ELECTRONICS AND ELECTRICAL ENGINEERING
ISSN 1392 – 1215 2006. Nr. 7(71)
ELEKTRONIKA IR ELEKTROTECHNIKA

T 111 IMAGE TECHNOLOGY
VAIZDŲ TECHNOLOGIJA

Detection of Hand Position using 3-D Computer Vision

P. Serafinavičius
Department of Electronics Engineering, Kaunas University of Technology
50 Studentų str., Kaunas, Lithuania, phone: +370 614 90990, e-mail: paulius.se@gmail.com
G. Daunys
Department of Electronics, Šiauliai University
141 Vilniaus str., Šiauliai, Lithuania, phone: +370 41 595835, e-mail: g.daunys@tf.su.lt

Introduction

Today in computer science the special care is dedicated to the research and development of 3-D computer vision based systems for identification of various objects [1]. One of the most important fields of 3-D object identification applications is the non-intrusive human-computer interfaces, enabling user to interact with a system more efficiently.

Interfaces usually enable a bidirectional communication. Input devices allow users to issue commands to the system. Output devices provide users with both responses to commands and feedback about user action.

Conventional input devices, such as keyboard and mouse, are often used in standard interfaces. They have the advantage of working equally efficient in most working places, but they physically limit the mobility of the user.

Advanced input devices, such as data helmets and glasses, data gloves, body markers and smart card technologies ensure higher user mobility. They can be effective in sophisticated interaction, more close to user experience in the real world. The main disadvantages of them are intrusiveness and forcing the user to adapt new instruments and sensors. This usually results in comfortless interaction with a system.

Non-intrusive input devices are based on non-contact sensors such as cameras, microphones. User can interact through hand gestures or voice commands. Hand gestures provide natural and device-free means for human-computer interactions. Many applications can be controlled more intuitively with hand gestures rather than traditional devices like the mouse and the keyboard. In addition to being intuitive, hand gestures also offer higher dimensionality [2]. In some vision-based interaction approaches the aim is to derive a semantic interpretation of human hand gestures and facial expressions. Other approaches are based on full body, hand, head or eye motion. In this category computer-vision based hand pointing systems are more useful in human-computer interfaces, because hand pointing is an everyday life operation which does not require any special training.

There are many researchers who develop and investigate non-intrusive human-computer interfaces. One of them was implemented in the Museum of Palazzo Medici Riccardi of Florence. Users can ask the system to display on the screen additional information about specific parts of the paintings by pointing at the screen locations of interest [3].

In this paper we propose the conception of hand pointing system based on tracking feature points of 3-D objects in real time. The system we implemented is based on Intel's open source Computer Vision Library and two inexpensive web cameras.

Theoretical background

The main interaction elements of hand pointing system are two cameras placed so that the user's pointing gesture would be in view. A computer performs image analysis and computes the pointing gesture direction from the projection points in two cameras' images. We use pinhole camera model and triangulation for computing the 3D position of points in the images for the 3D reconstruction [4].

The 3D position of any point in world coordinates can be reconstructed from the perspective projections of these points on the image planes of the cameras, once the relative position and orientation of the two cameras are known. We choose the 3D world reference system to be the left camera reference system. The right camera is translated and rotated with respect from the left one; therefore six parameters describe this transformation.

The simplest case arises when the optical axes of two cameras are parallel, and the translation of the right camera is only along the X axis. Let us consider the optical setting in standard model (Fig. 1):

1. L and R are two pinhole cameras with parallel optical axes. Let f be the focal length of both cameras.

2. The baseline (that is the line connecting the two lens centers) is perpendicular to the optical axes. Let b be the distance between the two lens optical centers.

3. XZ is the plane where the optical axes lie, XY plane is parallel to the image plane of both cameras, X axis equals the baseline and the origin of (X, Y, Z) world reference system is the lens center of the left camera.

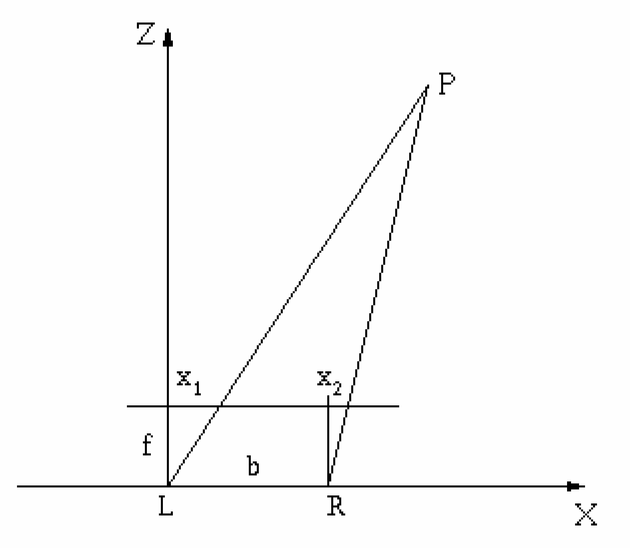

Fig. 1. Standard case model for parallel cameras

In this model equations of stereo triangulation are:

$$Z = b\frac{f}{x_1 - x_2}; \tag{1}$$

$$X = x_1\frac{Z}{f}; \tag{2}$$

$$Y = y_1\frac{Z}{f}. \tag{3}$$

Assuming that, we have chosen parallel cameras optical setup, we derived model of measuring the angle between two different direction lines of arm (Fig. 2). We added line OP_1, which corresponds to the length of human arm, where O is the base point (point of human arm's elbow); P marks the end of pointing finger at initial position; P_1 marks the end of the pointing finger after changing arm direction by angle γ in reference of line OP. The angle γ as shown in Fig. 2 is parallel to plane ZX. In general case it is not necessary. The plane, where measuring angle lies on, can be oriented in any other way.

Equations for calculating world coordinates of point P (initial position of pointing finger end):

$$Z_P = b\frac{f}{x_{PL} - x_{PR}}; \tag{4}$$

$$X_P = x_{PL}\frac{Z_P}{f}; \tag{5}$$

$$Y_P = y_{PL}\frac{Z_P}{f}. \tag{6}$$

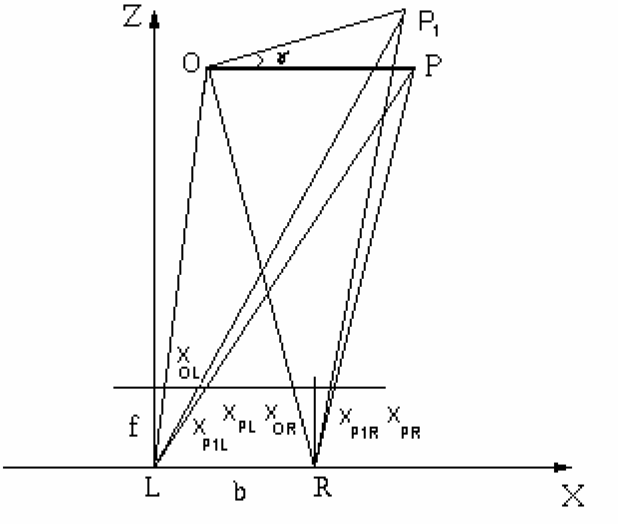

Fig. 2. Geometrical model for determination of human arm direction

where x_{PL} is P projection point coordinate in left camera image, x_{PL} – respectively in right camera image, y_{PL} – coordinate of P projection in Y plane, b – Euclidean distance between two cameras optical centers, f – focal length of camera's lens. Equations for calculating the world coordinates of point O (reference point of an elbow):

$$Z_O = b\frac{f}{x_{OL} - x_{OR}}; \tag{7}$$

$$X_O = x_{OL}\frac{Z_O}{f}; \tag{8}$$

$$Y_O = y_{OL}\frac{Z_O}{f}. \tag{9}$$

Equations for calculating the world coordinates of point P_1 (position after changing arm direction):

$$Z_{P1} = b\frac{f}{x_{P1L} - x_{P1R}}; \tag{10}$$

$$X_{P1} = x_{P1L}\frac{Z_{P1}}{f}; \tag{11}$$

$$Y_{P1} = y_{P1L}\frac{Z_{P1}}{f}. \tag{12}$$

Euclidean distances between world points from O to P and from P to P_1:

$$OP = \sqrt{(Z_P - Z_O)^2 + (X_P - X_O)^2 + (Y_P - Y_O)^2}; \tag{13}$$

$$P_1P = \sqrt{(Z_{P1} - Z_P)^2 + (X_{P1} - X_P)^2 + (Y_{P1} - Y_P)^2}. \tag{14}$$

The law of cosine can be applied to calculate the angle γ. Note, that $OP_1 = OP$. So, according to (13), (14) and the Law of cosine, the angle can be calculated by the following equation (in degrees)

$$\gamma = \arccos\left(1 - \frac{(P_1P)^2}{2(OP)^2}\right)\frac{180}{\pi}. \tag{15}$$

Intrinsic and extrinsic calibration of both cameras must be done in order to get focal length and Euclidean distance between two of them. The intrinsic camera parameters specify the camera characteristics properly; these parameters are [5]:

- focal length, that is, the distance between the camera lens and the image plane;
- location of the image center in pixel coordinates;
- effective pixel size;
- radial distortion coefficient of the lens.

The extrinsic camera parameters describe spatial relationship between the camera and the world; they are

- rotation matrix:

$$R = \begin{bmatrix} r_{11} & r_{12} & r_{13} \\ r_{21} & r_{22} & r_{23} \\ r_{31} & r_{32} & r_{33} \end{bmatrix}; \qquad (16)$$

- translation vector:

$$t = \begin{bmatrix} t_1 \\ t_2 \\ t_3 \end{bmatrix}. \qquad (17)$$

In our case Euclidean distance between two cameras can be calculated as follows

$$b = \sqrt{(t_{1L} - t_{1R})^2 + (t_{2L} - t_{2R})^2 + (t_{3L} - t_{3R})^2}\ , \qquad (18)$$

where t_{1L}, t_{2L}, t_{3L}, t_{1R}, t_{2R}, t_{3R} are translation vectors of left and right cameras respectively.

Experiments methodology

Special software for evaluating our approach was created using Intel's open source project - OpenCV library. The software allows:

- capturing video from two cameras;
- calibrating the cameras internally and externally;
- selecting the initial points of human arm;
- measuring the size of angle in reference with selected initial position in real time.

Currently, points O and P of the arm are not detected automatically. They should be selected by the user on the left camera image. When the user selects these points, automatic tracking of them starts in both left and right images.

Automatic tracking is based on OpenCV functions *cvGoodFeaturesToTrack* and *cvCalcOpticalFlowPyrLK.* The first one determines the strong corners, i.e., corners with big eigenvalues on left image. The function first calculates the minimal eigenvalue for every pixel of the source image and then performs non-maxima suppression. The next step is rejecting the corners with the minimal eigenvalue. Finally, the function ensures that all the corners found are distanced enough from one another by getting two strongest features and checking that the distance between the points is satisfactory. If not, the point is rejected.

In order to find corresponding points in the right image, calculation of optical flow is performed using iterative Lucas-Kanade method in pyramids. The function *cvCalcOpticalFlowPyrLK* implements sparse iterative version of Lucas-Kanade optical flow in pyramids. It calculates the optical flow between two images for the given set of points. The function finds the flow with sub-pixel accuracy. We apply the set of detected feature points on left image as the given set of points for the function.

Calibration of cameras and removing the optical distortion of images are also implemented using OpenCV library functions.

Procedure of determining the angle between arm positions using our software is as follows:

1. Do intrinsic calibration of left and right cameras separately. The focal length is calculated in this step.
2. Do extrinsic calibration of left and right cameras together. Euclidean distance between the cameras is calculated according to equation (18).
3. Select the reference point O on elbow of an arm on left camera image.
4. Select initial point P of pointing finger.
5. Change direction of arm, not changing position of elbow. The angle is calculated in reference of initial position on every frame refresh.

Technical equipment of evaluation system consists of notebook with 1,6 GHz CPU and two USB web cameras capturing video at 30 frames per second on VGA resolution.

Measurements of angle between different arm orientations were made. Angle between different arm directions were measured in 3 cases when arm angle orientation plane is parallel to the plane of:

1. ZX,
2. XY,
3. ZY.

Size of the angle was changed from 0^o to 50^o with step of 10^o.

Results

The results of the experiment are shown in tables 1-3. The first row of table describes the condition of evaluation according to 3 cases of arm orientations. All the cases evaluation was repeated 10 times and mean values of γ are shown here according to each change of angle step.

Table 1. Angle orientation plane is parallel to ZX

ZX	10^o	20^o	30^o	40^o	50^o
$\overline{\gamma}$, $(^o)$	8,8	17,5	24,6	32	37,5

Table 2. Angle orientation plane is parallel to XY

XY	10^o	20^o	30^o	40^o	50^o
$\overline{\gamma}$, $(^o)$	8,9	19,7	29,9	38,9	49

Table 3. Angle orientation plane is parallel to ZY

ZY	10^o	20^o	30^o	40^o	50^o
$\overline{\gamma}$, $(^o)$	7	17,1	24,1	27,9	37

Discussion and Conclusion

The software for detection of human arm position was created and evaluated experimentally. The highest accuracy of determining angle between different arm directions had been achieved when a plane of measuring angle is perpendicular to cameras' optical axes. The lowest accuracy had been achieved when a plane of measuring angle is parallel to cameras' optical axes. This happens due to not enough accurate determination of pointing finger Z coordinate. So, this approach is not suitable where high precision of 3-D points positions are required. It can be useful in commercial applications - non-intrusive input devices for computer and home environment control because it does not require expensive hardware and the software is implemented using open source library.

References

1. **Meškauskas R., Jokužis V**. The Development and Investigation of the 3D Object Identification System // Electronics and Electrical Engineering. – Kaunas: Technologija, 2006. – No. 5(69). – P. 41 – 44.
2. **Segen J., Kumar S**. Shadow gestures: 3d Hand Pose Estimation using a Single Camera. // IEEE conference on Computer Vision and Pattern Recognition, 1999. – P. 479 – 485.
3. **Colombo C., Del Bimbo A., Valli A.** Visual Capture and Understanding of Hand Pointing Actions in a 3-D Environment IEEE Transactions on systems, man, and cybernetics – part B: Cybernetics, 2003. – Vol. 33, No. 4. – P. 677 – 685.
4. **Hartley R., Zisserman A.** Multiple View Geometry in Computer Vision. Second Edition ISBN 0521 54051 8 Cambridge University Press, 2003. – P. 263.
5. **Serafinavičius P.** Investigation of technical equipments in computer stereo vision: camera calibration techniques // Electronics and Electrical Engineering. – Kaunas: Technologija, 2005. – No. 3(59). – P. 24 – 27.

Submitted for publication 2006 05 25

P. Serafinavičius, G. Daunys. Detection of Hand Position using 3-D Computer Vision // Electronics and Electrical Engineering. – Kaunas: Technologija, 2006. – No. 7(71). – P. 63–66.

Determination of hand position based on one of the simplest technique of 3-D points reconstruction – pinhole camera model and triangulation – is described. Evaluation of this method was carried out using specially developed software based on open source OpenCV library. The software measures an angle between different arm directions in real time. The results of experimental evaluation prove that the highest accuracy is achieved when a plane of measuring angle is perpendicular to cameras' optical axes. Ill.2, bibl. 5 (in English; abstracts in English, Russian and Lithuanian).

П. Серафинавичюс, Г. Даунис. Детекция позиции руки с помощью стереозрения // Электроника и электротехника. – Каунас: Технология, 2006. – № 7(71). – С. 63–66.

Описан способ определения позиции руки, основанный на одном из простейших методов для реконструкции стерео изображений, который использует модель камеры с малым отверстием и триангуляцию. Программа для осуществления экспериментов была создана на базе библиотеки открытого кода OpenCV. Она измеряет угол в реальном времени. Результаты показали, что наибольшая точность измерения угла достигается в плоскости, перпендикулярной оптической оси камеры. Ил. 2, библ. 5 (на английском языке; рефераты на английском, русском и литовском яз.).

P. Serafinavičius, G. Daunys. Rankos pozicijos nustatymas naudojant erdvinio vaizdo kompiuterinę regą // Elektronika ir elektrotechnika. – Kaunas: Technologija, 2006. – Nr. 7(71). – P. 63–66.

Aprašomas rankos pozicijos nustatymo metodas, pagrįstas vienu paprasčiausių erdvinio vaizdo taškų rekonstravimo principų – mažos angos vaizdo kameros modeliu bei trianguliacija. Aprašytam metodui patikrinti buvo sukurta speciali programinė įranga, naudojanti atviro kodo kompiuterinės regos biblioteką OpenCV. Sukurta programinė įranga gali matuoti kampus tarp skirtingų rankos pozicijų realiu laiku. Gauti eksperimento rezultatai rodo, kad kampas tiksliausiai nustatomas, kai jis yra plokštumoje, statmenoje vaizdo kamerų optinėms ašims. Il. 2, bibl. 5 (anglų kalba; santraukos anglų, rusų ir lietuvių k.).