Active Wearable Vision Sensor —Detecting Person's Blink Points and Estimating Human Motion Trajectory—

Akihiro Sugimoto[†] and Takashi Matsuyama[‡]

[†]National Institute of Informatics

[‡]Graduate School of Informatics, Kyoto University

e-mail: sugimoto@nii.ac.jp tm@i.kyoto-u.ac.jp

Abstract

To realize versatile real-time man-machine interactions based on understanding human intention and activities, we develop an active wearable vision sensor. The sensor consists of the detector of person's viewing lines and two active cameras. First, we establish a method for calibrating the sensor so that it can detect person's blink points accurately even in a real situation such that the depth of blink points changes. We formulate errors of the viewing-line detector in terms of the depth in blink points from the person and employ the stereo algorithm to correct the errors. Secondly, we propose a method, the binocular independent fixation control, for incrementally estimating the motion trajectory of the person wearing the sensor. In this method, while the person moves we control the two active cameras independently so that each fixates its optical axis to its own fixation point.

1 Introduction

With the rapid progress of computer facility, computer usage in every aspect of our daily life has become more and more popular; wearing the computer in our everyday life is becoming tangible to reality. Thus, a tremendous amount of efforts has been made to establish technologies for realizing the wearable computer (see [1, 2, 6, 7, 8] for example).

The current approach for the interactions between human beings and computers such as graphical user interface is, on the other hand, based on the concept that the computer is a tool to help our activities. The user, therefore, has to explicitly manipulate objects on a computer monitor to interact with the computer. This kind of our interactions with the computer can be regarded as so-called a *master-servant interaction model.* Namely, we human beings are masters whereas computers are servants, and the computer is just a tool that gives us no response without any order from us. Though multi-modal interface [11] and PUI (Perceptual User Interface) [12, 15] have been proposed for usage of the wearable computer, such interfaces are also based on the master-servant interaction model. In fact, in their context the computer is regarded as a tool to enhance our capabilities and more flexible and simpler interfaces for us to use the computer are being studied.

This current concept of the relationship between human beings and computers should change. The digital barrier in using the computer in our daily life will not disappear, otherwise. The computer in our daily life in the 21st century should have its own identity and exist as a partner of us. We should introduce so-called a *man-machine symbiotic interaction model* to design interactions between human beings and computers. In the man-machine symbiotic interaction model, not only the computer gives us responses based on our orders but also it itself autonomously understands our situation, intention as well as activities, and then provides us in good time with useful information at that time. Such bidirectional interactions between us and computers should also be done in real time.

The above observation motivated us to develop a wearable vision sensor [14]. Our sensor consists of the detector of person's viewing lines and two active cameras. With the cameras that have the common field of view with a person wearing this sensor, the computer can detect the viewing lines of the person. First, we establish a method for calibrating the sensor so that it can detect person's blink points accurately even in a real situation such that the depth of blink points changes. We formulate errors of the viewing-line detector in terms of the depth in blink points from the person and employ the stereo algorithm to correct the errors. Secondly, we propose a method for incrementally estimating the motion trajectory of the person wearing the sensor. In our method, we propose the binocular independent fixation control. That is, while the person moves we control the two active cameras independently so that each fixates its optical axis to its own fixation point. The correspondence of the fixation point over two frames together with the correspondence of lines nearby the fixation point gives us sufficient constraints to determine the person's motion in 3D.

2 Significance of Active Wearable Vision Sensor

A device sensing information in the scene nearby a person is indispensable to the computer for understanding his/her situation, intention as well as activities. In particular, the camera is most promising because of two reasons. One is the amount of acquired information and the other is the capability of having the common field of view with a person. It is also quite natural to regard that the viewing lines of a person result in strongly reflecting his/her interest or attention regardless of his/her consciousness [3, 13].

We may take an alternative approach to understand person's activities where we embed in the surrounding environment multiple sensors such as cameras or magnetic sensors and process information acquired by them. Information acquired by the sensors embedded in the surrounding environment, i.e., information through an *objective* point of view, however, is not satisfactory from the point of view that we capture the intension and interest of the person moving in the environment. Information through the person's viewpoint, i.e., information through a *subjective* point of view, is necessary for such tasks. This can be supported by our experience that we often feel difficulty in communicating our intention to a person who is in a spatially different place. From the point of view that we understand human intention and activities, sharing the common field of view with a person and sharing common inputs with the person are also indispensable. This establishes the significance of the wearable vision sensor. The wearable vision sensor allows the computer to share not only the common field of view with a person but also common inputs to understand the surrounding environment and the human motion.

Moreover, if the camera is active, namely, we can control the optical axis of the camera through a computer, the function of acquiring information is highly enhanced: the computer can control the camera to autonomously acquire information of the environment independent of the person's viewing lines. In other words, depending on the situation, the computer can switch two kinds of functions: (1) the acquisition of *subjective information*, i.e., sharing the common field of view with the person to acquire information from the person's viewpoint, and (2) the acquisition of *objective information*, i.e., autonomously acquiring information of the environment independent of the person's viewing lines. This is the great advantage of acquiring information that cannot be realized with a camera whose optical axis is fixed.

As seen above, the active wearable vision sensor enables the computer to versatilely acquire visual information for understanding human intention and activities and is promising for realizing the interactions between us and computers based on the man-machine symbiotic interaction model. The organization of the paper follows this viewpoint. Namely, detecting blink points of a person in Section 4.2 is devoted for an example of the acquisition of subjective information whereas estimating the human motion trajectory in Section 5 is for an example of the acquisition of objective information.

3 Sensor Configuration

Our wearable vision sensor consists of the head part and the computer (Fig. 1). The head part (Fig. 2) has two cameras and a detector of person's viewing lines. The projection centers of the two cameras are designed to be aligned with the centers of the person's

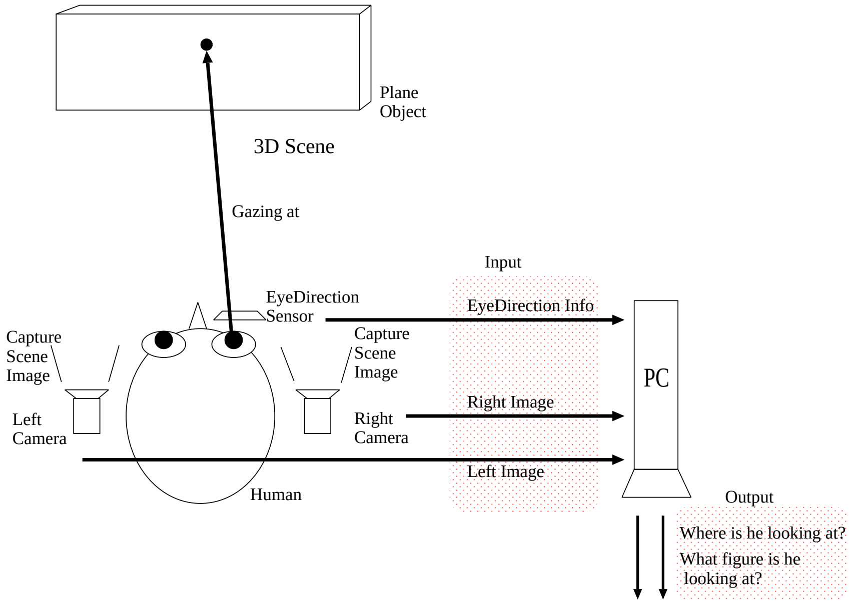


Figure 1: Active wearable vision sensor we developed.

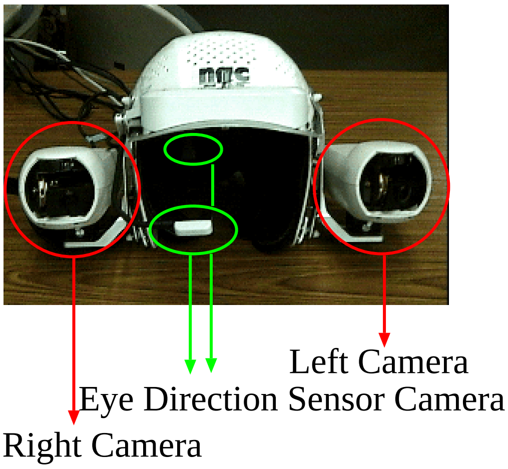


Figure 2: Head part.

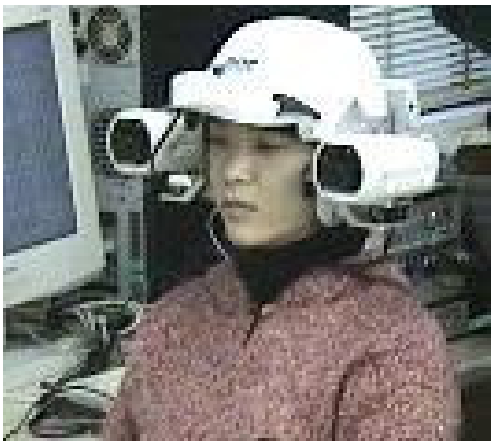


Figure 3: A person with the head part.

eyeballs. The computer, on the other hand, is a PC with Pentium III 750MHz and 1GB memory.

Figure 3 shows a person with the head part of our sensor. Eye-mark recorder EMR-8 from NAC Image Technology is employed as the detector of person's viewing lines. EMR-8 uses the pupil-corneal reflection method in eye tracking and overlays *person's blink points*, i.e., the points in 3D at which the person looks while blinking, onto the image captured by the right camera. This overlaid point is called an "eye mark". The sampling rate of eye marks by EMR-8 is 60Hz (about 17ms). As an active camera, we employed the off-the-shelf camera, EVI-G20, produced by Sony. EVI-G20 has two motors inside the body and accepts commands from a computer to rotate its optical axis by the pan (within ±30°) and the tilt (within ±15°). It is also designed so that the projection center of the camera is identical with the rotation center of the camera body.

The viewing line of a person detected by EMR-8 and two images captured by the two cameras are all put into the computer. The blink point of the person's right eye is superimposed as the eye mark on the right-camera image.

4 Detecting Person's Blink Points

4.1 Introduction of coordinate systems

We have to set some coordinate systems for analysing the relationship between the viewing line of a person and his/her blink point. They are the right-camera coordinates, the left-camera coordinates, the person-centered coordinates, and the viewing-line angle coordinates with respect to the person's right eyeball (Fig. 4).

We introduce the camera coordinate system to each camera where the projection center of the camera is identical with the origin. For reconstructing the 3D position of a point of interest, we calibrate the intrinsic and extrinsic camera parameters in advance and then employ stereo vision technique. In this paper, we employ the method proposed by Zhang [16] to calibrate the camera parameters. We certified that the optical axes of the two cameras are almost parallel with each other and that the poses are identical.

The origin of the person-centered coordinates is set to the middle point between the origins of the two camera coordinate systems. As a result, the origin is almost identical with the middle point between the centers of the person's two eyeballs. The pose of the person-centered coordinates, on the other hand, is set to be identical with that of the two camera coordinate systems.

We set the rotation center of the person's right eyeball as the origin of the viewing-line angle coordinates. In this coordinate system, the coordinates represent rotation

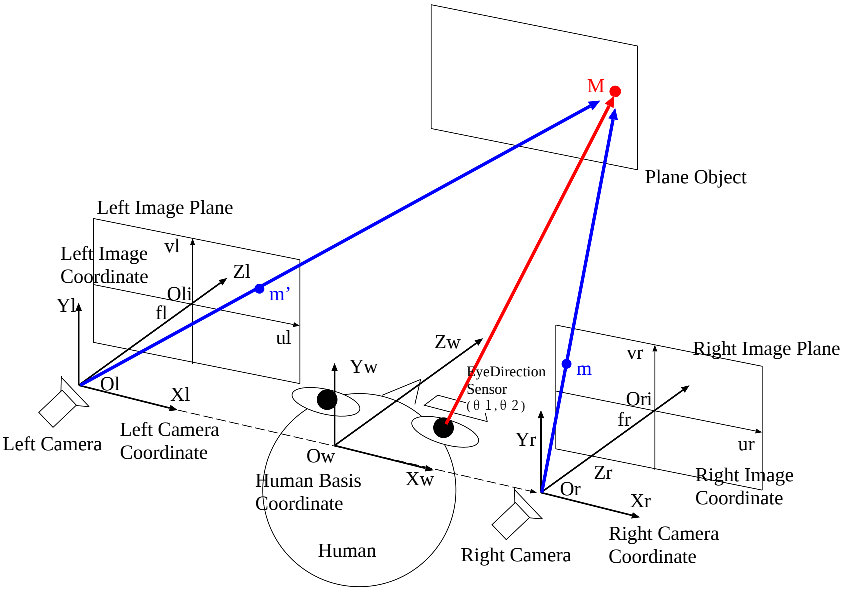


Figure 4: Introduced coordinate systems.

angles, pan and tilt angles, with respect to the optical axis of the right-camera coordinate system. EMR-8 measures the rotation angles of the person's right eyeball in terms of the coordinates in this viewing-line angle coordinate system. In this measurement, the pupil-corneal reflection method is employed where the cornea is illuminated by an infrared light and the light reflected back from the cornea is then captured to estimate the direction of the cornea.

4.2 Sensor calibration

To observe person's blink points we overlay them onto the image captured by the right camera. For this purpose, we have to calibrate the relative position and pose between the right-camera coordinates and the viewing-line angle coordinates. The algorithm for this calibration is provided with EMR-8. Namely, a person gazes at given nine points on a plane (called a calibration plane) one by one in a given order, and then the nine pairs of viewing-line angles and the images of the points are used to calibrate the two coordinates. This algorithm allows the system to overlay the person's blink point onto the image captured by the right camera.

Unfortunately, however, EMR-8 assumes in its usage that the distance between a calibration plane and a person is not large and that the person always keeps his blink points on the calibration plane. These assumptions cause the problem that the overlaid eye

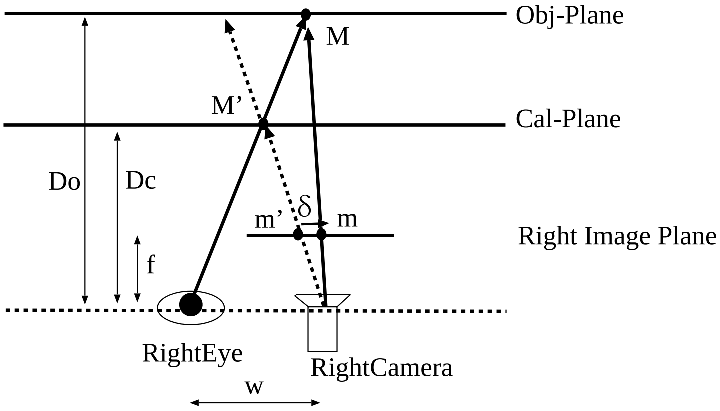


Figure 5: Blink point farther than the calibration plane.

marks do not accurately reflect person's blink points in the image for the case where the distance between the person and his blink points dynamically changes; this case always occurs in our daily life.

For example, we consider the case where person's blink points are farther than a calibration plane (Fig. 5). Let M be the blink point of a person. EMR-8 then (incorrectly) identifies M' on the calibration plane as the blink point of the person and overlays its image m' onto the right-camera image as the person's eye mark at that time. As seen above, this overlay is incorrect because the image m of M should be overlaid. The horizontal $(x-)$ component δ of the residual of m' from m follows from Fig. 5:

$$\delta = wf\left(\frac{1}{D_{\rm c}} - \frac{1}{D_{\rm o}}\right), \tag{1}$$

where f and w respectively denote the focal length of the right camera and the horizontal component of the distance between the rotation center of the person's right eyeball and the projection center of the right camera. $D_{\rm c}$ and $D_{\rm o}$ are the distance of the calibration plane and the blink point from the projection center of the right camera, respectively. This is the formulation of errors of detected eye marks in terms of the depth, $D_{\rm o}$, in blink points. We remark that the vertical $(y-)$ components of the residual can be also derived in the same way. For the case where person's blink points are nearer than the calibration plane, the residual is represented in the similar equation as (1).

To correct the point to be overlaid, we have to know $w, f, D_{\rm c}$ and $D_{\rm o}$ in advance, and then compute δ. We can measure w, f and $D_{\rm c}$ since we can calibrate them beforehand. $D_{\rm o}$, on the other hand, can be computed by a stereo algorithm since two calibrated cameras are mounted on our system. Accordingly, we can correct the residual δ, and this

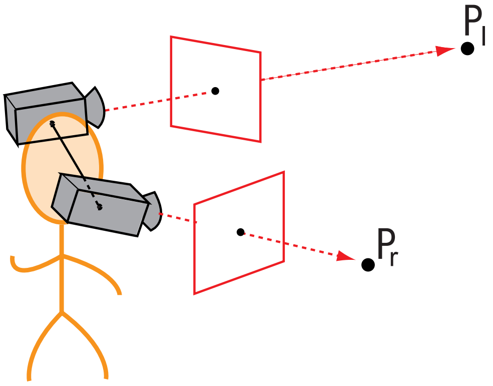


Figure 6: Binocular independent fixation control.

correction enables the system to correctly overlay the blink point onto the image even though the distance between the person and his/her blink points dynamically changes.

5 Estimating Human Motion Trajectory by Binocular Independent Fixation Control

In the previous section, we used the active wearable vision sensor in the context of the acquisition of subjective information: two cameras shared the common field of view with a person wearing the cameras. In this section, by contrast, the sensor is used for the acquisition of objective information: the field of view of the cameras is independent of the person's.

To dynamically interact with a person, the computer should first understand where the person was and where the person is/was going. We describe here the method for incrementally estimating the human motion trajectory by using two active wearable cameras where the *fixation control*, the camera control in which the camera fixates its optical axis to a selected point (called the *fixation point*) in 3D, plays a key role. In our method, we apply the fixation control independently to each active camera. That is, while the person moves, we control the two active cameras independently so that each fixates its optical axis to its own fixation point. We call this camera control the *binocular independent fixation control* (Fig. 6). The correspondence of the fixation point over two frames together with the correspondence of lines nearby the fixation point gives us sufficient constraints to determine the person's motion in 3D.

In the robotics literatures, the framework of stereo vision is widely used to estimate the position and motion of a moving robot [4, 9, 10]. When we employ the stereo vision algorithm, however, we have to make two cameras share the common field of view and, moreover, establish feature correspondences between the images captured by the two cameras. This kind of processing has difficulty in its stableness. In addition, we have another problem in using the stereo vision framework in the context of wearable cameras. Namely, though the accuracy of the estimation is well known to highly depend on the baseline, i.e., the length between the projection centers of the two cameras, in the stereo vision framework, keeping the baseline of two cameras wide is hard when we wear cameras. Therefore, the accuracy of the estimation of the motion trajectory is limited if we employ the stereo vision algorithm.

In the binocular independent fixation control, on the other hand, each camera fixates its optical axis to its own fixation point in 3D and two fixation points are not necessarily the same. This indicates that the two cameras need not share the common field of view. Moreover, in the binocular independent fixation control, the estimation accuracy becomes independent of the baseline of two cameras and is expected to be higher than the case where we use the stereo vision algorithm. This can be understood as follows. If we assume that we set a camera at each fixation point and that the optical axis of each camera is toward the person, then the binocular independent fixation control can be regarded as the situation where we apply the stereo vision framework to estimating the position of the person from the two fixation points. The baseline in this case is identical with the length of the two fixation points. This means that the estimation accuracy is independent of the baseline of the two cameras that the person wears and that selecting fixation points as far as possible from each other allows the estimation accuracy to become high.

5.1 Fixation control

5.1.1 Fixation-point detection and camera control

To realize the fixation control, the computer should autonomously select a point in 3D as the fixation point of the camera and then control the camera so that the camera fixates its optical axis to the point.

In the static scene, properties below are required for a point that is selected as the fixation point of a camera. The point satisfying the properties is suitable for a fixation point and the computer has to detect such a point.

- Actual existence in 3D. (Two twisted lines in 3D, for example, form a point in the image as the intersection of their image lines. Such a point, however, should not be selected as the fixation point since it does not exist in 3D.)

- Easiness in identification in the image. A fixation point should be easy to identify in the image for the accurate fixation control.
- Enough margins to fixate in the physical control of the camera. The point should not easily disappear from the field of view of the camera during the fixation control.
- Enough distance from the other fixation point. As addressed above, the estimation accuracy depends on the distance between two fixation points in the binocular independent fixation control.

When we select a point as the fixation point in the current frame, to realize the fixation control we should first identify the position where the point is in the next frame, and then head the optical axis of the camera toward the new position of the point. The template matching enables the computer to effectively conduct these procedures. That is, we first detect from the current frame, the fixed-size region whose center is the fixation point as a reference template, and then apply the template matching to the next frame to identify the position of the fixation point in the next frame. We now only have to send the command of pan and tilt parameters to the camera so that the optical axis of the camera passes through the new position of the fixation point in the next image. Iterating these procedures realizes the fixation control.

5.1.2 Updating fixation point

To estimate the human motion trajectory in the scene, the binocular independent fixation control should continue without any interruption. When a person widely moves in the scene the case occurs during the motion where the camera cannot capture the current fixation point due to its physical constraint, i.e., the angle limitation of pan and tilt of the camera. In such a case, a new fixation point should be selected: updating the fixation point is necessary.

The point that is newly selected as the fixation point should also satisfy the properties listed in Section 5.1.1. We remark that in the binocular independent fixation control, each camera selects its new fixation point independently at different time. This is because we control two cameras independently.

In the implementation, before a camera cannot capture the current fixation point due to its physical constraint, we keep a point that can be a new fixation point in the image of the camera. We replace the current fixation point by the point to obtain a new fixation point as soon as the camera loses the current fixation point. We then apply the fixation control with respect to the new fixation point. In this way, the estimation of the human motion trajectory continues without any interruption even though a person wearing the cameras widely moves in the scene.

5.2 Geometric constraint derivation on the human motion

We here derive geometric constraints on the human motion based on information obtained during the binocular independent fixation control. Between two cameras that a person is wearing, i.e., a right camera and a left camera, we set the right camera is the base in this paper. Moreover, for simplicity, we assume that the human motion, the motion of a person wearing the cameras, is identical with the camera motion where the camera motion is defined as the motion of the projection center of the base camera. We thus develop a method for estimating the camera motion below.

We assume that the extrinsic parameters between the two cameras as well as the intrinsic parameters of each camera are calibrated in advance. Namely, we let the translation vector and the rotation matrix to make the left-camera coordinates identical with the right-camera coordinates be $\boldsymbol{T}_{\mathrm{in}}$ in the left-camera coordinates and R_{in} in the right-camera coordinates, respectively. $\boldsymbol{T}_{\mathrm{in}}$ and R_{in} are both assumed to be known. We also assume that the orientation of the camera coordinates does not change even though we change pan and tilt of the camera for the fixation control. This means that only the human motion causes changes in orientation and translation of the camera coordinates.

5.2.1 Constraints from fixation correspondence

The fixation control gives us the correspondence of the viewing lines of a camera toward the fixation point over time-series frames. We call this correspondence a *fixation correspondence*. The fixation correspondence enables us to derive a constraint on the camera motion.

Let the projection centers of the left camera and the right camera be C_{ℓ}^{t} and C_{r}^{t} in 3D at time t. We assume that the both cameras have their own fixation points P_{ℓ} and P_{r}. We denote by $\boldsymbol{v}_{\mathrm{r}}^{t}$ the unit vector from C_{r}^{t} to P_{r} in the right-camera coordinates at time t. We see that $\boldsymbol{v}_{\mathrm{r}}^{t}$ represents the viewing line of the right camera toward the fixation point at time t. We also denote by $\boldsymbol{v}_{\ell}^{t}$ the unit vector from C_{ℓ}^{t} to P_{ℓ} in the left-camera coordinates at time t (Fig. 7).

We first focus on the base camera, i.e., the right camera. We assume that the projection center of the right camera moves from C_{r}^{t} to C_{r}^{t+1} in 3D due to the human motion from time t to $t+1$ (Fig. 8). We also assume that the rotation and the translation of the right camera incurred by the human motion are expressed as rotation matrix R in the

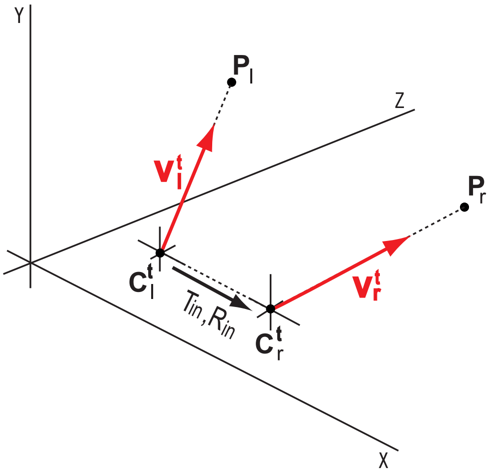


Figure 7: Relationship between the projection centers and the fixation points at time t.

right-camera coordinates at time t and translation vector $\boldsymbol{T}$ in the world coordinates. We remark that the orientation of the world coordinates is assumed to be obtained by applying rotation matrix R_0^{-1} to the orientation of the right-camera coordinates at time t. Our aim here is to derive constraints on R and $\boldsymbol{T}$.

It follows from the fixation correspondence of the right camera that

$$\lambda R_0 \boldsymbol{v}_{\mathrm{r}}^t \quad = \quad \lambda' R_0 R \boldsymbol{v}_{\mathrm{r}}^{t+1} + \boldsymbol{T},$$

where λ and λ' are non-zero constants. The above equation is rewritten by

$$\det \left[\begin{array}{ccc} R_0 \boldsymbol{v}_{\mathrm{r}}^t & R_0 R \boldsymbol{v}_{\mathrm{r}}^{t+1} & \boldsymbol{T} \end{array} \right] \quad = \quad 0, \tag{2}$$

which gives the constraint on the camera motion, R and $\boldsymbol{T}$, derived from the fixation correspondence of the right camera.

On the other hand, $\boldsymbol{v}_\ell^t$ in the left-camera coordinates at time t is identical with $R_{\mathrm{in}} \boldsymbol{v}_\ell^t$ in the right-camera coordinates at time t. The rotation R of the right-camera coordinates from time t to $t+1$ causes the translation $-R_0(R-I)R_{\mathrm{in}}\boldsymbol{T}_{\mathrm{in}}$ of the left-camera coordinates in the world coordinates. This yields

$$\det \left[\begin{array}{ccc} R_0 R_{\mathrm{in}} \boldsymbol{v}_\ell^t & R_0 R R_{\mathrm{in}} \boldsymbol{v}_\ell^{t+1} & \boldsymbol{T} - R_0(R-I)R_{\mathrm{in}}\boldsymbol{T}_{\mathrm{in}} \end{array} \right] \quad = \quad 0, \tag{3}$$

where I is the 3×3 unit matrix. (3) is the constraint on the camera motion derived from the fixation correspondence of the left camera.

(2) and (3) are the constraints on the camera motion in 3D derived from the fixation correspondences obtained by the binocular independent fixation control. When we have

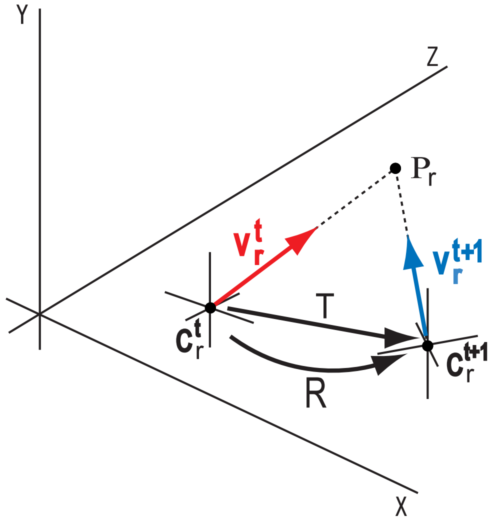


Figure 8: Geometry based on the fixation correspondence of the right camera.

estimated the camera motion up to time t, we know R_0. Then, the unknown parameters in (2) and (3) are R and $\boldsymbol{T}$. We see that (2) and (3) give homogeneous quadratic constraints on R and $\boldsymbol{T}$ respectively.

5.2.2 Constraints from line correspondence

The camera motion has 6 degrees of freedom: 3 for a rotation and 3 for a translation. The number of constraints on the camera motion derived from two fixation correspondences, on the other hand, is two ((2) and (3)). We therefore need to derive more constraints to estimate the camera motion.

We employ lines nearby the fixation point to obtain other constraints on the camera motion. This is because

(i) we find many lines in the indoor scene, for example, the boundaries between walls and a ceiling, the boundaries of windows and those of doors,

(ii) we can easily and accurately detect lines with less computation by using the Hough transformation, and

(iii) we can easily establish line correspondences over time-series frames due to their spatial extents.

In addition to the advantages listed above, constraints on the camera motion derived from line correspondences depend only on the rotation. This is addressed in detail below.

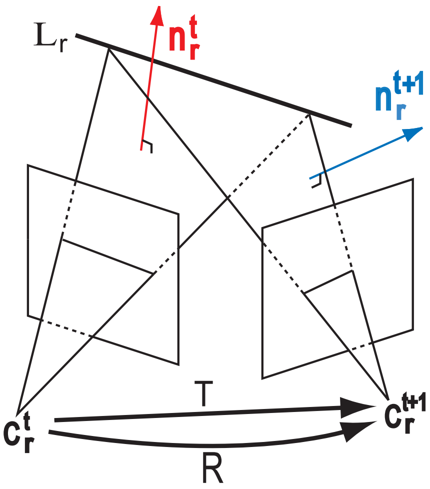


Figure 9: Geometry based on the line correspondence of the right camera.

We first focus on the base camera, i.e., the right camera. Let the projection center of the right camera be $C_{\rm r}^t$ in 3D. We then assume that we establish the correspondence of images of line $L_{\rm r}$ in 3D over time t and $t+1$, where line $L_{\rm r}$ is selected nearby the fixation point of the right camera. Line $L_{\rm r}$ is called a *focused line* in this paper. We denote by $\boldsymbol{L}_{\rm r}$ the unit direction vector of the focused line $L_{\rm r}$ in the world coordinates[1] . We define the *interpretation plane* of the focused line $L_{\rm r}$ as the plane on which both $C_{\rm r}^t$ (the projection center of the camera at the observation time) and the focused line $L_{\rm r}$ exist, and denote by $\boldsymbol{n}_{\rm r}^t$ the unit normal vector of the interpretation plane of the focused line $L_{\rm r}$ in the right-camera coordinates at time t (Fig. 9).

From the relationship of the orientations among the world coordinates, the right-camera coordinates at time t and the right-camera coordinates at time $t+1$, we see that $\boldsymbol{n}_{\rm r}^t$ and $\boldsymbol{n}_{\rm r}^{t+1}$ are expressed as $R_0\boldsymbol{n}_{\rm r}^t$ and $R_0R\boldsymbol{n}_{\rm r}^{t+1}$ in the world coordinates. Since $R_0\boldsymbol{n}_{\rm r}^t$ and $\boldsymbol{L}_{\rm r}$ are orthogonal, and $R_0R\boldsymbol{n}_{\rm r}^{t+1}$ and $\boldsymbol{L}_{\rm r}$ are also orthogonal, we obtain the following constraint on the camera motion from the line correspondence over two frames captured by the right camera:

$$\mu_{\rm r}\boldsymbol{L}_{\rm r} \quad = \quad (R_0\boldsymbol{n}_{\rm r}^t) \times (R_0R\boldsymbol{n}_{\rm r}^{t+1}), \tag{4}$$

where $\mu_{\rm r}$ is a non-zero constant and depends on the focused line.

In the similar way, we obtain the constraint on the camera motion that is derived from

[1] We assume here that the unit direction vector of a focused line in the world is known. The vector, however, can be estimated from (4) during the motion estimation. Namely, we can compute $\boldsymbol{L}_{\rm r}$ (with $\|\boldsymbol{L}_{\rm r}\| = 1$) from (4) because we know R_0 and R if we have estimated the camera motion up to time $t+1$.

the line correspondence of the left camera.

$$\mu_\ell \boldsymbol{L}_\ell \;=\; (R_0 R_{\mathrm{in}} \boldsymbol{n}_\ell^t) \times (R_0 R R_{\mathrm{in}} \boldsymbol{n}_\ell^{t+1}), \qquad (5)$$

where $\boldsymbol{L}_\ell$ denotes the unit direction vector, in the world coordinates, of focused line L_ℓ in the left-camera case and μ_ℓ is a non-zero constant depending on the focused line L_ℓ. $\boldsymbol{n}_\ell^t$ denotes the unit normal vector, in the left-camera coordinates at time t, of the interpretation plane of the focused line L_ℓ in the left camera case.

We see that the translation factors of the camera motion are not involved in the constraints, ((4) and (5)), derived from the line correspondence in each camera. We also see that these constraints are linear homogeneous with respect to R and the non-zero constants.

5.2.3 Estimation of rotation and translation

As investigated in Section 5.2.2, the constraints derived from line correspondences depend only on the rotation of the camera motion. We can thus divide the camera motion estimation into two steps: the rotation estimation and the translation estimation.

The first step is the rotation estimation of the camera motion. We suppose that we have correspondences of n focused lines over two time-series frames. Then, we have $n+3$ unknowns whereas we have $3n$ constraints in this case. Therefore, we can estimate the rotation of the camera motion if we have correspondences of more than two focused lines. To be more concrete, we form a simultaneous system of nonlinear equations that consists of the constraints derived from line correspondences and the orthogonality constraints, i.e., $RR^\top = I$, and then apply a nonlinear optimization algorithm to solve the system. In general, a nonlinear system has multiple solutions and the local minimum trap problem is serious. In our case, however, employing redundant line correspondences allows us to avoid being trapped in a local minimum. This is because the constraints derived from line correspondences are linear with respect to unknown parameters and because such redundancy of linear constraints excludes spurious solutions[2] .

When we finish estimating the rotation of the camera motion, we can move to the second step: the translation estimation. The unknowns are now just the translation factors. The constraint derived from the fixation correspondence thus becomes homogeneous linear

[2] In addition to this, we have another reason to employ the redundancy in line correspondence. In the case where the camera motion is just a translation and where the projection center of a camera moves on the interpretation plane of a focused line that is observed by the camera, the constraints derived from the line correspondence becomes the identical equation. Namely, the constraints do not make sense and no independent constraint on the camera motion is obtained. Employing the redundancy in line correspondence prevents us from falling into such cases.

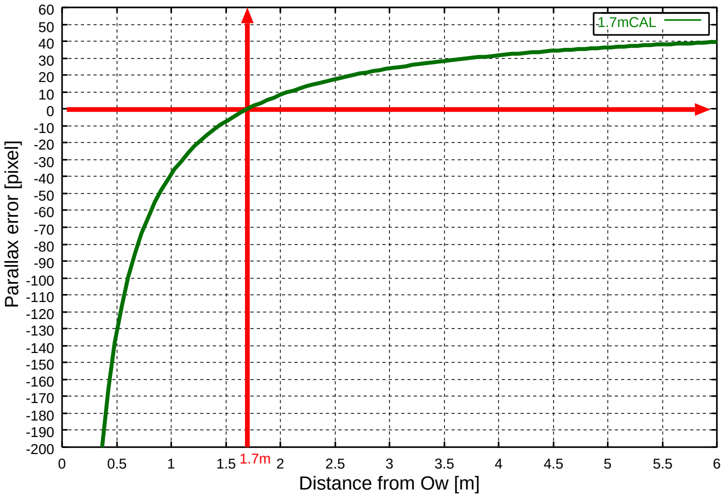


Figure 10: Calibration curve (w = 114mm).

with respect to unknown parameters. Hence, we can obtain the translation of the camera motion up to scale from two fixation correspondences with only linear computation[3] .

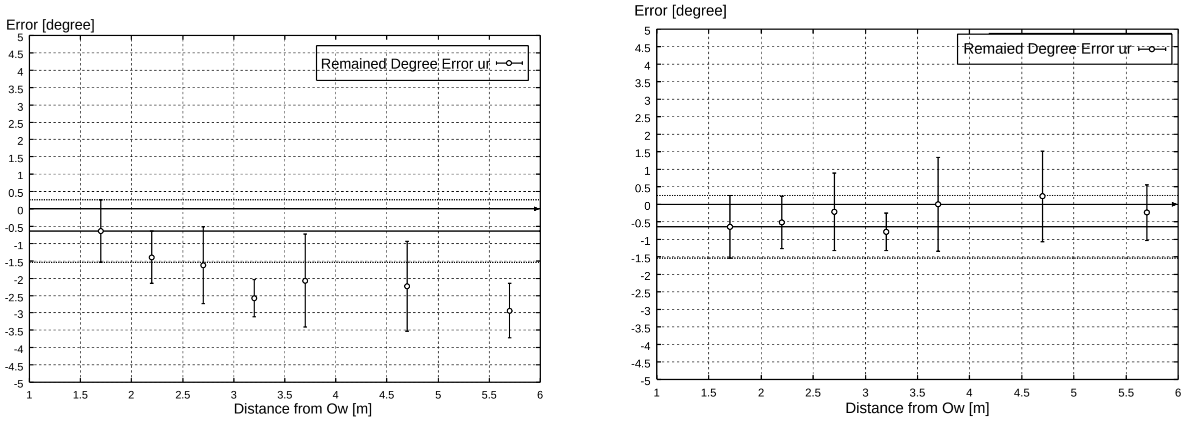
6 Experiments

6.1 Precision evaluation of detected blink points

We evaluated the effect of our correction to the detected eye marks by EMR-8 described in Section 4.2 under the condition that the depth of person's blink points changes.

We first set a calibration plane whose distance is 1.7m from a person, and then instructed the person to gaze at a set of nine points on the plane one by one to calibrate EMR-8. Next we moved the plane so that the distance from the person changes by 0.5m from 1.7m to 5.7m in turn. At each distance, we instructed the person to gaze at another set of nine points and obtained the coordinates in the right-camera image of the eye marks detected by EMR-8. In fact, we used the average of the coordinates of the stably detected eye marks. The nine points here, on the other hand, were also captured by the right camera and formed their images in the right-camera image, whose coordinates were used as the ground truths to evaluate the precision of our correction. Next, we applied our

[3] Whenever we estimate the translation of the camera motion over two frames, we have one unknown scale factor. The trilinear constraints [5] on corresponding points over three frames enable us to adjust the unknown scales with only linear computation. The concrete procedure for this adjustment is given in the appendix.



(a) not corrected

(b) corrected

Figure 11: Errors of detected blink points.

correction described in (1) to the eye marks detected by EMR-8 to obtain the corrected coordinates of images of the blink points. We remark here that we carefully measured D_c and w to obtain $D_c = 1.7$m and $w = 114$mm. We therefore had the calibration curve shown in Fig. 10.

For the cases with/without our correction, we computed the average and the variance of the residuals from the ground truths over the given set of nine points, and compared the two cases (Fig. 11). Note that error bars in Fig. 11 represent the standard deviation. Fig. 11 shows that the residuals of corrected coordinates almost stably remain small independent of the change in distance of the plane from the person. In fact, they are within perturbation of the standard deviation from the average for the distance 1.7m at which EMR-8 was calibrated. This observation indicates that our correction is valid and effective.

6.2 Experiments on estimating a human motion trajectory

We employed the cameras (EVI-G20 from Sony) and the computer (a PC with PentiumIII 750MHz and 1GB memory), both of which are used to develop our active wearable vision sensor (Fig. 2), and applied to them the binocular independent fixation control described in Section 5 to estimate a camera motion trajectory.

We set up an imitative active wearable vision sensor where two cameras with the baseline of about 27cm were mounted on the stage of a tripod (Fig. 12). We then calibrated the intrinsic and extrinsic parameters of the two cameras based on the method proposed by Zhang [16]. The size of images captured by each camera was 640 × 480 pixels.

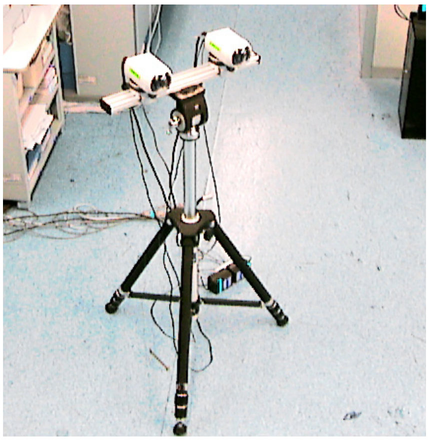
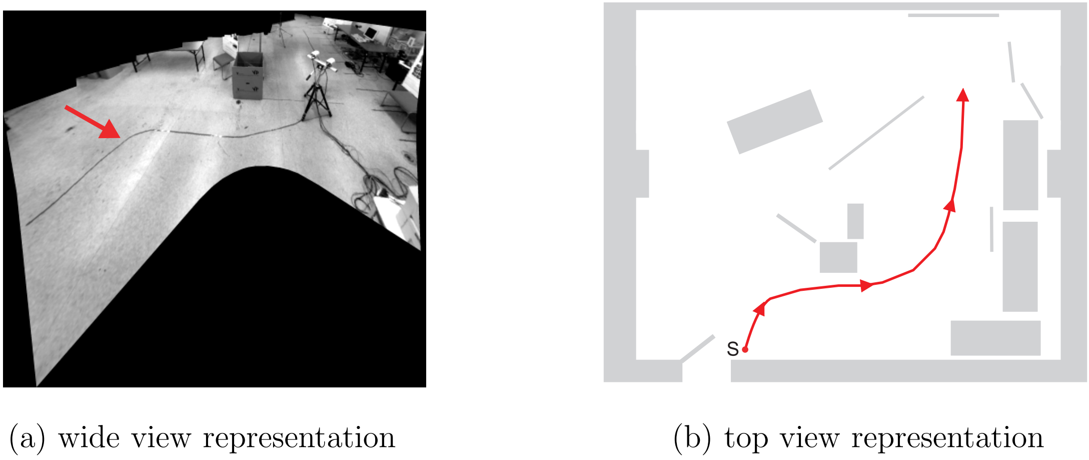


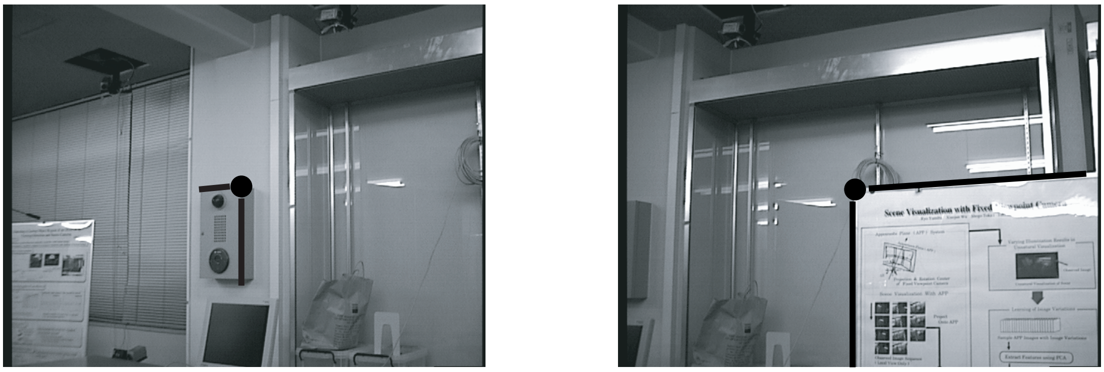
Figure 12: Imitative active wearable vision sensor.



(a) wide view representation

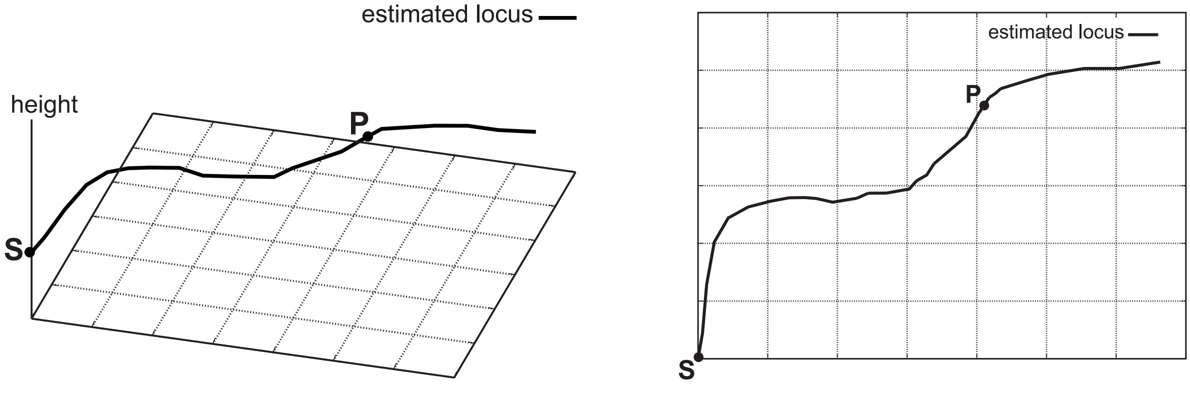
(b) top view representation

Figure 13: Camera motion trajectory.



(a) left-camera image (b) right-camera image

Figure 14: Example of images acquired by the two cameras during the camera motion.



(a) 3D representation (b) top-view representation

Figure 15: Estimated trajectory of the camera motion.

We moved the imitative active wearable vision sensor in the scene. The trajectory of the right-camera motion is shown in Fig. 13. The length of the trajectory was about 6m. We marked 35 points on the trajectory and regarded them as samples during the motion. (In other words, 35 points were sampled during the camera motion of about 6m.) We then applied the binocular independent fixation control only to the samples to estimate the right-camera motion.

In each image captured by each camera at the starting point of the camera motion, we applied the Hough transformation to edges detected from the image to find the intersection points of pairs of lines and then identified a point by hand that actually exists in 3D. The point was set as the fixation point. During the estimation, we updated fixation points 8 times. This updating was also conducted by hand. Fig. 14 shows an example of image pairs captured by the right and left cameras at a marked point. We see that little field of view of the two cameras is common[4] . We remark that the fixation point (the black circle) and two focused lines (the black thick lines) are overlaid onto the images in Fig. 14.

Under the above conditions, we estimated the right-camera motion at each marked point. In estimating the motion at each marked point, we used two focused lines for each camera (we thus used four focused lines in total). In detecting lines, we employed the Hough transformation. Fig. 15 shows the trajectory of the right-camera motion that was obtained by concatenating the estimated motions at the marked points. We note that S means the starting point of the motion.

From the comparison of the estimated trajectory and the actual trajectory of the right-camera motion, we observe that the height from the floor is almost accurately estimated over the trajectory. As for the component parallel to the floor, however, the former part (from S to P in Fig. 15) of the estimated trajectory almost coincides with that of the actual trajectory whereas the latter part (after P) of the estimated trajectory has great aberration from the actual trajectory. We have two reasons that may cause this aberration. One is the incorrect estimation of the motion at P and the other is the effect of the estimation error at P upon the subsequent estimations. In other words, since the motion is incrementally estimated, the accumulation of estimation errors and an incorrect estimation at just one marked point cause aberration. The estimation error can be caused by errors in the fixation correspondence or errors in the line detection. Calibration errors of the two cameras also may cause estimation errors.

As see above, we may conclude that (1) the proposed method accurately estimates the motion trajectory of the camera in general, and that (2) the accumulation of estimated

[4] It is therefore hard to apply the stereo vision algorithm to such input image pairs. In addition, since there were not enough textures in the scene, only few features could be detected. This also causes difficulty in application of the stereo vision algorithm.

errors reduces the accuracy of the total shape of the estimated trajectory.

7 Conclusion

We developed an active wearable vision sensor for versatile man-machine interactions based on understanding human intention and activities. The sensor consists of the detector of person's viewing lines and two active cameras. We first proposed a method for calibrating the sensor so that it detects person's blink points accurately even in a real situation such that the depth of blink points changes. We then proposed a method for incrementally estimating the motion trajectory of the person who are wearing the sensor. In the former method, we realized the information acquisition from the person's viewpoint where the vision sensor shares the common field of view with the person. In the latter method, on the other hand, we realized the autonomous information acquisition for understanding the person's motion trajectory where the field of view of the vision sensor is independent of that of the person. In this way, our active wearable vision sensor enables us to versatilely acquire information for understanding human intention and activities.

Eliminating the accumulation errors in estimating the motion trajectory and improving the accuracy of the estimation are included in the future work. We also plan to develop methods for estimating the position of a person wearing our active wearable vision sensor and for identifying the fixation of his/her viewing lines to make the computer understand the situation where he/she is.

Acknowledgements

The authors are grateful to Akihiro Nakayama and Wataru Nagatomo for the discussion and comments on this work as well as helping to perform the experiments. This work is supported by Grant-in-Aid for Scientific Research of the Ministry of Education, Culture, Sports, Science and Technology of Japan under the contraction of 13224051 and 14380161.

References

[1] H. Aoki, B. Schiele and A. Pentland: Realtime Personal Positioning System for Wearable Computers, Vision and Modeling Technical Report, TR-520, Media Lab. MIT, 2000.

[2] B. Clarkson, K. Mase and A. Pentland: *Recognizing User's Context from Wearable*

Sensors: Baseline System, Vision and Modeling Technical Report, TR-519, Media Lab. MIT, 2000.

[3] R. Carpenter: *Movements of the Eyes*, 2nd ed., Pion, London, 1988.

[4] A. J. Davison and D. W. Murray: Mobile Robot Localisation Using Active Vision, *Proc. of ECCV*, Vol. 2, pp. 809–825, 1998.

[5] R. Hartley and A. Zisserman: *Multiple View Geometry in Computer Vision*, Cambridge Univ. Press, 2000.

[6] M. Kourogi, T. Kurata and K. Sakaue: A Panorama-Based Method of Personal Positioning And Orientation And Its Real-Time Applications for Wearable Computers, *Proc. of Int. Symposium on Wearable Computers*, Switzerland, pp.107-114, 2001.

[7] S. W. Lee and K. Mase: Incremental Motion-Based Location Recognition, *Proc. of Int. Symposium on Wearable Computers*, pp. 123–130, 2001.

[8] W. W. Mayol, B. Trdoff and D. W. Murray: Wearable Visual Robots, *Proc. of Int. Symposium on Wearable Computers*, pp. 95–102, 2000.

[9] N. Molton and M. Brady: Practical Structure and Motion from Stereo When Motion is Unconstrained, *Int. J. of Computer Vision*, Vol. 39, No. 1, pp. 5–23 (2000).

[10] D. W. Murray, I. D. Reid and A. J. Davison: Steering and Navigation Behaviours Using Fixation, *Proc. of British Machine Vision Conference*, 1996.

[11] S. Oviatt and P. Cohen: Multimodal Interfaces that Process What Comes Naturally, *Communications of the ACM*, Vol. 43, No. 3, pp. 45–53 (2000).

[12] A. Pentland: Perceptual Intelligence, *Communications of the ACM*, Vol. 43, No. 3, pp. 35–44 (2000).

[13] A. F. Sanders: *The Selective Progress in the Functional Field of View*, Van Gorcum & Comp., N. V., Amsterdam, 1964.

[14] A. Sugimoto, A. Nakayama and T. Matsuyama: Detecting a Gazing Region by Visual Direction and Stereo Cameras, *Proc. of the 16th International Conference on Pattern Recognition*, Vol. III, pp. 278–282, 2002.

[15] M. Turk and G. Robertson: Perceptual User Interfaces, *Communications of the ACM*, Vol. 43, No. 3, pp. 33–34 (2000).

[16] Z. Zhang: A Flexible New Technique for Camera Calibration, *IEEE Transactions on PAMI*, Vol. 22, No. 11, pp. 1330–1334 (2000).

Appendix (Adjusting scales incurred by translation estimation)

The camera is understood to be the right camera below if not explicitly stated. We suppose that we have obtained rotation matrix R^t and translation vector $\boldsymbol{T}^t$ from time t to $t+1$, both of which are estimated from two images captured at time t and $t+1$. We also suppose that we have obtained rotation matrix R^{t+1} and translation vector $\boldsymbol{T}^{t+1}$ from time $t+1$ to $t+2$, both of which are estimated from two images captured at time $t+1$ and $t+2$. Here the rotation matrix is expressed in the camera coordinates at the time when the motion to be estimated starts (t and $t+1$), and that the translation vector is expressed in the world coordinates. Since we have one unknown scale factor in estimating each translation vector, the translation vectors that express the camera translations are $k^t\boldsymbol{T}^t$ and $k^{t+1}\boldsymbol{T}^{t+1}$, where k^t and k^{t+1} are respectively, non-zero scale factors at time t and $t+1$. Adjusting scales of the two estimated translation vectors is then equivalent to determining the ratio between k^t and k^{t+1}. Now we define

$$k \ := \ \frac{k^{t+1}}{k^t}$$

and develop a method of determining k. We note that the camera is assumed to fixate its optical axis toward fixation point Q (with its homogeneous coordinates $\boldsymbol{X}$ in the world coordinates) in 3D.

Without loss of generality, we may assume that the camera coordinates at time t is identical with the world coordinates (we regard the camera coordinates at time t as the base coordinates). Then, projection matrices P^t, P^{t+1} and P^{t+2} at time $t, t+1$ and $t+2$ are respectively expressed by

$$\begin{aligned} P^t &= K\left[\ I \,\middle|\, \mathbf{0}\ \right], \\ P^{t+1} &= K\left[\ R^t \,\middle|\, -R^t\boldsymbol{T}^t\ \right], \\ P^{t+1} &= K\left[\ R^{t+1}R^t \,\middle|\, -R^{t+1}R^t\boldsymbol{T}^t - R^{t+1}k\boldsymbol{T}^{t+1}\ \right], \end{aligned}$$

where K is the 3×3 matrix expressing the intrinsic camera parameters.

Letting the homogeneous coordinates of the images of Q at time $t, t+1, t+2$ be

$\boldsymbol{x}^t, \boldsymbol{x}^{t+1}, \boldsymbol{x}^{t+2}$, we have

$$\left[\begin{array}{c|ccc} P^t & \boldsymbol{x}^t & 0 & 0 \\ \hline P^{t+1} & 0 & \boldsymbol{x}^{t+1} & 0 \\ \hline P^{t+2} & 0 & 0 & \boldsymbol{x}^{t+2} \end{array}\right] \begin{bmatrix} \boldsymbol{X} \\ -\lambda^t \\ -\lambda^{t+1} \\ -\lambda^{t+2} \end{bmatrix} = 0,$$

where $\lambda^t, \lambda^{t+1}, \lambda^{t+2}$ are all non-zero constants. We note that the existence of Q in 3D is ensured by $\boldsymbol{X} \neq \mathbf{0}$. If we regard the above equation as the equation with respect to $[\boldsymbol{X}\ -\lambda^t\ -\lambda^{t+1}\ -\lambda^{t+2}]^\top$, the necessary and sufficient condition on the existence of Q in 3D is that rank of the 9×7 coefficient matrix is at most 6. In other words, all the (7×7)-minors of the coefficient matrix have vanishing determinants. $K, R^t, \boldsymbol{T}^t, R^{t+1}, \boldsymbol{T}^{t+1}, k, \boldsymbol{x}^t, \boldsymbol{x}^{t+1}$ and $\boldsymbol{x}^{t+2}$ appear in such minors and only k is unknown. It is easy to see that the determinant is linear with respect to k. We can thus obtain k with only linear computation.

The above computation leads to the adjustment of two scales corresponding to one adjacent pair of estimated translation vectors. Chaining this computation over more pairs allows us to adjust all the scales incurred by the translation estimation. After all, only one scale remains unknown.