Camera Networks for Building Shape Models from Video

???
Computer Vision Laboratory
University of Maryland
College Park, MD 20742-3275

E-mail: {???}@cfar.umd.edu

Abstract

Fundamental discoveries do not necessarily rely on exploring new landscapes but on employing new eyes—thus indicated Marcel Proust, and the wisdom of his metaphor regarding the power of new eyes is strongly reflected in ancient Greek mythology. Recall Argus, the hundred-eyed guardian of Hera, the goddess of Olympus, who alone defeated a whole army of Cyclopes, one-eyed giants. Similar ideas appear in this paper which shows how to use existing cameras in various ways to create new cameras—new ways to see the world. Autonomous or semi-autonomous intelligent systems, in order to function appropriately, need to create models of their environment, i.e., models of space-time. These are descriptions of objects and scenes and descriptions of changes of space over time, that is, events and actions. Despite the large amount of research on this problem, as a community we are still far from developing robust descriptions of a system's spatiotemporal environment using video input (image sequences). Undoubtedly, some progress has been made regarding the understanding of estimating the structure of visual space, but it has not led to solutions to specific applications. There is, however, an alternative approach which is in line with today's "zeitgeist." The vision of artificial systems can be enhanced by providing them with new eyes. If conventional video cameras are put together in various configurations, new sensors can be constructed that have much more power and the way they "see" the world makes it much easier to solve problems of vision. This research is motivated by examining the wide variety of eye design in the biological world and obtaining inspiration for an ensemble of computational studies that relate how a system sees to what that system does (i.e., relating perception to action). This, coupled with the geometry of multiple views that has flourished in terms of theoretical results in the past few years, points to new ways of constructing powerful imaging devices which suit particular tasks in robotics, visualization, video processing, virtual reality and various computer vision applications, better than conventional cameras. This paper presents a new sensor that we built using common video cameras and shows its superiority with regard to developing models of space from long video sequences.

1. Introduction: Models of space-time

Technological advances make it possible to arrange video cameras in some space, connect them with a high-speed network and collect synchronized video. Such developments open new avenues in many areas, making it possible to address, for the first time, a variety of applications in surveillance and monitoring, graphics and visualization, robotics and augmented reality. But as the need for applications grows, there does not yet exist a clear idea on how to put together many cameras for solving a variety of problems. That is, the mathematics of multiple-view vision is not yet understood in a way that relates the configuration of the camera network to the task under consideration. Existing approaches treat almost all problems as multiple stereo problems, thus missing important information hidden in the multiple videos. The goal of this paper is to provide the first steps in filling the gap described above. We consider a multi-camera network as a new eye and we perform a comparative analysis of these new eyes with traditional video cameras. To achieve this we concentrate here on developing models of space. The exposition is such that it motivates the new eyes, by first describing the problems of developing models of shape using a common video camera and pointing out inherent difficulties.

Images, for a standard pinhole camera, are formed by central projection on a plane (Figure 1a). The focal length is f and the coordinate system $OXYZ$ is attached to the camera, with Z being the optical axis, perpendicular to the image plane.

Image points are represented as vectors $\mathbf{r} = [\mathbf{x}, \mathbf{y}, \mathbf{f}]^{\mathrm{T}}$, where x and y are the image coordinates of the point in the coordinate system oxy, with $ox||OX$, $oy||OY$ and O the intersection of the axis OZ with the image plane, and f is the focal length in pixels. A scene point $\mathbf{R}$ is projected onto the image point

$$\mathbf{r} = \mathbf{f}\,\frac{\mathbf{R}}{\mathbf{R}\cdot\hat{\mathbf{z}}} \tag{1}$$

where $\hat{\mathbf{z}}$ is the unit vector in the direction of the Z axis.

In general, when a scene is viewed from two positions, there are two concepts of interest: (a) The 3D transformation relating the two viewpoints. This is a rigid motion transformation, consisting of a translation and a rotation (six degrees of freedom). When the viewpoints are close together, this transformation is modeled by the 3D motion of the eye (or camera). (b) The 2D transformation relating the pixels in the two images, i.e., a transformation that given a point in the first image maps it onto its corresponding one in the second image (that is, these two points are the projections of the same scene point). When the viewpoints are close together, this transformation amounts to a vector field denoting the velocity of each pixel, called an image motion field. Perfect knowledge of both transformations described above leads to perfect knowledge of models of space. Since knowing exactly how the two viewpoints and the images are related provides the exact position of each scene point in space. Thus, a key to the basic problem of building models of space is the recovery of the two transformations described before and any difficulty in building such models can be traced to the difficulty of estimating these two transformations. What are the limitations in achieving this task?

2. Inherent limitations: Image motion and 3D motion

If $\mathbf{r}$ is an image point (x, y, f), the projection of the motion vector $\dot{\mathbf{r}}$ on the gradient $\mathbf{n}$ at this point is the well known normal flow u_n, with

$$u_n = \dot{\mathbf{r}}\cdot\mathbf{n}. \tag{2}$$

where $\mathbf{n}$ is a unit vector at an image point denoting the orientation of the gradient at that point. The normal flow is a robust measurement from a moving image and can be computed locally and in parallel. To compute then the values of the flow, one would need to utilize the normal flow values along with additional constraints.

All approaches start with the normal flow measurements and then fit some parametric model for the flow or employ a regularization scheme. In both cases there are problems because of the unknown location of depth discontinuities. If we knew where the discontinuities are, estimating flow would be easy, but to know where the discontinuities are we need first to find 3D motion and use it to find depth—but to do that we need to know the values of the flow! The whole problem is clearly a chicken/egg problem.

There exists an additional reason causing incorrect flow estimates that only recently was understood [9], and is related to the image texture. It has to do with the statistical difficulty of integrating local, 1D motion signals into 2D image velocity measurements. Any procedure for estimating image motion has to start with normal flow measurements, that is, the image motion component perpendicular to local edges. It has been shown [9] that when these local measurements are combined in a neighborhood to produce image motion, an estimate of flow is obtained which is biased. The estimated value depends on the distribution of image gradients, the actual flow and the error in the normal flow. This is strikingly observed in the Ouchi illusion (Figure 2). The pattern in Figure 2 has the surprising property that small motions can cause illusory relative motion between the inset and background regions.[1] The reason for this illusion is that for the particular spatial gradient distributions of the Ouchi pattern, the bias in the estimation of flow is highly pronounced, giving rise to a large difference in the velocity estimates in the two regions. Situations like this occur too often in real imagery (neighboring textures of different orientation). Thus, there are two basic problems with the estimation of correspondence, i.e., the motion field. One is geometric, related to scene discontinuities, and the other is statistical, related to how the image texture looks.

Regarding 3D motion estimation, there exists a veritable cornucopia of techniques for finding 3D motion from a video sequence. Almost all techniques are based on the so-called epipolar constraint, which shows how the motion of image points is related to 3D rigid motion and the scene. This constraint, at each image point $\mathbf{r}$, is written as $(\mathbf{t}\times\mathbf{r})\cdot(\dot{\mathbf{r}}+\boldsymbol{\omega}\times\mathbf{r}) = \mathbf{0}$ [3].

One is interested in the estimates of translation $\hat{\mathbf{t}}$ and rotation $\hat{\boldsymbol{\omega}}$ which best satisfy the epipolar constraint at every point $\mathbf{r}$ according to some criteria of deviation. Usually the Euclidean norm is considered leading to the minimization of function.

$$E_{ep} = \int\!\!\int_{\text{image}} \left[(\hat{\mathbf{t}}\times\mathbf{r})\cdot(\dot{\mathbf{r}}+\hat{\boldsymbol{\omega}}\times\mathbf{r})\right]^2 d\mathbf{r} \tag{3}$$

The reason for the large amount of literature is that the problem is very difficult. One main reason for this has to do with the apparent confusion between translation and rotation in the motion field. This is easy to understand at an intuitive level. If we look straight ahead at a shallow scene, whether we rotate around our vertical axis or translate parallel to the

[1] The effect can be attained with small retinal motions or a slight jiggling of the paper and is robust over large changes in the patterns, frequencies and boundary shapes.

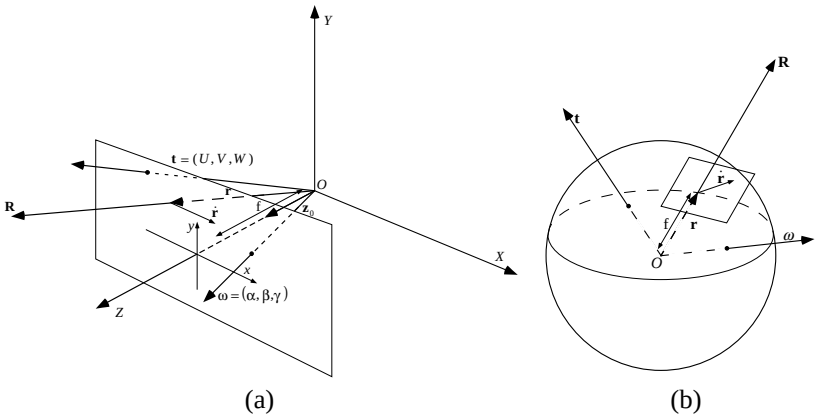

Figure 1. Image formation on the plane (a) and on the sphere (b). The system moves with a rigid motion with translational velocity $\mathbf{t}$ and rotational velocity $\boldsymbol{\omega}$. Scene points $\mathbf{R}$ project onto image points $\mathbf{r}$ and the 3D velocity $\dot{\mathbf{R}}$ of a scene point is observed in the image as image velocity $\dot{\mathbf{r}}$.

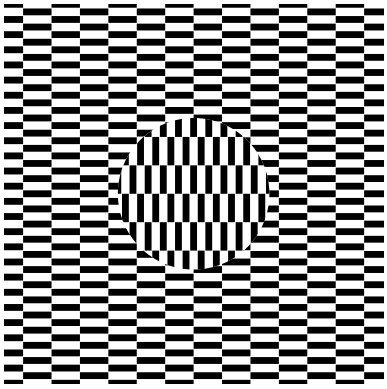

Figure 2. A pattern similar to one by Ouchi.

scene, the motion field at the center of the image is very similar in the two cases. Thus, for example, translation along the x axis is confused with rotation around the y axis. The basic understanding of this confusion has attracted few investigators over the years [2, 3]. In [6, 8] a geometrical statistical analysis of the problem has been conducted. On the basis of (3) the expected value of E_{ep} has been formulated as a five-dimensional function of the motion parameters (two dimensions for $\mathbf{t}/|\mathbf{t}|$ and three for $\boldsymbol{\omega}$). Independent of specific estimators the topographic structure of the surface defined by this function explains the behavior of 3D-motion estimation. Intuitively speaking, it turns out that the minima of this function lie in a valley. This is a cause for inherent instability because, in a real situation, any point on that valley or flat area could serve as the minimum, thus introducing errors in the computation.

In particular, the result obtained can be formulated as follows: Denote the five unknown motion parameters as (x_0, y_0) (direction of translation) and (α, β, γ) (rotation). Then, *no matter how 3D motion is estimated from the motion field*, the expected solution will contain errors $(x_{0_\epsilon}, y_{0_\epsilon})$, $(\alpha_\epsilon, \beta_\epsilon, \gamma_\epsilon)$ that satisfy two constraints:

(a) The orthogonality constraint: $\dfrac{x_{0_\epsilon}}{y_{0_\epsilon}} = -\dfrac{\beta_\epsilon}{\alpha_\epsilon}$

(b) The line constraint: $\dfrac{x_0}{y_0} = \dfrac{x_{0_\epsilon}}{y_{0_\epsilon}}$

In addition, we must also have $\gamma_\epsilon = 0$. The result states that the solution contains errors that are mingled and create a confusion between rotation and translation that cannot be cleared up, with the exception of the rotation around the optical axis (γ). The errors may be small or large, but their expected value will always satisfy the above conditions. Although the 3D-motion estimation approaches described above may provide answers that could be sufficient for various navigation tasks, they cannot be used for deriving object models because the depth Z that is computed will be distorted [1].

3. Looking at the world

We are interested in space and action descriptions that can be extracted from visual data. This requires that there exists an eye or device imaging the scene. All along we took it for granted that our basic device was a camera-type eye, that is, a common video camera whose basic principle is the pinhole model, but there was no particular reason to make this assumption.

An examination of the design of eyes in the biological world reveals a very wide variety. The mechanisms organisms have evolved for collecting photons and forming images that they use to perform various actions in their environment depend on a number of factors. Chief among these are the individual organism's computational capacity and the tasks that the organism performs. Michael Land, a prominent British zoologist and the world's foremost expert on the science of eyes, has provided a landscape of eye evolution. Considering evolution as a mountain, with the lower hills representing the earlier steps in the evolutionary ladder, and the highest peaks representing the later stages of evolution, the situation is pictured in Figure 3 [4]. It has been estimated that eyes have evolved no fewer than forty times,

independently, in diverse parts of the animal kingdom. In some cases, these eyes use radically different principles and the "eye landscape" of Figure 3 shows nine basic types of eyes. Eyes low in the hierarchy (such as the nautilus' pinhole eye or the marine snail eye) make very crude images of the world, but at higher levels of evolution we find different types of compound eyes and camera-type eyes (like the ones we use) such as the corneal eyes of land vertebrates and fish eyes.

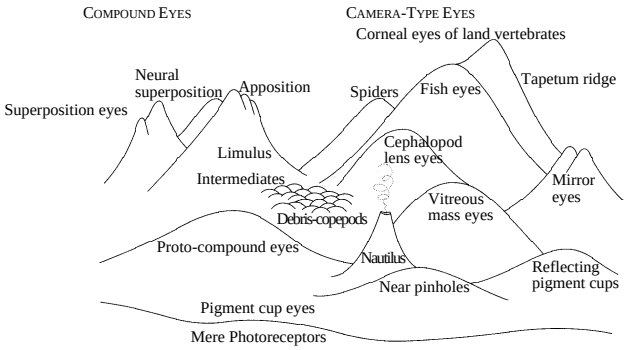

Figure 3. Michael Land's landscape of eye evolution.

Inspiration for our work has come from the compound eyes of insects which are particularly intriguing, especially in view of the fact that insects compute excellently 3D motion. Their lives depend on their ability to fly with precision through cluttered environments, avoid obstacles and land on demand on surfaces oriented in various ways. In addition, they perform these tasks with minimal memory and computational capacity, much less than an average personal computer of today. Could it be possible that much of their success emanates from the special construction of their eyes?[2]

4. New eyes

Why is it that biological systems that need to fly and thus require good estimates of 3D motion (insects, birds) have

[2]Compound eyes exist in several varieties, and can be classified in two categories, the *apposition* and *superposition* ones. The apposition eye is built as a dense cluster of long, straight tubes radiating out in all directions as from the roof of a dome. Each tube is like a gun sight which sees only a small part of the world in its own direct line of fire. Thus, rays coming from other parts of the wall are prevented by the walls of the tube and the backing of the dome from hitting the back of the tube where the photocells are (Figure 4). In practice, each of the little tube eyes called ommatidia, is a bit more than a tube. It has its own private lens and its own private retina of about half a dozen photocells. The ommatidium works like a long, poor quality, camera eye. Superposition compound eyes, on the other hand, do not trap rays in tubes. They allow rays that pass through the lens of one ommatidium to be picked up by a neighboring ommatidium's photocells. There is an empty, transparent zone shared by all ommatidia. The lenses of all ommatidia conspire to form a single image on a shared retina which is put together from the light-sensitive cells of all the ommatidia.

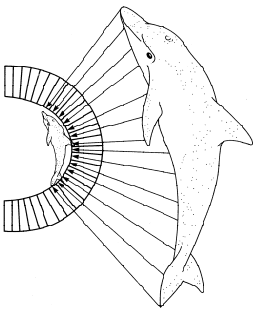

Figure 4. (Adapted from [4].) Example of the principle of the apposition compound eye, forming the image of a dolphin. The arrows don't represent rays (which would be bent by the lenses) but mappings from the points of the object in view (a dolphin) to points in the bottoms of the tubes.

panoramic vision implemented either as a compound eye or by placing camera-type eyes on opposite sides of the head? This is a fascinating question that has remained open since the time of the pioneer investigator, Sigmund Exner, at the beginning of this century. The obvious answer is, of course, that flying systems should perceive the whole space around them—thus panoramic vision emerged. There is, however, a deeper mathematical reason and it has to with the ability of a system to estimate 3D motion when it analyzes panoramic images, as shown in this section. Put simply, a spherical eye (360 degree field of view) is superior to a planar eye (restricted field) with regard to 3D motion estimation. Recall from Section 2 that, given a sequence of images, 3D motion is estimated by minimizing function E that represents deviation from the epipolar constraint. It was shown that in the case of images captured by a planar eye (e.g., a common video camera), this function has a special topography which is such that the errors in the motion are mingled, causing confusion between rotation and translation and thus producing a wrong result. If, however, the field of view goes to 360 degrees, the topography of the surface drastically changes with the minimum clearly standing out in most cases. Panoramic vision is modeled by projecting onto a sphere, with the sphere's center as the center of projection (Figure 1b). In this case, the image $\mathbf{r}$ of any point $\mathbf{R}$ is $\mathbf{r} = \frac{\mathbf{Rf}}{|\mathbf{R}|}$, with R being the norm of $\mathbf{R}$ (the range), and the image motion is

$$\dot{\mathbf{r}} = \frac{1}{|\mathbf{R}|f}\left((\mathbf{t}\cdot\mathbf{r})\,\mathbf{r} - \mathbf{t}\right) - \boldsymbol{\omega}\times\mathbf{r} = \frac{\mathbf{1}}{\mathbf{R}}\,\mathbf{u}_{\mathrm{tr}}(\mathbf{t}) + \mathbf{u}_{\mathrm{rot}}(\boldsymbol{\omega}). \tag{4}$$

The function E_{ep} representing deviation from the epipolar constraint on the sphere has the exact same form as in the plane for our nomenclature. We integrate over the range R

within an interval bounded by $R_{\min}$ and $R_{\max}$ and obtain

$$E_{ep} = \int_{R_{\min}}^{R_{\max}} \iint_{\text{sphere}} \left\{ \left(\frac{\mathbf{r} \times (\mathbf{r} \times \mathbf{t})}{R} - (\boldsymbol{\omega}_\epsilon \times \mathbf{r}) \right) \cdot \left(\hat{\mathbf{t}} \times \mathbf{r} \right) \right\}^2 dA\, dR$$

where A refers to a surface element. Due to the sphere's symmetry, for each point $\mathbf{r}$ on the sphere, there exists a point with coordinates $-\mathbf{r}$. Since $\mathbf{u}_{\text{tr}}(\mathbf{r}) = \mathbf{u}_{\text{tr}}(-\mathbf{r})$ and $\mathbf{u}_{\text{rot}}(\mathbf{r}) = -\mathbf{u}_{\text{rot}}(-\mathbf{r})$, when the integrand is expanded the product terms integrated over the sphere vanish. Thus

$$E_{ep} = \int_{R_{\min}}^{R_{\max}} \iint_{\text{sphere}} \left\{ \frac{\left((\mathbf{t} \times \hat{\mathbf{t}}) \cdot \mathbf{r} \right)^2}{R^2} + \left((\boldsymbol{\omega}_\epsilon \times \mathbf{r}) \cdot \left(\hat{\mathbf{t}} \times \mathbf{r} \right) \right)^2 \right\} dA\, dR$$

(a) Assuming that translation $\hat{\mathbf{t}}$ has been estimated, the $\boldsymbol{\omega}_\epsilon$ that minimizes E_{ep} is $\boldsymbol{\omega}_\epsilon = 0$, since the resulting function is non-negative quadratic in $\boldsymbol{\omega}_\epsilon$ (minimum at zero). The difference between sphere and plane is already clear. In the spherical case, as shown here, if an error in the translation is made we do not need to compensate for it by making an error in the rotation ($\boldsymbol{\omega}_\epsilon = 0$), while in the planar case we need to compensate to ensure that the orthogonality constraint is satisfied!

(b) Assuming that rotation has been estimated with an error $\boldsymbol{\omega}_\epsilon$, what is the translation $\hat{\mathbf{t}}$ that minimizes E_{ep}? Since R is assumed to be uniformly distributed, integrating over R does not alter the form of the error in the optimization. Thus, E_{ep} consists of the sum of two terms:

$$K = K_1 \iint_{\text{sphere}} \left((\mathbf{t} \times \hat{\mathbf{t}}) \cdot \mathbf{r} \right)^2 dA \quad \text{and}$$

$$L = L_1 \iint_{\text{sphere}} \left((\boldsymbol{\omega}_\epsilon \times \mathbf{r}) \cdot \left(\hat{\mathbf{t}} \times \mathbf{r} \right) \right)^2 dA,$$

where K_1, L_1 are multiplicative factors depending only on $R_{\min}$ and $R_{\max}$. For angles between $\mathbf{t}, \hat{\mathbf{t}}$ and $\hat{\mathbf{t}}, \boldsymbol{\omega}_\epsilon$ in the range of 0 to $\pi/2$, K and L are monotonic functions. K attains its minimum when $\mathbf{t} = \hat{\mathbf{t}}$ and L when $\hat{\mathbf{t}} \perp \boldsymbol{\omega}_\epsilon$. Fix the distance between $\mathbf{t}$ and $\hat{\mathbf{t}}$ leading to a certain value K, and change the position of $\hat{\mathbf{t}}$. L takes its minimum when $(\mathbf{t} \times \hat{\mathbf{t}}) \cdot \boldsymbol{\omega}_\epsilon = 0$, as follows from the cosine theorem. Thus E_{ep} achieves its minimum when $\hat{\mathbf{t}}$ lies on the great circle passing through $\mathbf{t}$ and $\boldsymbol{\omega}_\epsilon$, with the exact position depending on $|\boldsymbol{\omega}_\epsilon|$ and the scene in view.

(c) For the general case where no information about rotation or translation is available, we study the subspaces where E_{ep} changes the least at its absolute minimum, i.e., we are again interested in the direction of the smallest second derivative at 0. For points defined by this direction we calculate, using Maple, $\mathbf{t} = \hat{\mathbf{t}}$ and $\boldsymbol{\omega}_\epsilon \perp \mathbf{t}$.

5. What if correspondence is not available?

The preceding sections investigated the differences between camera-type eyes (restricted field of view) and spherical eyes (full field of view) with regard to 3D motion estimation, when an estimate of correspondence or flow was available. One may wonder how this comparative analysis becomes when correspondence is not available, but all we have at our disposal is the normal flow. This case is harder to analyze and we provide here results from proofs that appeared recently [6, 7].

If normal flow is given, the only available constraint is scalar equation (2), along with the inequality $Z > 0$ which states that since the surface in view is in front of the eye its depth must be positive. Substituting (4) into (2) and solving for the estimated depth $\hat{Z}$ or range $\hat{R}$, we obtain for a given estimate $\hat{\mathbf{t}}, \hat{\boldsymbol{\omega}}$ at each point $\mathbf{r}$:

$$\hat{Z}(\text{or } \hat{R}) = \frac{\mathbf{u}_{\text{tr}}(\hat{\mathbf{t}}) \cdot \mathbf{n}}{(\dot{\mathbf{r}} - \mathbf{u}_{\text{rot}}(\hat{\boldsymbol{\omega}})) \cdot \mathbf{n}}. \tag{5}$$

Substituting into (5) the value of $\dot{\mathbf{r}}$ from (4) gives

$$\hat{Z}(\text{or } \hat{R}) = \frac{\mathbf{u}_{\text{tr}}(\hat{\mathbf{t}}) \cdot \mathbf{n}}{\left(\frac{\mathbf{u}_{\text{tr}}(\mathbf{t})}{Z(\text{or } R)} - \mathbf{u}_{\text{rot}}(\boldsymbol{\omega}_\epsilon) \right) \cdot \mathbf{n}}$$

with $\boldsymbol{\omega}_\epsilon = \boldsymbol{\omega} - \hat{\boldsymbol{\omega}}$. This equation shows that for every $\mathbf{n}$ and $\mathbf{r}$ a range of values for Z (or R) is obtained which result in negative estimates of $\hat{Z}$ (or $\hat{R}$). Thus for each direction $\mathbf{n}$, considering all image points $\mathbf{r}$, we obtain a volume in space corresponding to negative depth estimates. The sum of all these volumes for all directions is termed the "negative depth" volume, and calculating 3D motion in this case amounts to minimizing this volume. Minimization of this volume provides conditions for the errors in the motion parameters.

Applying this analysis to the plane and the sphere provides results that are shown in Table 1 along with a summary of the epipolar minimization case.

6. Eyes from eyes

The preceding results demonstrate the advantages of spherical eyes for the process of 3D motion estimation. Table 1 lists the eight out of ten cases which lead to clearly defined error configurations. It shows that 3D motion can be

Table 1. Summary of results

	Spherical Eye	Camera-type Eye
Epipolar minimization, given optic flow	(a) Given a translational error $\mathbf{t}_\epsilon$, the rotational error $\boldsymbol{\omega}_\epsilon = 0$. (b) Without any prior information, $\mathbf{t}_\epsilon = 0$ and $\boldsymbol{\omega}_\epsilon \perp \mathbf{t}$.	(a) For a fixed translational error $(x_{0_\epsilon}, y_{0_\epsilon})$, the rotational error $(\alpha_\epsilon, \beta_\epsilon, \gamma_\epsilon)$ is of the form $\gamma_\epsilon = 0$, $\alpha_\epsilon/\beta_\epsilon = -x_{0_\epsilon}/y_{0_\epsilon}$ (b) Without any a priori information about the motion, the errors satisfy $\gamma_\epsilon = 0$, $\alpha_\epsilon/\beta_\epsilon = -x_{0_\epsilon}/y_{0_\epsilon}$, $x_0/y_0 = x_{0_\epsilon}/y_{0_\epsilon}$
Minimization of negative depth volume, given normal flow	(a) Given a rotational error $\boldsymbol{\omega}_\epsilon$, the translational error $\mathbf{t}_\epsilon = 0$. (b) Without any prior information, $\mathbf{t}_\epsilon = 0$ and $\boldsymbol{\omega}_\epsilon \perp \mathbf{t}$.	(a) Given a rotational error, the translational error is of the form $-x_{0_\epsilon}/y_{0_\epsilon} = \alpha_\epsilon/\beta_\epsilon$ (b) Without any error information, the errors satisfy $\gamma_\epsilon = 0$, $\alpha_\epsilon/\beta_\epsilon = -x_{0_\epsilon}/y_{0_\epsilon}$, $x_0/y_0 = x_{0_\epsilon}/y_{0_\epsilon}$

estimated more accurately with spherical eyes. Depending on the estimation procedure used—and systems might use different procedures for different tasks—either the translation or the rotation can be estimated very accurately. For planar eyes, this is not the case, as for all possible procedures there exists confusion between the translation and rotation. The error configurations also allow systems with inertial sensors to use more efficient estimation procedures. If a system utilizes a gyrosensor which provides an approximate estimate of its rotation, it can employ a simple algorithm based on the negative depth constraint for only translational motion fields to derive its translation and obtain a very accurate estimate. Such algorithms are much easier to implement than algorithms designed for completely unknown rigid motions, as they amount to searches in 2D as opposed to 5D spaces [5]. Similarly, there exist computational advantages for systems with translational inertial sensors in estimating the remaining unknown rotation.

Since it turns out that spherical eyes such as the ones of insects, or, in general, panoramic vision provides much better capability for 3D motion estimation, and since our problem of building accurate space and action descriptions depends on accurate 3D motion computation, it makes sense to reconsider what the eye for our problem should be. There are a few ways to create panoramic vision cameras, and the recent literature is rich in alternative approaches, but there is a way to take advantage of both the panoramic vision of flying systems and the high resolution vision of primates. An eye like the one in Figure 5, assembled from a few video cameras arranged on the surface of a sphere,[3] can easily estimate 3D motion since, while it is moving, it is sampling a spherical motion field!

Figure 5. A compound-like eye composed of conventional video cameras.

An eye like the one in Figure 5 not only has panoramic properties, eliminating the rotation/translation confusion, but it has the unexpected benefit of making it easy to esti-

[3] Like a compound eye with video cameras replacing ommatidia

mate image motion with high accuracy. Any two cameras with overlapping fields of view also provide high-resolution stereo vision, and this collection of stereo systems makes it possible to locate a large number of depth discontinuities. It is well known that, given scene discontinuities, image motion can be estimated very accurately. As a consequence, the eye in Figure 5 is very well suited to developing accurate models of the world.

There is a very large number of ways in which one can utilize multiple videos like the ones captured by the cameras of the sensor in Figure 5 for recovering 3D structure and motion. The obvious ones include: (a) treat the flow fields close to the center of each camera approximately as parts of a spherical motion field and apply algorithms such as those in [6]; (b) perform epipolar minimization in each video while enforcing the constraints relating the motions of different cameras comprising the sensor. The results of Table 1 can serve as a guide for choosing particular algorithmic procedures, e.g., should rotation or translation be estimated first, or should all parameters be estimated simultaneously, depending on whether epipolar or negative depth minimization is used, depending on whether inertial sensors are available, etc.

To summarize, a full visual field provides 3D motion very accurately, and thus very good models of the world. Existing sensors for capturing panoramic images (such as [10]) are not adequate for this problem due to low resolution. One would need a high-resolution spherical field of view. As this is currently technologically impossible, we resort to sampling the whole visual field with high resolution. See, for example, the sensor in Figure 6 (called the Argus eye), built in our laboratory, consisting of six cameras looking in different directions. If all cameras shared a common nodal point, then the cameras would sample parts of a sphere. When this is not true, a calibration is required.[4] Knowledge of the rigid transformations relating the difference camera coordinate systems, allows 3D motion and structure estimation through the use of all videos.

Figure 6. The Argus eye.

A new algorithm we developed for the eye in Figure 6 is based on the analysis presented in Section 2. Recall that when 3D motion is estimated using a common video camera, the expected errors $x_{0_\epsilon}, y_{0_\epsilon}$ in translation and $\alpha_\epsilon, \beta_\epsilon, \gamma_\epsilon$ in rotation satisfy the orthogonality and line constraints and the constraint $\gamma_\epsilon = 0$. Consider the six videos collected as the Argus eye moves in some space. Recall that the rotational velocity is the same for all cameras (only the translation differs). The algorithm proceeds by finding all motion parameters in each video and keeping only the value of γ. Thus we have the projection of the rotational vector on the optical axis of each camera. From this, the rotation is computed and subsequently estimation of the translation is easy. Using the estimated 3D motion the shape of the scene can be estimated. Remarkable results are obtained and described in [REF].

[4] Due to lack of space, we do not describe the calibration step.

References

[1] L. Cheong, C. Fermüller, and Y. Aloimonos. Effects of errors in the viewing geometry on shape estimation. *Computer Vision and Image Understanding*, 71:356–372, 1998.

[2] K. Daniilidis. *On the Error Sensitivity in the Recovery of Object Descriptions*. PhD thesis, Department of Informatics, University of Karlsruhe, Germany, 1992. In German.

[3] K. Daniilidis and M. E. Spetsakis. Understanding noise sensitivity in structure from motion. In Y. Aloimonos, editor, *Visual Navigation: From Biological Systems to Unmanned Ground Vehicles*, Advances in Computer Vision, chapter 4. Lawrence Erlbaum Associates, Mahwah, NJ, 1997.

[4] R. Dawkins. *Climbing Mount Improbable*. Norton, New York, 1996.

[5] C. Fermüller and Y. Aloimonos. Direct perception of three-dimensional motion from patterns of visual motion. *Science*, 270:1973–1976, 1995.

[6] C. Fermüller and Y. Aloimonos. Ambiguity in structure from motion: Sphere versus plane. *International Journal of Computer Vision*, 28:137–154, 1998.

[7] C. Fermüller and Y. Aloimonos. Geometry of eye design: Biology and technology. Technical Report CAR-TR-901, Center for Automation Research, University of Maryland, 1998.

[8] C. Fermüller and Y. Aloimonos. What is computed by structure from motion algorithms? In *Proc. European Conference on Computer Vision*, pages 359–375, Freiburg, Germany, 1998.

[9] C. Fermüller, R. Pless, and Y. Aloimonos. The Ouchi illusion as an artifact of biased flow estimation. *Vision Research*, 1999. In press.

[10] S. Nayar. Catadioptric omnidirectional camera. In *Proc. IEEE Conference on Computer Vision and Pattern Recognition*, pages 482–488, Puerto Rico, 1997.