The Role of Fixation in Visual Motion Analysis

Cornelia Fermüller[1,2]
Yiannis Aloimonos[1]

[1]Computer Vision Laboratory
Center for Automation Research
University of Maryland
College Park, MD 20742-3411
and
[2] Department for Pattern Recognition
and Image Processing
Institute for Automation
Technical University of Vienna
Treitlstraße 3, A–1040 Vienna, Austria

Abstract

The human eye is different from existing electronic cameras because it is not equipped with a uniform resolution over the whole visual field. Near the optical axis it has the fovea where the resolution (over a one degree range) is higher by an order of magnitude than that in the periphery. With a small fovea in a large visual field it is not surprising that the human visual system has developed mechanisms, usually called saccades or pursuits, for moving the fovea in a fast way. It is important to understand both the structure and function of eye movements in the process of solving visual tasks. In other words, how does this particular ability of humans and primates to fixate on environmental points in the presence of relative motion help their visual systems in solving various tasks? To state the question in a more formal setting, we investigate in this paper the following problem: Suppose that we have an anthropomorphic active vision system, that is, a pair of cameras resting on a platform and being controlled through motors by a computer that has access to the images sensed by the cameras in real time. The platform can move freely in the environment. If this machine can fixate on targets being in relative motion with it, can it solve visual tasks in an efficient and robust manner? By restricting our attention to a set of navigational tasks, we find that such an active observer can solve the problems of 3-D motion estimation, egomotion recovery and estimation of time to contact in a very efficient manner. The algorithms for solving these problems are robust and of qualitative nature and employ as input only the spatiotemporal derivatives of the image intensity function (i.e. they make no use of correspondence or optic flow). Fixation is achieved through camera rotation. This amounts to a change of the input (motion field) in a controlled way. From this change additional information is derived making the previously mentioned navigational tasks easier to solve. The potential of a machine possessing gaze control capabilities to successfully address other problems, such as figure-ground segmentation, stereo-fusion, visual servoing for manipulatory tasks and relative depth, as discussed in [8, 17], demonstrate that gaze control is a principle of active vision, as already proposed in [5, 6, 7].

1. Introduction: Visual Motion Interpretation

Visual navigation problems in the past were mostly studied for the case of a passive observer. In order to demonstrate the computational advantages underlying the perceptual capabilities of an active observer capable of controlling its gaze, we first discuss the limitations of passive vision with regard to problems of visual motion analysis.

For years visual motion interpretation has been approached through studying the "structure from motion" problem. The idea is to find methods of recovering the three-dimensional motion parameters and the structure of the objects in view from the dynamic imagery ([15], [24]). The way the problem has been addressed was first to compute the exact position where each point in the image moved to. In cases of small motion the vector field that represents the change of every point in the image, the so called "optical flow field", is computed from the spatiotemporal derivatives of the image intensity function. This requires the employment of additional constraints, such as smoothness. In cases where the motion is considered large, features such as points, lines or contours in images taken at different time instances are corresponded. From the derived optical flow field or the correspondence between features the three–dimensional motion is then determined.

One can distinguish three phases in the evolution of research on the structure from motion problem. First, work dealt with the question of the existence of a solution, i.e. can we extract any information from a sequence of images about the structure and 3-D motion of the scene that cannot be found from a single image? Intensive research has been conducted in this field and several theoretical results have appeared that deal with questions such as: what can be recovered from a certain number of feature points in a given number of frames [24, 4]? Then the uniqueness aspects of the problem were studied. Non-linear algorithms for the recovery of structure and motion from point [14] or line correspondences and optic flow [26] appeared increasingly in the literature. Such algorithms were based on iterative approximation techniques, so they lacked guaranteed convergence as well as clear analytical formulations that would make a proof of uniqueness possible and allow other researchers to build upon them. Later "linear" algorithms and uniqueness proofs came out for points [23] and lines [20], as well as flow [1]; all were based on the same linearization technique. Although research along these lines has been accompanied by many experiments, none of the existing techniques could be used as a basis for an integrated system, working robustly in general environments.

The reasons for the lack of applicability to real world problems are due to the difficulty of estimating retinal correspondence, which is an ill-posed problem; the assumptions that have to be made to derive optical flow; and the sensitivity of 3-D motion estimation to small changes in the data. Even optimal algorithms [19]—optimal under the assumption of Gaussian noise—perform quite poorly in the presence of moderate noise. The efforts to remove these shortcomings contributed to the birth of a new concept, active vision [5, 6, 7].

An observer is active when he has the capability to control the geometric parameters of his sensory apparatus. In [5] it was shown that an observer with the ability of controlled self-motion can solve several recovery problems in a more efficient manner than a passive observer. In this paper we study the computational advantages of fixation in space-time, usually referred to as gaze control. We show that an active observer with the ability to control its gaze and keep an environmental feature stationary on its image can solve several navigational tasks very efficiently, using well defined input and spending very little computational effort.

2. Overview

If we can "recover from a sequence of images the involved structure of the imaged scene and the relative three-dimensional motion", then various subsets of the computed parameters provide sufficient information to solve many practical problems, such as detection of independent motion, passive navigation, obstacle avoidance, prey catching, etc., as well as many other problems related to robotics and automation—hand-eye coordination, automatic docking, teleconferencing, etc. The difficulties posed from the structure from motion problem raise the idea to seek direct solutions to the above problems that don't presume complete recovery. If we can furthermore supply additional information to the solution-finding task, we may solve problems that were originally considered as ill-posed, ill-conditioned and nonlinear. Additional information can be obtained by making the observer active and allowing him therefore to manipulate and control certain parameters. This is the approach called for by the paradigm of Active Vision [6, 5]. In their paper Aloimonos et al. discuss solutions to a few problems for an active observer possessing controlled self-motion, but they consider optical flow as input to their modules. Here, by exploiting the advantages of gaze control, we develop solutions to the 3-D motion estimation problem which do not rely on optic flow or correspondences but use as input only the spatiotemporal derivatives of the image intensity function.

From the measurements on the image we can only recover the relative motion between the observer and any point in the 3-D scene. The model that has mostly been employed in previous research to relate 2-D image measurements to 3-D motion and structure is the one of rigid motion. Consequently, the case of egomotion recovery for an observer moving in a static world has been treated in the same way as the estimation of an object's 3-D motion relative to an observer. We argue here that the rigid motion model is the appropriate one if only the observer is moving, while this holds only for a restricted subset of moving objects—mainly man-made ones. Indeed, all objects in the natural world move non-rigidly. However, considering only a small patch in the image of a moving object, a rigid motion approximation is legitimate.

Therefore, for the case of egomotion we can use data from all parts of the image plane, whereas for object motion we can only employ local information. Hence, we develop two conceptually different algorithms for explaining the mechanisms underlying the perceptual processes of *egomotion recovery* and *3-D object motion recovery*.

We analyze the following two problems:

(a) "Given an active observer viewing an object moving in a rigid manner (translation + rotation), recover the direction of the 3-D translation and the time to collision by using only the spatio-temporal derivatives of the image intensity function". Although this problem is not equivalent to "structure from motion", because it does not fully recover the 3-D motion, it is of importance in a variety of situations. If an object is rotating around itself and also translating in some direction, we are usually interested in its translation—for example in problems related to tracking, prey catching, interception, obstacle avoidance, etc.

(b) Given an active observer moving rigidly in a static environment, recover the direction of its translation and its rotation and determine relative depth. This is the process of passive navigation,[1] a term used to describe the set of processes by which a system can estimate its motion with respect to the environment.

3. The Input

We want to avoid using optical flow and use data that is derived from just the variations in the image intensity function as the input to the estimation of 3-D motion. As the only available constraint for the flow (u, v) of the time changing image $I(x, y, t)$ we accept the constraint $I_x u + I_y v + I_t = 0$ [12], where subscripts denote partial differentiation. This just means that we can only compute the projection of the flow on the gradient direction $((I_x, I_y) \cdot (u, v) = -I_t)$, i.e. the so-called normal flow. This equation, the optic flow constraint equation, is derived when assuming that the irradiance at time t at point $P(x, y)$ and at time $t + \delta t$ at point $P(x + \delta x, y + \delta y)$ are the same, or in other words, $\frac{dI}{dt} = 0$. The input we use is the spatio-temporal variation in the brightness pattern, which is associated with the vector field of apparent velocities, the optical flow field. It is often considered to coincide with the motion field, the projection of the 3-D motion on the image plane. This fact is stated through the assumption $\frac{dI}{dt} = 0$, which says that the two fields are the same. However, the optic flow field and the motion field are not equal in general. Verri and Poggio [25] reported some general results in an attempt to quantify the difference between them. In Fermüller and Aloimonos [10] the difference between the normal components of these two fields is estimated by using a first-order Taylor series approximation for the spatio-temporal variation in the image intensity. If u_n denotes the normal flow value at point (x, y) and $\bar{u}_n$ the normal motion value at the same point, then the difference

[1]Passive navigation is a prerequisite for any other navigational ability. A system can be guided only if there is a way for it to acquire information about its motion and to control its parameters. Although it is possible to obtain the necessary information by using expensive inertial guidance systems, it remains a challenge to solve the task by visual means.

is given by:

$$\bar{u}_n - u_n = \frac{1}{\| \nabla I \|} \frac{dI}{dt}$$

This shows that the two fields are closer when the local image intensity gradient ∇I is high. Thus, if we measure normal flow only in regions where the intensity gradients are of high magnitude, we will guarantee that the normal flow measurements can be used for inferring 3-D motion.

4. Previous Research

Our work is directed towards the recovery of 3-D motion using the activity of fixation or tracking and the image gradients (normal flow).

The idea of employing fixation and using the tracking parameters for motion estimation appeared in [5, 9], where a closed form solution is provided for the computation of the egomotion parameters for a binocular observer by employing the rotation angle and its first and second derivative (angular velocity and acceleration) along with values of the optic flow field. We go two steps further and we develop a solution for a monocular observer using normal flow, i.e. without employing correspondence. On the other hand, the idea of using the image gradients to directly estimate 3-D motion without going through the intermediate stage of calculating the optic flow field first appeared in the work of Aloimonos and Brown [3]. They presented a complete solution for the case of pure rotation, whereas a detailed study of translational motion can be found in Horn and Weldon [13] and Negahdaripour [16]. Finally, a hybrid technique appeared recently [21], using both optical flow and image gradients for addressing 3-D motion estimation in the general case (rotation and translation). Our contribution here lies in the introduction of several novel geometric properties of a normal flow field due to rigid motion that give rise to simple pattern matching techniques for recovering 3-D motion.

5. The Observer and the Choice of the Coordinate System

Figure 1 depicts a pictorial description of the active observer. Notice that the camera, controllable by a motor, is resting on a platform that can move rigidly. Figure 2 shows a geometric model of the camera. O denotes the nodal point of the eye and the image plane is perpendicular to the optical axis OZ at distance f (focal length) from the origin. The image is formed through perspective projection.

Since motion parameters are expressed relative to a coordinate system, prediction of the position of the moving entity (object or observer) at the next time instance is dependent on the choice of the coordinate system. In the case of egomotion it makes sense to attach the coordinate system onto the observer, simply because the

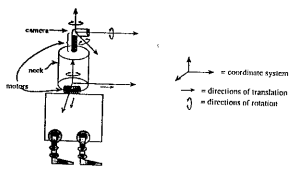


Figure 1: The active observer.

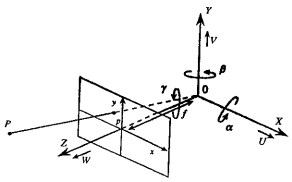


Figure 2: Imaging geometry and motion representation (camera-centered).

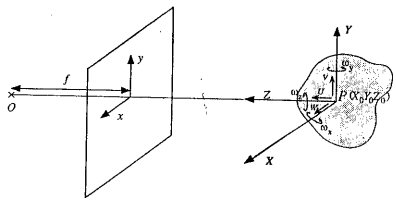


Figure 3: Object-centered coordinate system.

quantities recovered are directly related to the way the observer moves (Figure 2). On the other hand, when the observer needs to make inferences regarding another object's motion, the ideal place to put the origin of the coordinate system would be the mass center of the object (the natural system).

Since the mass center is not known, different choices have to be made. Most commonly the camera's nodal point is chosen as the center of the coordinate system ("camera-centered" coordinate system). Rotation is described around the nodal point. In the case of object motion this leads to different values for the motion parameters for each new frame, which is an unwelcome effect in the task of finding translational motion.

We therefore decided to attach the center of rotation to the object's point of intersection with the optical axis (an "object-centered" coordinate system) (see Figure 3). The active observer is free in its choice of the center and will therefore decide for a point belonging to a neighborhood of non-uniform brightness with distinguishable features.

This approach can be justified by the following argument: When choosing as fixated point the mass center of the object's image or a point in its near neighborhood, the resulting motion parameters are in many cases close to those of the natural system. In the natural coordinate system with center O_{natural} the velocity v at point P is due to the translational and the rotational component:

$$v = t_{\text{natural}} + \omega \times \overrightarrow{O_{\text{natural}} P}$$

and in the object-centered coordinate system with center O_{object} the same velocity is expressed as

$$v = t_{\text{object}} + \omega \times \overrightarrow{O_{\text{object}} P}$$

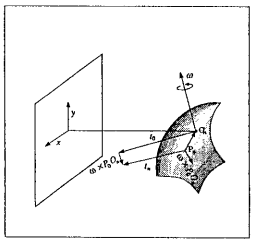


Figure 4: The difference in translation between t_n in the natural system with center O_n and t_o in the object centered system with center O_o is $\omega \times \overrightarrow{O_o O_n}$.

Therefore the difference in translation between t_{natural} and t_{object} (see Figure 4) is given by:

$$\begin{aligned} t_{\text{natural}} - t_{\text{object}} &= \omega \times (\overrightarrow{O\text{object}P} - \overrightarrow{O\text{natural}P}) \\ &= \omega \times \overrightarrow{O_{\text{object}} O_{\text{natural}}} \end{aligned}$$

This value becomes smaller as $\overrightarrow{O_{\text{object}} O_{\text{natural}}}$ decreases.

6. Active 3-D Motion Estimation

We are accomplishing the computation of the FOE and the time to collision through three modules that involve the activities of fixation and tracking.

1. By fixating at an object point, which we consider to be the origin of the used coordinate system, we get image velocity at the center that represents the projection of parallel translation. We show how tracking can be used to derive the projection of parallel translation from just the spatio-temporal derivatives.
2. In the next step, the output of the first module is used to acquire information about translation parallel to the optical axis. Again tracking is used, here as a tool for accumulating depth information over time.
3. In a third module we show that time to collision is related to the FOE and how to estimate it from the spatio-temporal information at the fixated point.

6.1. Tracking gives parallel translation

The first activity used in this approach is fixation. This action provides us with linear relations between the 3-D and the 2-D velocity-parameters. An object at distance Z in front of the camera moves in the 3-D environment with translational velocity (U, V, W) and rotational velocity $(\omega_x, \omega_y, \omega_z)$. In an object-centered coordinate system with center $P(X_0, Y_0, Z_0)$ under perspective projection the optical flow (u, v) is related to these parameters through the following equations:

$$\frac{dx}{dt} = u = \frac{Uf}{Z} - \frac{Wx}{Z} - \frac{xy\omega_1}{f} + \omega_2\left(\frac{x^2}{f} + \frac{f(Z-Z_0)}{Z}\right) - \omega_3 y$$
$$\frac{dy}{dt} = v = \frac{Vf}{Z} - \frac{Wy}{Z} - \omega_1\left(\frac{y^2}{f} + \frac{f(Z-Z_0)}{Z}\right) + \frac{\omega_2 xy}{f} + \omega_3 x$$

In a small area around the center x, y and $\frac{(Z-Z_0)}{Z}$ are close to zero. The optical flow components due to rotation and due to translation parallel to the optical axis converge to zero, and u becomes $\frac{Uf}{Z}$ and v becomes $\frac{Vf}{Z}$.

The flow at the center of the image gives the projection of parallel translation, but only normal flow is available. We show that tracking can be used for the evaluation of optical flow in an iterative technique and prove the convergence of the method to the exact solution.

The problem of current optical flow algorithms is that additional constraints are imposed. Constraints that impose a relationship on the values of the flow field are usually used, and this results in assumptions, such as smoothness, about the scene in view. This basic problem is overcome by providing the observer with activity. The computation is thus transferred to the active observer, who has the ability to iteratively adjust his motion through his control mechanism to the given situation.

In cases where the dominant motion of the object is translation towards the observer, the resulting optical flow vectors are emanating from a point which lies inside the object's image. The coordinates of this point, the FOE, are consequently close to zero. Otherwise the optical flow pattern is due to vectors that are about parallel and have about the same magnitude. Typical normal flow patterns for both cases are shown in Figure 5.

For these cases, where the FOE lies inside the object, the normal flow vectors are mainly due to translation, because the rotational components near the object center are very small. Therefore a simple technique using only the direction of the normal flow measurements can be applied. Given the normal flow vector at a point, we know that the FOE lies in the half-plane, which is separated from the one containing the normal flow vector through the greylevel edge. Considering every available normal flow measurement will narrow the possible location of the FOE to a small area (see Figure 6) (see also [13, 2]). When dealing with such normal flow patterns, it would make no sense to use the method introduced in this paper; we are concerned here with the more complicated case as displayed in Figure 5b.

Let us compute the normal flow in a set of directions in a small area around the

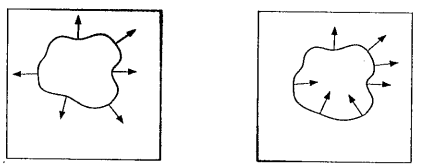


Figure 5: (a) Normal flow vectors emanating from a point inside the object. (b) Normal flow vectors, when the translational component parallel to the image plane is not much larger than the component perpendicular to the plane.

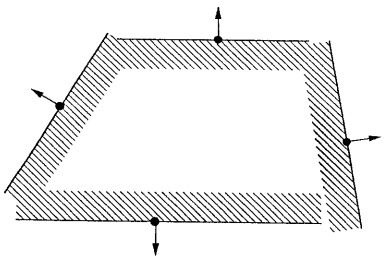


Figure 6: Each available normal flow measurement constrains the possible location of the FOE.

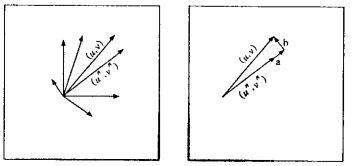


Figure 7: (a) Normal flow vectors measured in different directions. (b) The new flow vector (resulting from object motion and tracking) is due to a, the error in magnitude, and due to b, the error in direction.

origin (fixation point). The normal flow is the projection of the optical flow on the gradient direction. The largest of the normal flow values in the different directions is therefore the one closest to the optical flow. Let us call this normal flow vector the "maximum normal flow" and denote it by (u^n, v^n) (see Figure 7a). We take it as an approximation to the correct optic flow and use it to track the fixated point. The purpose of tracking is to correct for the error in the approximation. In order to keep a point with optical flow (u, v) in the center of the image the observer has to perform a movement that produces the same value of optical flow in the opposite direction. The way our observer accomplishes this task is by rotating the camera around the nodal point about the x- and y-axis. While the observer is moving it takes the next image and computes again the normal flow vectors. If the maximum normal flow was equal to the optical flow, a new optical flow (due to object motion and egomotion) of zero will be achieved.

Usually, however, the maximum normal flow and the optical flow are not equal; they will differ in magnitude and/or in direction. An error in magnitude results in a flow vector in the direction of maximum normal flow, and an error in direction creates a flow vector perpendicular to it (see Figure 7b). The actual error is usually in both magnitude and direction. Thus the new flow vector is a vector sum of the two components. Again it can be approximated by the largest normal flow vector measurement. The new measured normal flow is used as a feedback value to correct the optical flow and the tracking parameters; the new normal flow vector is added to the maximum normal flow vector computed in the first step. Proceeding by applying the same technique to the successive estimated errors will result in an accurate estimate of the actual flow after a few iterations. The proof of convergence to the exact solution follows:

We use here a simplified model to explain tracking. The change of the local coordinate system during tracking and the fact that the object is coming closer is not considered. Since for the purpose of optical flow estimation the number of tracking steps is small, the error originating from this model is not essential. Concerning a specific application, the algorithm will stop when the computed error is smaller than a given threshold, which will cover model errors.

In each iteration step we are computing an approximation to the difference between the observer's egomotion and the object motion. Considering the possible sources of error we have to show that the approximation error will become zero.

Deviations of the chosen maximum normal flow from the optical flow value are due to the following reasons:

- Deviations covered through the model:
  The fact that normal flow measurements are computed in a finite number of directions causes an error in direction of up to half the size of the interval between two normal flow measurements. If measurements in n directions are performed the maximum error y is bounded by: $y < \frac{\pi}{n}$.
- Deviations coming from simplifications and discrete computations:
  In the evaluation of flow measurements the parts linear and quadratic in x, y, and $Z - Z_0$ are ignored. Furthermore each measurement in one direction is computed as the average of the normal flow values in a range y of directions. These reasons may cause errors in magnitude as well as direction, and a different vector than the closest normal flow vector may be chosen.
- General errors occuring in normal flow computation:
  Sensor noise in normal flow measurements and the numerical computation of the derivatives of the image intensity function can influence the magnitude and the direction of the estimated value.

Let v be the magnitude of the actual optical flow. The error sources give rise to specifying the error in magnitude, x, in percentage of the actual value. x_i is the magnitude error in the maximum normal flow measurement in step i and y_i is the the angle between maximum normal flow vector and the optical flow vector, where $x_i < x$ and $y_i < y$. Therefore the difference between the optical flow and the first measurement of maximum normal flow is given by $diff_1 = \begin{pmatrix} vx_1 \cos y_1 \\ v \sin y_1 \end{pmatrix}$, where the x-axis is aligned with the maximum normal flow vector (see Figure 8). The square of its magnitude is computed as:

$$||diff_1||^2 = v^2 x_1^2 \cos^2 y_1 + v^2 \sin^2 y_1$$

The second normal flow vector, if measured from the direction of the maximum normal flow vector derived in the second step, is given by $diff_2 = \begin{pmatrix} ||diff_1|| x_2 \cos y_2 \\ ||diff_1|| \sin y_2 \end{pmatrix}$, and the square of its magnitude is therefore

$$||diff_2||^2 = x_1^2 x_2^2 v^2 \cos^2 y_1 \cos^2 y_2 + x_1^2 v^2 \cos^2 y_1 \sin^2 y_2 + v^2 \sin^2 y_1 \sin^2 y_2 + x_2^2 v^2 \sin^2 y_1 \cos^2 y_2$$

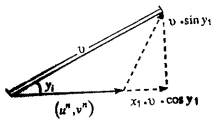


Figure 8: Difference between optical flow vector and maximum normal flow vector.

In general, if we denote by $\{a, b\}$ the fact that either a or b has to be chosen, then $||diff_n||^2$ can be expressed as

$$||diff_n||^2 = v^2 \sum_{\text{all permutations}} \prod_{i=1}^{n} \{x_i^2 \cos y_i^2, \sin y_i^2\}$$

Since $x_i < 1$ and $\sin y_i < 1$ it follows that $\prod_i \{x_i^2 \cos y_i^2, \sin y_i^2\}$ and thus the whole term converges to zero. Therefore we have shown the convergence of the approximation value to the actual optical flow value for the "simplified tracking model".

6.2. Estimating the FOE using tracking

When continuing with tracking over time, as an object comes closer and the value of Z becomes smaller, the optical flow value increases. In order to track correctly and adjust to the increasing magnitude of the optical flow value, the tracking parameters have to be changed, too. ¿From the change of the tracking parameters the change in Z can be derived. If tracking is accomplished by rotation with a certain angular velocity, this just means that the change in depth is derived from the angular acceleration. In the sequel we show the relation between image motion and tracking movement and explain the computation of the tracking parameters, which have to be changed in every step. We explain the exact process of tracking for a geometric setting consisting of a camera that is allowed to rotate around two fixed axes: X- and Y-. These axes coincide with the local coordinate system of the image plane at the beginning of the tracking process.

We describe rotation by an angle ϕ around and axis, which is given by its directional cosines n_1, n_2, n_3, where $n_1^2 + n_2^2 + n_3^2 = 1$. The transformation of a point P with coordinates (X, Y, Z) before and (X', Y', Z') after motion is described through the linear relation:

$$\begin{pmatrix} X' \\ Y' \\ Z' \end{pmatrix} = R \begin{pmatrix} X \\ Y \\ Z \end{pmatrix}$$

where the transformation matrix R is of the following form:

$$\begin{pmatrix} n_1^2 + (1 - n_1^2)\cos\phi & n_1 n_2(1-\cos\phi) - n_3 \sin\phi & n_1 n_3(1-\cos\phi) + n_2 \sin\phi \\ n_1 n_2(1-\cos\phi) + n_3 \sin\phi & n_2^2 + (1-n_2^2)\cos\phi & n_2 n_3(1-\cos\phi) - n_1 \sin\phi \\ n_1 n_3(1-\cos\phi) - n_2 \sin\phi & n_2 n_3(1-\cos\phi) + n_1 \sin\phi & n_3^2 + (1-n_3^2)\cos\phi \end{pmatrix}$$

$$\equiv \begin{pmatrix} r_1 & r_2 & r_3 \\ r_4 & r_5 & r_6 \\ r_7 & r_8 & r_9 \end{pmatrix}$$

Since the image coordinates (x, y) are related to the 3-D coordinates through: $x = Xf/Z$ and $y = Yf/Z$, we get the following equations:

$$x' = \frac{(r_1 x + r_2 y + r_3 f) f}{(r_7 x + r_8 y + r_9 f)}$$

$$y' = \frac{(r_4 x + r_5 y + r_6 f) f}{(r_7 x + r_8 y + r_9 f)}$$

In order to compensate for the image motion (u, v) of the point P_o, which moves from (0, 0) to (u, v) at one time unit the camera has to be rotated by ϕ, n_1, and n_2, where

$$\begin{aligned} u &= n_2 f \tan\phi \\ v &= -n_1 f \tan\phi \end{aligned}$$

Taking at the center of the image the flow measurements (u, v) at the beginning of the tracking process at time t_1, and assuming that the object doesn't change its distance Z_1 to the camera, we can conclude that during a time interval Δt an image flow $(u\Delta t, v\Delta t)$ would be measured. The tracking motion necessary for compensation is given by

$$\frac{Uf}{Z_1} = n_2 \tan\phi.$$

But at time t_2 the object has moved to distance Z_2 and we measure a rotation

$$\frac{Uf}{Z_2} = n_2' \tan\phi'$$

Figure 9 shows the relationship between the 3-D motion and the tracking parameter. Since $Z_2 - Z_1 = W\Delta t$, the change in the reciprocal of the rotation angle is proportional to $\frac{W}{U}$, because

$$\frac{1}{n_2 \tan\phi} - \frac{1}{n_2' \tan\phi'} = \frac{Z_2 - Z_1}{U\Delta t} = \frac{W\Delta t}{U\Delta t}$$

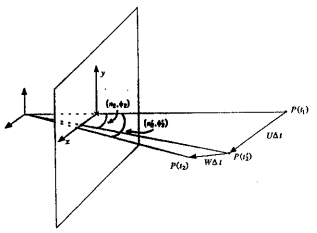


Figure 9: ¿From the optical flow value, which is due only to translation parallel to the image plane, a translation of P from $P(t_1)$ to $P(t_2)$ is inferred, and therefore the tracking parameters (n_2, ϕ) are expected. But actually the point has moved to $P(t_2)$ and a rotation described by $(n_{2'}, \phi')$ is measured.

and the FOE $(\frac{U}{W}, \frac{V}{W})$ can be computed as

$$\frac{U}{W} = 1/(\frac{1}{n_2' \tan\phi'} - \frac{1}{n_2 \tan\phi}) = 1/(\frac{1}{n_2' \tan\phi'} - \frac{f}{u\Delta t})$$

and

$$\frac{V}{W} = 1/(\frac{1}{-n_1' \tan\phi'} - \frac{f}{v\Delta t}).$$

It remains to be explained how tracking is actually pursued, since we are facing the problem of a constantly changing local coordinate system. The interested reader can consult [11], which is devoted to the tracking parameter computation.

6.3. Estimating the time to collision

If the values of the motion parameters don't change over the tracking time the value $\frac{Z}{W}$, the time to collision, expresses the time left until the object will hit the infinitely large image plane. A relationship between FOE and time to collision is inherent in the scalar product of the optical flow vector $\overrightarrow{(u, v)}$ with the vectors in gradient direction $\overrightarrow{(\alpha, \beta)}$:

$$\begin{pmatrix} u \\ v \end{pmatrix} \begin{pmatrix} \alpha \\ \beta \end{pmatrix} = ||v^n||$$

For the pixels in the center, for which we ignore the linear and quadratic parts in x, y and $\frac{Z - Z_0}{Z}$ in the relation between optical flow and 3-D parameters we get the relationship:

$$\frac{Uf}{Z}\alpha + \frac{Vf}{Z}\beta. = ||v^n||$$

$$\frac{Uf}{W}\alpha + \frac{Vf}{W}\beta = ||v^n||\frac{Z}{W}$$

Since we know the FOE, we can compute the time to collision from this relationship, by measuring the normal flow value in all directions of the set and by solving an overdetermined system of linear equations by minimizing the least square error.

7. Active Egomotion Recovery

For an active monocular observer undergoing unrestricted rigid motion in the 3-D world we compute the parameters describing this motion. Using a camera-centered coordinate system, the equations relating the velocity (u, v) of an image point to the 3-D velocity and the depth Z of corresponding scene point are [14]:

$$u = \frac{(-Uf + xW)}{Z} + \alpha\frac{xy}{f} - \beta(\frac{x^2}{f} + f) + \gamma y$$

$$v = \frac{(-Vf + yW)}{Z} + \alpha(\frac{y^2}{f} + f) - \beta\frac{xy}{f} - \gamma x$$

where (U, V, W) denotes the translation and (α, β, γ) the rotation vector.

The number of motion parameters that a monocular observer is able to compute under perspective projection is limited to five: the three rotational parameters and the direction of translation. We therefore introduce coordinates for the direction of translation $(x_0, y_0) = (Uf/W, Vf/W)$, and rewrite the right-hand sides of the above equation as sums of translational and rotational components:

$$u = u_{\text{trans}} + u_{\text{rot}} = (-x_0 + xf)\frac{W}{Z} + \alpha\frac{xy}{f} - \beta(\frac{x^2}{f} + f) + \gamma y$$

$$v = v_{\text{trans}} + v_{\text{rot}} = (-y_0 + yf)\frac{W}{Z} + \alpha(\frac{y^2}{f} + f) - \beta\frac{xy}{f} - \gamma x$$

Since we can only compute the normal flow, the projection of the optical flow on the gradient direction (n_x, n_y), only one constraint on the optical flow can be derived at any given point. The value u_n of the normal flow vector along the gradient direction is given by

$$
\begin{aligned}
u_n &= un_x + vn_y \\
u_n &= ((-x_0 + xf)\frac{W}{Z} + \alpha\frac{xy}{f} - \beta(\frac{x^2}{f} + f) + \gamma y)n_x \\
&\quad +((-y_0 + yf)\frac{W}{Z} + \alpha(\frac{y^2}{f} + f) - \beta\frac{xy}{f} - \gamma x)n_y \qquad (1)
\end{aligned}
$$

The above equation demonstrates the difficulties of motion computation using normal flow for a passive observer. There is only one constraint at every image point but there are five unknown motion parameters and every new point introduces one more unknown (a scaled depth component $-\frac{W}{Z}$). However, the ability of an active observer to fixate at an environmental point and keep it stationary at the center of the visual field can be exploited to provide additional information and thus simplify the problem. The estimation of an active observer's 3-D motion relative to a static scene is accomplished through four modules.

1. Through the fixation and tracking of a point in the scene additional information about the location of the FOE is derived. The FOE is constrained to lie on a straight line and this line also supplies partial information about the observer's rotation (Section 7.1).

2. Selected normal flow values form a global pattern in the image plane which is defined by the coordinates of the FOE and one rotational parameter. Using the information provided by the previous module, locating this pattern amounts to one-dimensional search. This procedure provides a set of possible locations for the FOE (Section 7.2).

3. In order to further narrow down the possible locations of the FOE and to compute the remaining rotational parameters, a process of "detranslation" is performed. For every candidate FOE provided by the previous module the normal flow vectors which do not contain that translation are examined to find out whether they are only rotational (Section 7.3).

4. Finally, the fourth module (total derotation) eliminates all impossible solutions by checking the validity of the five motion parameters at every image point (Section 7.4).

7.1. The fixation constraint

Assume that an active observer in rigid motion is tracking, as before, an environmental point whose image (x, y) lies at the center of the visual field ($(x, y) = (0, 0)$). Assume then that during a small time interval $[t_1, t_2]$ the motion of the observer remains constant and that during this time the camera, in order to correctly track, rotates around its x- and y-axes with rotational velocities $\omega_x(t), \omega_y(t)$ respectively, with $t \in [t_1, t_2]$. The tracking rotation adds to the existing flow field (u, v) a rotational flow field (u_{tr}, v_{tr}), where:

$$
\begin{aligned}
u &= \frac{-Uf + xW}{Z} + \frac{\alpha xy}{f} - \beta\left(\frac{x^2}{f} + f\right) + \gamma y \\
v &= \frac{-Vf + yW}{Z} + \alpha\left(\frac{y^2}{f} + f\right) - \beta\frac{xy}{f} - \gamma x \\
u_{tr} &= \omega_x\frac{xy}{f} - \omega_y\left(\frac{x^2}{f} + f\right) \\
v_{tr} &= \omega_x\left(\frac{y^2}{f} + f\right) - \omega_y\frac{xy}{f}.
\end{aligned}
$$

ω_x, ω_y are the tracking velocities at the time of the observation, and Z is the depth of the tracked point.

As before, if tracking rotation is represented by an angle ϕ around a rotation axis $(n_1, n_2, 0)$ with n_1, n_2 directional cosines, then the introduced flow (u_{tr}, v_{tr}) is given by:

$$
\begin{aligned}
u_{tr} &= n_2 f \tan\phi \\
v_{tr} &= -n_1 f \tan\phi
\end{aligned}
$$

Since the camera is continuously tracking the point at the origin, at any time $t \in [t_1, t_2]$ the introduced tracking motion compensates for the existing flow there, i.e.

$$
\begin{aligned}
n_{2_t} f \tan\phi_t &= \frac{Uf}{Z_t} + \beta f \\
n_{1_t} f \tan\phi_t &= -\frac{Vf}{Z_t} + \alpha f
\end{aligned}
$$

with the subscript t denoting the time of observation. Writing the above two constraints at times t_1 and t_2 we have:

$$
n_{2_{t_1}} f \tan\phi_{t_1} = \frac{Uf}{Z_{t_1}} + \beta f \qquad (2)
$$

$$
n_{1_{t_1}} f \tan\phi_{t_1} = -\frac{Vf}{Z_{t_1}} + \alpha f \qquad (3)
$$

$$
n_{2_{t_2}} f \tan\phi_{t_2} = \frac{Uf}{Z_{t_2}} + \beta f \qquad (4)
$$

$$
n_{1_{t_2}} f \tan\phi_{t_2} = -\frac{Vf}{Z_{t_2}} + \alpha f \qquad (5)
$$

Subtracting (4) from (2) and (5) from (3), we obtain:

$$
\begin{aligned}
f(n_{2_{t_1}} \tan\phi_{t_1} - n_{2_{t_2}} \tan\phi_{t_2}) &= Uf\left[\tfrac{1}{Z_{t_1}} - \tfrac{1}{Z_{t_2}}\right] \\
f(n_{1_{t_1}} \tan\phi_{t_1} - n_{1_{t_2}} \tan\phi_{t_2}) &= -Vf\left[\tfrac{1}{Z_{t_1}} - \tfrac{1}{Z_{t_2}}\right]
\end{aligned}
$$

or by dividing

$$
\frac{V}{U} = \frac{n_{1_{t_2}} \tan\phi_{t_2} - n_{1_{t_1}} \tan\phi_{t_1}}{n_{2_{t_1}} \tan\phi_{t_1} - n_{2_{t_2}} \tan\phi_{t_2}}
$$

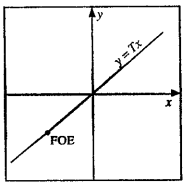


Figure 10: Fixation constrains the FOE lo lie on the line $y = Tx$ and provides the value for the ratio $\frac{\beta+\omega_y}{\alpha+\omega_x} = -T^{-1}$ (see 7.2).

In the sequel we denote the known quantity $\frac{n_{1_{t_2}} \tan\phi_{t_2} - n_{1_{t_1}} \tan\phi_{t_1}}{n_{2_{t_1}} \tan\phi_{t_1} - n_{2_{t_2}} \tan\phi_{t_2}}$ which is defined by the ratio of the tracking accelerations in the vertical and horizontal directions, by T. If $(x_0, y_0) = \left(\frac{Uf}{W}, \frac{Vf}{W}\right)$ is the FOE, the above equation becomes $\frac{y_0}{x_0} = \frac{V}{U} = T$, which is a linear constraint on the FOE. It restricts the location of the FOE to a straight line passing through the origin of the image coordinate system with slope T (see Figure 10).

7.2. Patterns of normal flow

Since the tracking rotation is only around the x- and y-axes, it would be interesting to examine the structure of the normal flow field values not depending on rotation around the z-axis. In other words, tracking adds a rotational field but does not affect the rotation around the z-axis.

In the sequel we concentrate on the normal flow vectors not containing rotation around the z-axis, hereafter called γ-vectors. These are all the normal flow vectors perpendicular to circles with center at the origin of the image coordinate system. The lines defining the directions of such vectors pass through the origin. Let us also call a γ-vector positive if it points in the direction (x, y) (Figure 11); otherwise, its orientation is said to be negative.

First, we concentrate on the rotational component of the γ-vectors: Along the positive direction, the rotational contribution is

$$
u_{\mathrm{rot}}(r, \phi) = -A\left(\frac{r^2}{f} + f\right)\sin\phi + B\left(\frac{r^2}{f} + f\right)\cos\phi
$$

where $A = \alpha + \omega_x$, $B = \beta + \omega_y$, r is distance from the image center and the angle ϕ is

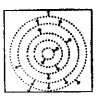


Figure 11: Positive γ-vectors.

measured from the x-axis. Thus, the rotational component of the normal flow along a vector pointing away from the image center can be described by a trigonometric function with amplitude $\max(A, B)$ and period 2π. Along the line which passes through the image center and makes angle $\phi = \arctan(B/A)$ with the x-axis, the values of the γ-vectors are zero. This line divides the plane into two halves. In one half the vectors point in the positive direction, and in the other half they point in the negative direction; in the future we simply refer to them as positive and negative vectors (Figure 12a).

We now turn our attention to the translational component of the γ-vectors: The translational component of the motion field is characterized by the location of the FOE in the image plane. The γ-vectors lie on lines passing through the image center and the optical flow values due to translation lie on lines passing through the FOE. These two lines are at right angles for all points on a circle which has the FOE and the image center as diametrical opposite points. At these points the γ vectors' translational components vanish. Thus, the geometric locus of all points where there is zero translational normal flow is a circle. The diameter of this circle is the line segment connecting the image center and the FOE. At all points inside this circle the two lines enclose an angle greater than 90° and the normal flow along the γ-vector therefore has a negative value. The normal flow values outside the circle are positive (Figure 12b).

In order to investigate the constraints associated with a general motion, the geometrical relations derived from rotation and from translation have to be combined. A circle separating the plane into positive and negative values and a line separating the plane into two halfplanes of opposite sign always intersect (in two points or one point in case the line is tangential to the circle), because both the line and the circle pass through the origin. This splits the plane into areas of only positive or only negative γ-vectors, and into areas in which the rotational and translational flows have opposite signs. In the latter areas, unless we make depth assumptions, no information is derivable (Figure 12c).

We thus obtain the following geometrical result for the case of general motion. Points in the image plane at which the gradient direction is perpendicular to circles

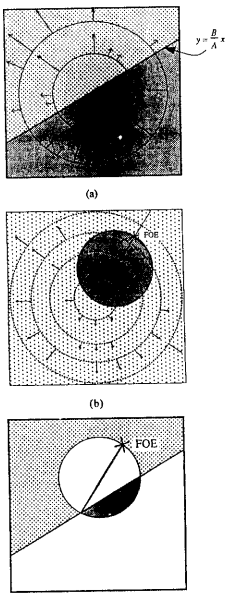


Figure 12: (a): The γ-vectors due to rotation separate the image plane into a halfplane of positive values and a halfplane of negative values. (b): The γ vectors due to translation are negative if they lie within the circle defined by the FOE and the image center and are positive at all other locations. (c) A general rigid motion defines an area of positive γ vectors and an area of negative γ-vectors. The rest of the image plane is not considered.

around the image center can be separated into two classes. For a given FOE, and for a line through the image center which represents the quotient of two of the three rotational parameters, there are two geometrically defined areas in the plane, one containing positive and one containing negative values. We call this structure on the γ-values the γ-pattern. It depends on the three parameters x_0, y_0 and $\frac{B}{A}$. If we can locate this pattern through some search then in effect we have located the position of the FOE and the value $\frac{B}{A}$. The γ-pattern depends on three parameters, but the constraints derived from fixation (previous section) reduce the search for the pattern's position to only one dimension.

Indeed, from equations (2) and (3) at the origin we have:

$$\frac{n_{1_{t_1}} f \tan\phi_{t_1} - \alpha f}{n_{2_{t_1}} f \tan\phi_{t_1} - \beta f} = -\frac{V}{U}$$

Since at the center $n_{1_{t_1}} f \tan\phi_{t_1}$ is equal to $-\omega_x f$ and $n_{2_{t_1}} f \tan\phi_{t_1}$ is equal to $-\omega_y f$, we obtain

$$\frac{\omega_x + \alpha}{\omega_y + \beta} = \frac{A}{B} = -\frac{V}{U} = -\frac{y_0}{x_0} = -T$$

or $\frac{B}{A} = -T^{-1}$ and $\frac{y_0}{x_0} = T$

In other words, tracking provides not only the line $\frac{y_0}{x_0} = T$ on which the FOE lies, but also defines the line $y = \frac{B}{A}x$ which separates positive and negative rotational flow. This reduces the search for the pattern of Figure 12c to one dimension. We simply search for a circle with diameter the segment connecting the origin with a point along the line $\frac{y}{x} = T$. This is a robust procedure as it only utilizes the sign of the normal flow. If a wide-angle lens or logarithmic retinae [22] are employed most of the directions representing the FOE lie in a bounded area of the image plane. Alternatively, in order to cover all possible cases, the search can be realized in the stereographic space [18] where the space of all orientations is bounded.

Pattern matching, since it does not utilize all values of the normal flow, may provide a set of solutions for the location of the FOE. To further narrow down the space of possible FOE location and to estimate the rotational parameters, the process of detranslation (next section) is performed.

7.3. The process of detranslation

By detranslation we refer to the process that, given the position of the FOE, selects the normal flow vectors due to rotation only. Indeed, if the location of the FOE is given, the directions of the translational motion components are also known. The translational vectors lie on lines passing through the FOE. The normal flow vectors perpendicular to these lines do not contain translational components; they have only rotational components. This can be seen from equation (1). If the selected gradient

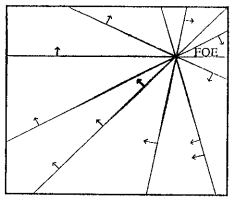


Figure 13: Normal flow vectors perpendicular to lines passing through the FOE are only due to rotation.

direction at a point (x, y) is $((y_0-y), (-x_0+x))$ the scalar product of the translational motion component and a vector in the gradient direction is zero (Figure 13).

For each of the possible solutions (x_{0_i}, y_{0_i}), $i = 1, \ldots, n$, for the FOE provided by the pattern matching of the previous section, the normal flow vectors perpendicular to the lines passing through (x_{0_i}, y_{0_i}) have to be tested to determine if they are only due to rotation (see Figure 13). This results in solving an overdetermined system of linear equations, with two unknowns, since the ratio $\frac{B}{A}$ is already known.

Indeed, suppose that we want to test if (x_{0_i}, y_{0_i}) is the correct location of the FOE. Consider all normal flow vectors $\vec{u}_{n_i} = u_{n_i}(n_{x_i}, n_{y_i})$, $i = 1, \ldots, k$, perpendicular to the lines passing through (x_{0_i}, y_{0_i}). Then,

$$u_{n_i} = \left(A\frac{xy}{f} - B\left(\frac{x^2}{f} + f\right) + Cy\right) n_{x_i} + \left(A\left(\frac{y^2}{f} + f\right) - B\frac{xy}{f} - Cx\right) n_{y_i}$$

and since $\frac{A}{B} = -T$, we have:

$$u_{n_i} = \left(-B\left(\frac{Txy}{f} + \frac{x^2}{f} + f\right) + Cy\right) n_{x_i} - \left(BT\left(\frac{y^2}{f} + f + \frac{xy}{f}\right) + Cx\right) n_{y_i} \quad i = 1, \ldots, k.$$

So, if the above k linear equations in the two unknowns B, C are consistent, then we have found a possible FOE $((x_{0_i}, y_{0_i}))$ and we have computed its corresponding rotation.

7.4. Complete derotation

Assume that the previous processes don't provide a single solution but a set of solutions $S = \{s_1, s_2, \ldots, s_n\}$ with $s_i = (x_{0_i}, y_{0_i}, \alpha_i, \beta_i, \gamma_i)$ candidate egomotion parameters. In order to eliminate all motion parameters which are not consistent with the given normal flow field, every normal flow vector has to be checked.

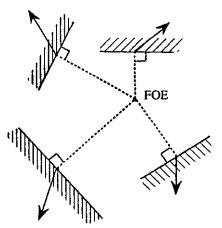


Figure 14: Normal flow vectors due to translation are constrained to line in halfplanes.

This check is performed using a "derotation" technique. For every parameter quintuple of S a possible FOE and a rotation is defined. The three rotational parameters are used to derotate the normal flow vectors by subtracting the rotational component $(u_{\text{rot}}, v_{\text{rot}})$. At every point the flow vector $(u_{\text{der}}, v_{\text{der}})$ is computed:

$$u_{\text{der}} = u_n n_x - u_{\text{rot}} n_x$$
$$v_{\text{der}} = u_n n_y - v_{\text{rot}} n_y$$

If the parameter quintuple defines the correct solution, the remaining normal flow is purely translational. Thus the corresponding optic flow field consists of vectors that all point away from one point, the FOE [13]. Since the direction of optical flow for a given FOE is known, the possible directions of the normal flow vectors can be determined. The normal flow vector at every point is confined to lie in a half plane (see Figure 14). The technique checks all points for this property and eliminates solutions that cannot give rise to the given normal flow field.

7.5. The algorithm

Assume that a rigidly moving observer is capable of tracking (with tracking velocities ω_x, ω_y) an environmental point whose image is at the origin. Then, the following algorithm outputs the observer's motion.

Step 1. The tracking acceleration provides a line $y = Tx$ on which the FOE lies, as well as the ratio $\frac{\alpha+\omega_x}{\beta+\omega_y}$ (Section 7.1).

Step 2. Using the result of the previous step, a 1-D search along the line $y = Tx$ for the pattern of Figure 12c is performed to find solutions for the FOE.

Step 3. The previous step may provide a set $\mathbf{S} = \{(x_{0_1}, y_{0_1}), (x_{0_2} y_{0_2}), \ldots, (x_{0_n}, y_{0_n})\}$. For each (x_{0_i}, y_{0_i}) we perform the process of detranslation which may have two consequences. One would be to reject (x_{0_i}, y_{0_i}) as a possible solution and the other would be to accept it with the computed rotation (A_i, B_i, C_i).

Step 4. Step 3 may provide a set S of candidate solutions for the translation and the rotation:

$$S = \{(x_{0_1}, y_{0_1}, A_1, B_1, C_1), \ldots, (x_{0_n}, y_{0_n}, A_n, B_n, C_n)\}.$$

In order to reject impossible solutions complete derotation is performed to check every single normal flow vector for consistency with the motion parameters.

8. Experiments

We have tested the technique computing object motion on synthetic imagery by using the graphics package Swivel. In this way we were able to simulate object motion as well as camera rotation. In order to analyze the robustness of the method, we evaluated the accuracy of the normal flow values in the center of the images. At every point we determined v_{act}, the projection of the known optical flow value on the gradient direction computed there. The error (err) in the normal flow values was defined as standardized difference between v_{act} and the normal flow value, v_{meas} ($err = (v_{\text{act}} - v_{\text{meas}})/v_{\text{act}}\%$). This way we computed an average error of 76.14% and a standard deviation of 179.64% for the motion sequence at the beginning of the tracking process. This constitutes a large error and is comparable to errors appearing in noisy real imagery.

The object displayed in Figure 15 moves in the direction $U/W = 4$ and $V/W = 2$, with an image motion at the center of $u = 0.004$ and $v = 0.002$ focal units, and we tracked it over a sequence of 100 images. Concerning the implementational details, we computed normal flow measurements in 10 directions in an area of 9×9 pixels at the center of the image. When testing the first module, with which parallel translation is estimated, we used a threshold of 0.0002 focal units. The method converges very quickly, usually after 2 to 3 iterations. We added rotation of growing magnitude to the object motion, and it turned out that the algorithm converges for this set-up even for relatively large rotations. (The object was 25 units away from the camera and moved with translational velocity of $U = 0.1$, $V = 0.05$, $W = 0.025$ units per time and the method converges for rotations of up to 0.3° per time unit around the x- y- and z- axis.) Some graphical representations are given: Figure 16 shows for the case of no rotation the three normal flow fields that were computed in the the 9×9 pixels large area, before convergence was achieved. In Figure 17 two maximum normal flow vector sequences are displayed (a: for no rotation, b: for rotation $\omega_x = 0.1°$, $\omega_z = 0.1°$).

Figure 15: First image in the sequence used for tracking.

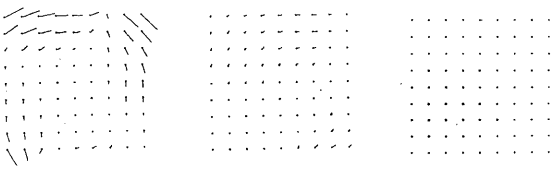
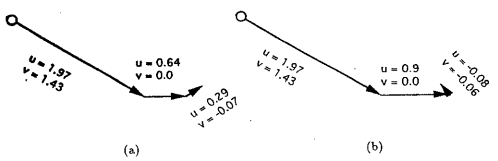


Figure 16: Normal flow fields for a tracking sequence.



(a) (b)

Figure 17: Maximum normalflow vectors for (a) no rotation and (b) rotation $\omega_x = 0.1°/\Delta t$, $\omega_z = 0.1°/\Delta t$.

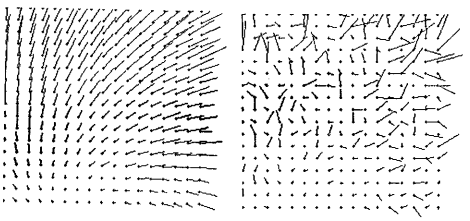
Using the estimates of parallel translation from this module and continuing with tracking over 100 steps resulted in FOE values of less than 15 % error (eg., for the case of no rotation we computed an FOE of $U/W = 4.21$ and $V/W = 1.79$). With these experiments we demonstrated that the technique to compute object motion can tolerate a large amount of noise in the input (normal flow).

Especially we showed that tracking can be successfully accomplished using only normal flow under noisy conditions and that tracking acceleration can be employed for robust parameter estimation.

Building upon a successful tracking mechanism in a second series of experiments we tested the last three modules of our egomotion recovery algorithm: pattern matching, detranslation and derotation. Concerning the implementation of these modules we took the following approach: The elimination of impossible parameters from the space of solutions involves discrimination on the basis of quantitative values. We have implemented this in the following way: Normal flow values in certain directions are selected, if they are within a tolerance interval of 10°. This relatively large degree of freedom, of course, will introduce some error, but there is a tradeoff between accuracy and the amount of data used by the algorithm. In the pattern matching and the derotation modules counting is applied to discriminate between possible and impossible solutions. The quality of the fitting, the "success rate", is measured by the number of values with correct signs normalized by the total number of selected values. The amount of rotation in the derotation module is computed through a simple linear least squares minimization and the discrimination between accepted and rejected motion parameters is based on the value of the residual.

In the pattern matching and derotation modules no quantitative use of values is made, since only the sign of the normal flow is considered. Such a limited use of data makes the modules very robust, and the correct solutions are usually found even in the presence of high amounts of noise. To give some quantitative justification of this we define the error in the normal flow at a point as a percentage of the correct vector's length. Since the sign of the vector is not affected as long as the error does not exceed the correct vector in value, our "pattern fitting" and derotation will find the correct solution in all cases of up to 100% error.

Several experiments have been performed on synthetic data. For different 3-D motion parameters normal flow fields were generated; the depth value within an interval and the gradient direction were chosen randomly. Pattern matching was tested by assuming knowledge of the lines $y = Tx$ (where the FOE lies) and $y = \frac{B}{A}x$ (which separates positive from negative rotational components) provided by the tracking constraint. The set of possible solutions was then further reduced by detranslation and derotation which were implemented as described above. In all experiments on noiseless data the correct solution was found as the best one. Figure 18 shows the optic flow field and the normal flow field for one of the generated data sets: The image size was 100×100, the FOE was at (–40,–40) and the ratio of the rotational components was $A : B : \gamma = 1 : -1 : 15$. In Figure 19 the fitting of the γ-pattern to



(a) Optical flow field. (b) Normal flow field.

Figure 18: Flow vectors for synthetic image.

the γ-vectors is displayed. Points with positive normal flow values are rendered in a light color and points with negative values are dark. Perturbation of the normal flow vectors' lengths by up to 50% did not prevent the method from finding the correct solution.

As an example of a real scene the NASA–Ames sequence[2] was chosen. The camera undergoes only translational motion, and we added different amounts of rotation: For all points at which translational motion can be found the rotational normal flow is computed, and the new position of each pixel is evaluated. The "rotated" image is then generated by computing the new grey levels through bilinear interpolation. The images were convolved with a Gaussian of kernel size 5×5 and standard deviation $\sigma = 1.4$. The normal flow was computed by using 3×3 Sobel operators to estimate the spatial derivatives in the x- and y-directions and by subtracting the 3×3 box-filtered values of consecutive images to estimate the temporal derivatives. When adding rotational normal flow on the order of a third to three times the amount of translational flow, the exact solution was always found among the best fitted parameter sets. In Figure 20 the computed normal flow vectors and the fitting of the γ-patterns for one of the "rotated" images are shown. Areas of negative normal flow vectors are marked by horizontal lines and areas of positive values with vertical lines. The ground truth for the FOE is (−5 ,−8), the focal length is 599 pixels, and the rotation between the two image frames is $\alpha = 0.0006$, $\beta = 0.0006$, and $\gamma = 0.004$. The algorithm computed the solution exactly.

[2]This is a calibrated motion sequence made public for the Workshop on Visual Motion, 1991.

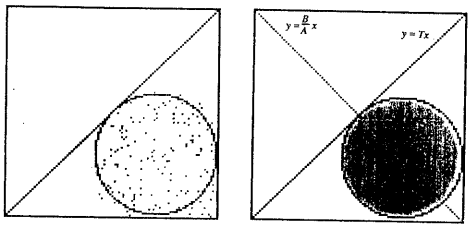
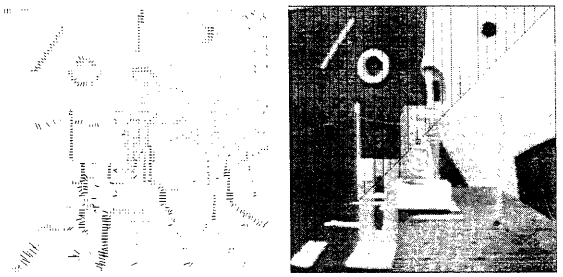


Figure 19: (a) Positive and negative γ-vectors. (b) Fitting of γ-pattern: The line $y = Tx$ on which the FOE lies and the line $y = \frac{B}{A}x$, which separates the rotational components, are found through the fixation constraint.



(a) (b)

Figure 20: Real scene: Normal flow field and fitting of γ-patterns.

9. Conclusions

It has been argued by psychologists that biological organisms use tracking in the motion estimation process. In this paper we have exploited the advantages of the tracking activity to estimate egomotion and to solve for a monocular observer the problem of computing a moving object's translational direction and its time to collision. We have presented a complete solution to this task by showing how tracking can be pursued when only normal flow measurements are used and how these parameters are of use in the 3-D motion parameter decoding strategy. The technique for estimating an object's motion consists of three modules. First, tracking is used in combination with fixation to estimate the motion components parallel to the image plane. Second, tracking serves to compute the perpendicular translational components and to estimate the FOE. The output of these modules is then employed to estimate the time to collision. A theoretical analysis of the tracking algorithm in the first module has been performed and the convergence of the method has been proved. Experimental studies have been conducted on synthetic imagery and yielded very good results.

In contrast to the first method where an object-centered coordinate system is used, for egomotion estimation a camera-centered coordinate system is more appropriate. The main difference between the two algorithms described in the paper lies in the fact that object motion is computed from local data while egomotion estimation is based on global data. The technique uses data from all parts of the image plane and exploits geometric relations that are characteristic of a normal flow field due to rigid motion. The algorithms can be regarded as a search technique in a parameter space where the use of fixation and tracking, along with an appropriate selection of normal flow values, is used to reduce the dimensionality of the motion estimation problem from five dimensions to one.

The theoretical analysis and the experiments described in this paper demonstrate that the introduced algorithms have the potential of being implemented in real hardware active vision systems, such as the ones described in [8, 17].

References

[1] G. Adiv, "Determining 3-D motion and structure from optical flow generated by several moving objects", *IEEE Trans. PAMI* **7**, 1985, 384–401.

[2] J. Y. Aloimonos, "Purposive and qualitative active vision", *Proc. DARPA Image Understanding Workshop*, 1990, 816–828.

[3] J. Aloimonos and C.M.Brown, "Direct processing of curvilinear sensor motion from a sequence of perspective images", *Proc. Workshop on Computer Vision: Representation and Control*, 1984, 72–77.

[4] J. Aloimonos and C.M.Brown, "On the kinetic depth effect", *Biological Cybernetics* **60**, 1989, 445–455.

[5] J. Aloimonos, I. Weiss, and A. Bandopadhay, "Active vision", *Int'l. Journal of Computer Vision* **2**, 1988, 333–356.

[6] R. Bajcsy, "Active perception vs. passive perception", *Proc. IEEE Workshop on Computer Vision*, 1985, 55–59.

[7] D.H. Ballard, "Animate vision", *Artificial Intelligence* **48**, 1991, 57–86.

[8] D.H. Ballard and C.M. Brown, "Principles of animate vision", *CVGIP: Image Understanding* **56**, 1992.

[9] A. Bandopadhay and D.H. Ballard, "Egomotion perception using visual tracking", *Computational Intelligence* **7**, 1991, 39–47.

[10] C. Fermüller and Y. Aloimonos, "Estimating 3-D motion from image gradients", Technical Report CAR-TR-564, Center for Automation Research, University of Maryland, College Park, 1991.

[11] C. Fermüller and Y. Aloimonos, "Tracking facilitates 3-D motion estimation", Technical Report CAR-TR-618, Center for Automation Research, University of Maryland, College Park, 1992.

[12] B. Horn and B. Schunck, "Determining optical flow", *Artificial Intelligence* **17**, 1981, 185–203.

[13] B.K.P. Horn and E.J. Weldon, "Computationally efficient methods of recovering translational motion", *Proc. Int'l. Conference on Computer Vision*, 1987, 2–11.

[14] H. C. Longuet-Higgins and K. Prazdny, "The interpretation of moving retinal images", *Proc. Royal Soc. London B* **208**, 1984, 385–397.

[15] D. Marr, *Vision*, W.H. Freeman, San Francisco, 1982.

[16] S. Negahdaripour, Ph.D. Thesis, MIT Artificial Intelligence Laboratory, 1986.

[17] K. Pahlavan and J.-O. Eklundh, "A head-eye system—Analysis and design", *CVGIP: Image Understanding* **56**, 1992.

[18] F.W. Sohon, *The Stereographic Projection*, Chelsea, NY, 1941.

[19] M.E. Spetsakis and J. Aloimonos, "Optimal visual motion estimation", *IEEE Trans. PAMI* **14**, 1992, 959–964.

[20] M.E. Spetsakis and Y. Aloimonos, "Structure from motion using line correspondences", *Int'l. J. Computer Vision* **4**, 1990, 171–183.

[21] M A. Taalebi-Nezhaad, "Direct recovery of motion and shape in the general case by fixation", *Proc. DARPA Image Understanding Workshop*, 1990, 284–291.

[22] M. Tistarelli and G. Sandini, "Dynamic aspects in active vision", *CVGIP: Image Understanding* **56**, 1992.

[23] R.Y. Tsai and T.S. Huang, "Uniqueness and estimation of three-dimensional motion parameters of rigid objects with curved surfaces", *IEEE Trans PAMI* **6**, 1984, 13–27.

[24] S. Ullman, *The Interpretation of Visual Motion*, MIT Press, Cambridge, MA, 1979.

[25] A. Verri and T. Poggio, "Against quantitative optic flow", *Proc. IEEE Int'l. Conference on Computer Vision*, 1987.

[26] A.M. Waxman, B. Kamgar-Parsi and M. Subbarao, "Closed-form solutions to image flow equations for 3-D structure and motion", *Int'l. Journal of Computer Vision* **1**, 1987, 239–258.