for hand pose estimation and motion analysis is needed, while for the second point a very restricted number of hand postures can be recognized by means of simple image processing techniques and profitably adopted as mouse buttons.

2.1 Image acquisition system and pose estimation

The techniques for 3-D motion estimation in vision are mainly based on the estimation of the apparent motion projected on the image plane – e.g., [14], [11], [2]. If a monocular image sequence of the moving object is available, the motion can be easily estimated. On the other hand, the 3-D motion cannot be easily recovered by inverting the perspective projection law [1]. In effect, in such conditions the motion estimation can be safely obtained only under particular conditions. A more robust approach is based on stereo vision. The estimation of the hand pose at each time instant can be the first phase of a system for implementing a 3-D mouse. If only simple postures are used for emulating the mouse buttons (for example, by considering only postures with fingers close to each other) the exact position of each finger can be neglected.

The approach proposed is based on stereo vision as depicted in Fig.1. These two cameras are called G and B, respectively. A point $\mathbf{P}$ in the 3-D space projects the points, p and p', which have different positions on the images G and B. By considering the system geometry and the projections of a point, $\mathbf{P}$, in the 3-D space in both images, then the position of $\mathbf{P}$ in the 3-D space can be estimated. By considering as primary the system of coordinates (O, X, Y, Z) associated with camera G, and the geometrical relationships between the two cameras, the transformation is estimated. The system of coordinates B is translated in O' whose coordinates are (X_o, Y_o, Z_o) in $OXYZ$, and rotated of 90 degrees around the Z'-axis and of α around the X'-axis (note that X- and Z-axes, and Y'- and Z'-axes, belong to the same plane). Therefore, the transformation results to be:

$$D = \begin{bmatrix} 0 & 1 & 0 & -Y_o \\ -C(\alpha) & 0 & -S(\alpha) & X_oC(\alpha)+Z_oS(\alpha) \\ -S(\alpha) & 0 & C(\alpha) & X_oS(\alpha)-Z_oC(\alpha) \\ 0 & 0 & 0 & 1 \end{bmatrix}. \tag{1}$$

From this transformation an expression for Z can be obtained:

$$Z = \frac{X_oll'C(\alpha)+Z_oll'S(\alpha)-X_oly'S(\alpha)+Z_oly'C(\alpha)}{xl'C(\alpha)-y'xS(\alpha)+y'xC(\alpha)+ll'S(\alpha)}. \tag{2}$$

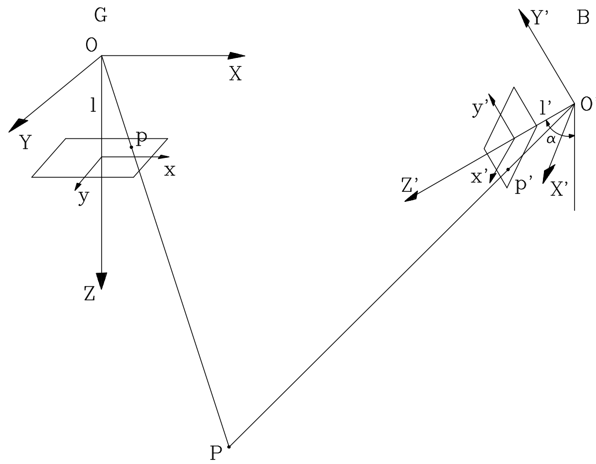

Figure 1: The geometry of the acquisition system.

By substituting this value into the equations of perspective projection, the coordinates of point $\mathbf{P} = (X, Y, Z)$ in the 3-D space can be estimated.

2.2 Hand postures as button switches

In order to emulate the presence of buttons and/or for switching off the transmission of hand movements (i.e., take off the mouse), two independent hand postures have been defined which can be easily identified with respect to the normal posture. In Fig.2a, the normal hand posture is reported as seen through both cameras[1]. In Fig.2b and 2c the hand postures used as buttons are presented: the *hand rotated* (HR) and the *hand closed* (HC). It should be noted that the hand-tracking process must be capable of measuring the hand position in the 3-D space even if the hand: (i) changes its posture in time, for example during the changing of posture for "pressing a button"; (ii) presents a difference in size, as it usually happens when the hand moves along the Z-axis and/or when hands of different users are considered.

3 Estimation Process

To avoid the procedure of device setup or adjustment with respect to the user's hand (usually needed in glove-based systems) the estimation process should be independent of the hand dimensions. In order

[1] The names of cameras derive from the fact that an acquisition board for color images has been used as a stereo image grabber. Cameras G and B have been connected with the green and blue channels of the acquisition board, respectively (8 bits/pixel of resolution for each image channel). This mechanism is feasible only if the cameras can be synchronized to each other. In order to analyze the image sequence off-line (to repeat the same experiment at different resolutions, with different techniques, with different sampling rates, etc..), the G and B signals have been combined in a video-composite and recorded on video tape. This process has rendered difficult the full separation of the two signals. In fact, the images produced by the B camera also present a light shade of the image G. On the other hand, as will be shown later, this effect is completely ininfluent for the algorithm behavior.

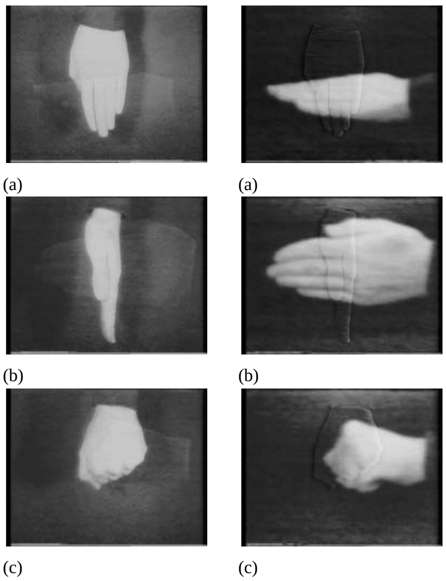

Figure 2: Hand postures as observed through cameras G (left side) and B (right side): (a) normal, HN, (b) rotated, HR, (c) closed, HC.

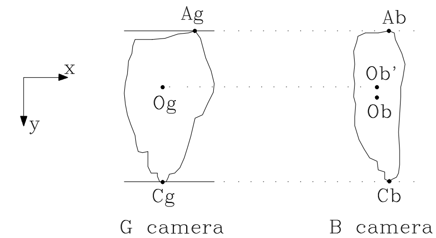

Figure 3: Centers of gravity, hand in the normal position as observed through G and B cameras.

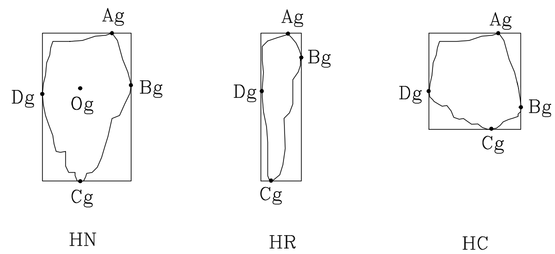

Figure 4: Shapes of the possible hand postures observed through the G camera: HN, normal; HR, rotated; HC, closed, with features points.

to satisfy these requirements, a specific estimation process has been adopted. The images are grabbed and binaryzed by using a threshold T_b depending on the mean value of the illumination. The binaryzed images are segmented in order to identify the hand shape in both views. As previously discussed, for the hand pose tracking the center of gravity is considered. In Fig.3, the mechanism for estimating the center of gravity of the hand shape into image B is summarized. In order to find in the hand shape of image B the corresponding coordinates of the center of gravity identified in image G, Og, a particular process has been adopted. Firstly, the centers of gravity, Og and Ob, of the hand shapes of both the images are estimated. Due to the different distances between the hand and the cameras the lengths of the hand shape ($\|Ob_y - Ab_y\| \neq \|Og_y - Ag_y\|$) are different in the views. In order to simplify the estimation process, the corresponding center of gravity of image B, Ob', is approximately considered as a point belonging to the projection of Og and to the axis passing through Ob; thus, $Ob'_x = Obx$ and $Ob'_y = Ab_y + (Ab_y - Cb_y)\frac{(Og_y - Ag_y)}{(Ag_y - Cg_y)}$. The measures of the positions of points: $Og = (Og_x, Og_y) = (x, y)$ and $Ob' = (Ob'_x, Ob'_y) = (x', y')$ are placed into equations (2) for estimating the position of the center of gravity in the 3-D space and, hence, the hand pose. The method proposed for the estimation of the center of gravity is quite robust with respect to changes in the hand shape. This also allows the adoption by the user of a relaxed hand posture during the use of the 3-D mouse proposed; thus, the use of this 3-D mouse becomes more comfortable.

The above discussed process for estimating the center of gravity of the hand is obviously an approximate technique, since it leads to optimal estimations only if fingers are moved together and the hand is maintained quite planar. On the other hand, the adoption of this approach keeps the computational cost very low, which has an asymptotical complexity equal to $N \times M$ where N, M are the image dimensions. Moreover, the latter steps can be performed during the binaryzation, while the coordinates of points A, B, C, D, and the centers of gravity Og and Ob, can be directly estimated. Therefore, the computational complexity of the algorithm proposed for hand pose estimation is an $O(NM)$.

For the recognition of hand postures, a technique based on the estimation of the ratio between the dimensions of the maximum enclosing rectangle has been adopted. Since the hand postures have been defined with respect to the system of coordinates G, only the corresponding images are considered for the estimation of the maximum enclosing rectangle. In Fig.4, the typical maximum enclosing rectangles obtained by considering the hand in the different postures are reported. The ratio, R_p, between the maximum length, M_l (e.g., $Ag_y - Cg_y$), and the maximum width, M_w (e.g., $Dg_x - Bg_x$) – i.e., $R_p = M_l / M_w$ – is independent of the distance from the hand and the camera G. The mechanism of posture recognition is simply based on a couple of thresholds T_r and T_c applied on the ratio R_p:

$$\text{Hand Posture} := \begin{cases} HR & \text{if } R_p > T_r \\ HC & \text{if } R_p < T_c \\ HN & \text{otherwise} \end{cases}$$

One of the proposed postures can be used for switching off the hand pose tracking in accordance with the behavior of traditional 2-D mice. The estimations which are needed for the hand postures recognition can be performed together with those which are needed for the hand pose estimation; thus, the global computational complexity remains an $O(NM)$.

During the hand-tracking, discontinuities in the position of the hand center of gravity could be observed when the hand changes its posture – e.g., passing from the normal position to that if hand closed,

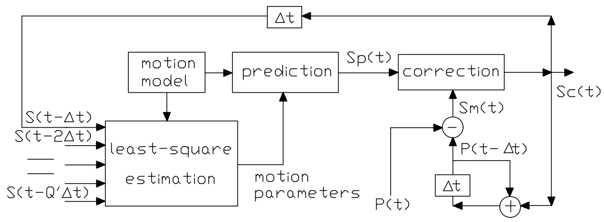

Figure 5: Prediction/correction approach applied on each hand pose component.

$HN \rightarrow HC$. On the other hand, these discontinuities are strongly attenuated in image sequences with low resolution. In order to compensate this effect, suitable offsets could be added, and these offsets should be estimated by measuring the hand that has to be tracked (i.e., during a calibration procedure). Moreover, the process adopted for estimating the center of gravity mapped on camera B has the attenuation of the discontinuities (due to changes in hand postures) as a side effect. Therefore, in our experiments no offset has been adopted. The robustness with respect to this type of discontinuities is also improved by the filtering stage discussed in the next subsection.

3.1 Motion smoothing

In order to avoid discontinuities due to noise and for improving the system robustness a phase of smoothing has been introduced after the hand pose estimation process. It has been obtained by defining an iterative process of prediction/correction. In Fig.5, the schema of the prediction/correction model and process is described. This technique is applied to each hand pose component, $\mathbf{P} = (X, Y, Z)$. The mathematical model adopted is polynomial such as in [3]:

$$Sp(t) = b_0 + b_1\ t + b_2\ t^2 + ... \qquad + b_n\ t^n + \sigma(t), \quad (3)$$

where: b_i for $i = 0, .., n$ are the *model parameters*, $\sigma(t)$ models the noise (e.g., Gaussian), and n is the order of the model. The motion model is defined by means of the polynomial interpolation by using the last Q points. For the polynomial model the parameters are also estimated by considering the values at the last Q time instants; thus, defining an over-determined system of Q equations in $n + 1$ unknowns. In the case of first-order model (i.e., $b_n = 0$ for $n > 1$), equation (3) becomes $S(t) = b_0 + b_1 t + \sigma(t)$. If the last Q time instants are considered, then an over-determined system of Q equations with b_0 and b_1 as unknowns is defined. Without loss of generality, Δt can be considered constant with respect to time; hence, in the over-determined system of equations time t can be scaled with respect to the beginning of the temporal window comprised of Q time instants. Thus the Q equations assume the form:

$$S(g) = b_0 + b_1 q + \sigma(q) \qquad \text{for} \quad q = 0, ..., Q - 1.$$

The least-square estimation consists in finding the parameters b_0, b_1 which minimize:

$$\sum_{q=0}^{Q-1} \left(S(q) - \sigma(q) - b_0 - b_1 q\right)^2 .$$

In order to minimize the above expression, the derivative of the above equation with respect to b_0 and b_1 is taken. Hence, the derived equations are posed equal to zero, obtaining a determined system of equations. It should be noted that the matrix of coefficients of the system of equations depends only on the dimension of the temporal window Q. This allows to estimate this matrix only one time when the model dimensions are defined and not at each time instant, thus reducing the computational effort (see [3] for a comparison between the autoregressive and the polynomial model). The values of model parameters are used at each time instant in equation (3) to estimate the predicted value $Sp(t)$ according to Fig.5. As can be noted, the model parameters are estimated on the basis of the history of corrected estimations ($Sc(j)$ per $j = t - \Delta t, ..., t - Q\Delta t$). Model parameters are used to estimate the predicted displacement, $Sp(t)$, and this is compared with the measured, $Sm(t)$, in order to obtain the predicted/corrected displacement, $Sc(t)$. The correction is based on a heuristic considerations founded on the statistic behavior of predicted and measured displacements [3].

By using motion models with prediction/correction techniques for tracking objects, the noise in measurements is strongly reduced. Moreover, these techniques trim the motion components at high frequence. This effect is more evident when the dimension Q of the temporal window is increased, leading to a decrease in the bandwidth. In the case of tracking of positions (i.e., trajectories), the reduction of the high frequency components leads to smooth the curvature of moving object trajectory. This is compensated by the fact that when the trajectory presents high values of curvature (e.g., in the presence of a direction inversion) a higher number of points (corresponding to hand positions) are received since the hand has usually a low velocity.

4 Experimental Results

In order to validate the approach proposed for hand pose tracking with posture recognition as mouse buttons, several real image sequences at different resolutions have been used. The geometry of the image acquisition system adopted has the following features: $O' = (X_o, Y_o, Z_o) = (1048, 0, 0)$ pixels, $\alpha = 65$ degrees, $l = 1097$ pixels, $l' = 3842$ pixels (the pixel dimension has been normalized in both cameras). It should be noted that two different cameras have been used, on which a previous calibration was performed (by considering lens used and their set) in order to get accurate camera parameters.

In Fig.6, the mean values of the relative errors, $\mathcal{E}r$, which have been obtained by using the proposed technique for estimating the hand pose as a function of the image resolution, are reported. The mean values have been e estimated by considering 400 images (i.e., $s = 400$) belonging to different image sequences, where the hand has been moved in a range equal to ± 100 for the X-axis, ± 45 for the Y-axis, and $400 \div 540$ for the Z-axis. By observing the above-mentioned figure it can be noted that the mean errors for the X and Y components increase their values with the decreasing of the image resolution, while the opposite trend is noticed for Z. This fact is due to the perspective projection law, where the coordinates on the image plane depend on the ratio between the X, Y values and Z.

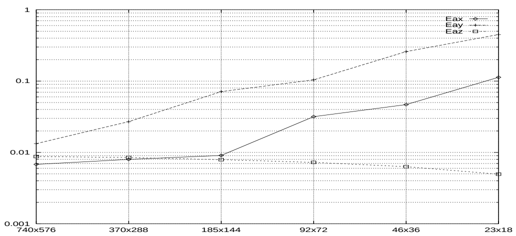

Figure 6: Mean values of the relative errors at different image resolutions.

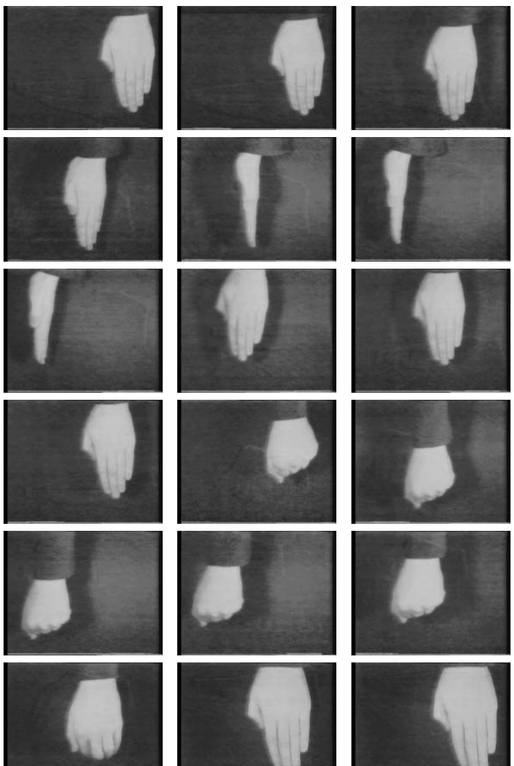

Figure 7: Example of hand-tracking with changes in hand postures (through camera G, 15 frames for seconds, step 4).

In Fig.7, a sequence of images presenting the typical hand behavior with changes of hand postures is shown. As regards this sequence, the behavior of 3-D components of the hand center of gravity measured with the proposed system as a function of the image resolution is depicted in Fig.8. From these figures it can be observed that very accurate estimations of the hand pose are obtained by using images at higher resolutions, while some errors are encountered when low image resolution is adopted – see for example 46×36 and 23×28.

In Fig.9, the results produced by the posture recognition algorithm at different image resolution (from 740×576 to 23×18) are reported. In our experiments, the thresholds for identifying the hand postures have been assumed to be: $T_r = 2.7$, and $T_c = 1.3$. As can be observed from these curves, for the two smaller resolutions manifest errors have been obtained, while at the higher resolutions correct posture recognitions

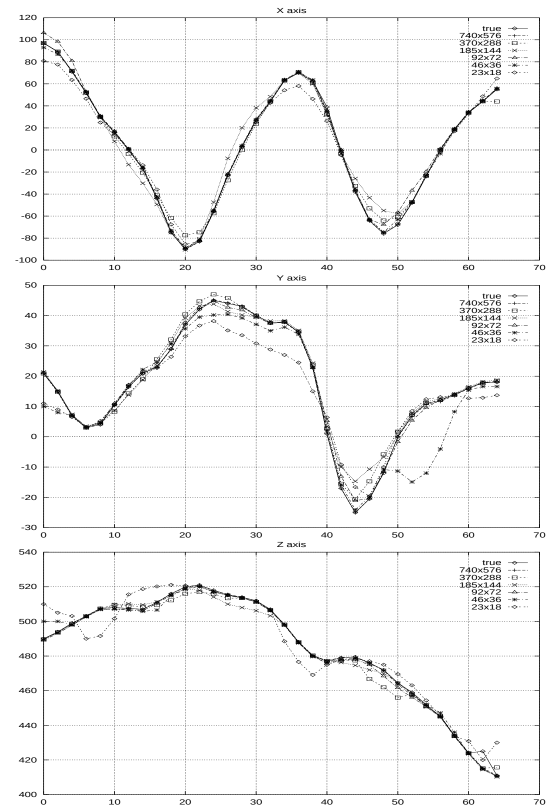

Figure 8: Behavior of the hand 3-D pose components with respect to image resolution (from 740×576 to 23×18 pixels).

have been performed. By comparing the patterns of motion components of the hand pose in time (see Fig.8) with those of the hand posture, it can be observed that no discontinuities are present in correspondence of changes in hand postures. This confirms the robustness of the algorithm with respect to the hand shape as previously discussed.

In Fig.10, the performance achievable by implementing the above algorithm for hand pose tracking and postures recognition on a 486 DX 33 MHz is reported for reference. The performance trend confirms that the algorithm complexity is an $O(NM)$. As can be noted at 92×72 pixels of image resolution 15 estimations per second are performed.

The above experiments have demonstrated that the approach proposed for the hand pose estimation is very robust with respect to noise. It should be noted that a 3-D mouse is usually adopted in a system where the user has a direct feedback about the hand movements and changes of status. In our experiments, by using the system proposed at low resolution, it has been observed that the accuracy obtained largely guarantees a robust hand pose tracking. In fact, as it can be observed from the above mentioned experiments good results can be obtained even at the lowest image resolution.

By observing Fig.6 and the differences among the patterns obtained at different image resolutions it can be noted that the best compromise between accuracy and image resolution (i.e., computational cost) is obtained by using 92×72 pixels of image resolution. At

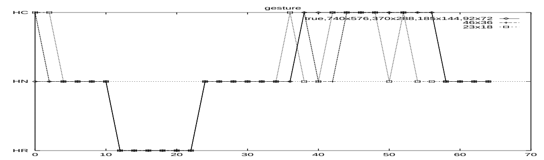

Figure 9: Results produced by the posture recognition algorithm with respect to image resolution (from 740×576 to 23×18 pixels). A positive and the negative value for the signal marks the detection of the HC and HR postures, respectively.

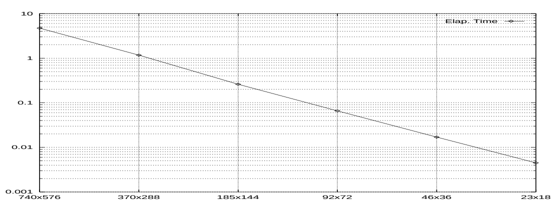

Figure 10: Performance of the algorithm proposed on a 486 DX 33 MHz based machine as a function of image resolution.

this image resolution, even the algorithm for recognizing the hand postures gives good results (see Fig.9) and, thus, the whole process of estimation and recognition can be executed 15 times per second on a non-dedicated machine.

5 Conclusions

A robust and computationally cheeper vision-based 3-D mouse was proposed. This is based on a method for hand pose tracking and gesture recognition in the 3-D space by using stereo vision. The main feature of this technique consists in the fact that the hand pose is estimated at each time instant by evaluation only the center of gravity, while hand posture recognition is transformed in a mono-dimensional problem. Hand pose tracking is very robust since a prediction/correction mechanism was used. This approach allows the adoption of the technique proposed even at low image resolution, as demonstrated by tests carried out on several real image sequences at different image resolutions. In addition, the performance on a reference machine with respect to image resolution was also evaluated. The results have showed that this technique is suitable to be used as a basis for the implementation of a dedicated and low cost 3-D mouse device.

Acknowledgments

The authors would like to thank S. Fantini for his cooperation in performing the early experiments.

References

[1] G. Adiv. Determining three-dimensional motion and structure from optical flow generated by several moving objects. *IEEE Trans. on Pat. Ana. and Mac. Intel.*, 7(4):384–401, 1985.

[2] D. Ben-Tzvi, A. DelBimbo, and P. Nesi. Optical flow from constraint lines parametrization. *Pattern Recognition*, 26(10):1549–1561, 1993.

[3] A. Borri, G. Bucci, and P. Nesi. A robust tracking of 3d motion. In *Proc. of the European Conference on Computer Vision*, pp.181–188, Stockholm, Sweden, 1994.

[4] R. Cipolla, Y. Okamoto, and Y. Kuno. Qualitative visual interpretation of 3d hand gesture using motion parallax. In *Proc. of the Int. Workshop on Machine Vision Applic. Tokyo*, pp.477–482, 1992.

[5] J. Davis and M. Shah. Visual gesture recognition. *IEE Proceedings*, 141(2):101–106, 1994.

[6] A. DelBimbo, M. Campanai, and P. Nesi. 3-d visual environment for image database querying. *IEEE Trans. on Soft. Eng.*, 19(10):997–1011, 1993.

[7] H. Eglowstein. Reach out and touch your data. *Byte*, pp.283–290, 1990.

[8] M. Fukumoto, K. Mase, and Y. Suenaga. Real-time detecting of pointing actions for a glove-free interface. In *Proc. of the Int. Workshop on Machine Vision Applic. Tokyo*, pp.437–476, 1992.

[9] G. G. Grinstein, M. T. Maybury, and R. B. Mitchell. Intelligent virtual interface for telerobotics. *SPIE Cooperative Intelligent Robotics in Space III*, 1829:401–408, 1992.

[10] K. Ishibuchi, H. Takemura, and F. Kishino. Real-time hand gesture recognition using 3d prediction model. In *Proc. of the 1993 IEEE Systems, Man and Cybernetics Conference, Le Touquet, France*, volume 5, pp.324–328, 1993.

[11] P. Nesi. Variational approach for optical flow estimation managing discontinuities. *Image and Vision Computing*, 11(7):419–439, 1993.

[12] P. O'Neill, K. Dwason, and D. Vernon. An optical hand pose tracking system. In *Proc. of the 2nd Int. Conf. Dedicated to Image Communication*, pp.175–180, Bordeaux, France, 1993.

[13] M. J. Papper and M. A. Gigante. Using gesture to control a virtual arm. In *Virtual Reality Systems, Cap.16*, pages 237–246. Academic Press, 1993.

[14] K. Prazdny. On the information in optical flows. *CVGIP*, 23:239–259, 1983.

[15] D. L. Quam. Gesture recognition with a dataglove. In *Proc. of the IEEE Nat. Aerospace and Elect. Conf.*, pp.755–760, Piscalaway, NJ, USA, 1990.

[16] J. M. Rehg and T. Kanade. Visual tracking of high DOF articulated structures: an application to human hand tracking. In *Proc. of the European Conf. on Computer Vision*, pp.35–46, Stockholm, Sweden, 1994.

[17] T. Takahashi and F. Kishino. A hand gesture recognition method and its application. *Systems and Computers in Japan*, 23(3):38–48, 1992.

[18] C. Ware and D. R. Jessome. Using the bat: a six-dimensional mouse for object placement. *IEEE CG & A*, 8(6):65–70, 1988.

[19] T. G. Zimmerman and J. Lanier. A hand gesture interface device. In *Proc. of Human Factors in Computing Systems and Graphics Interfaces*, pp.189–192, Toronto, Ontario, USA, 1987.