LOCAL AND GLOBAL INTERPRETATION OF MOVING IMAGES

G.L.Scott

June 1986

Serial No. CSRP.083

Cognitive Studies Programme,
School of Cognitive Science
The University of Sussex
Brighton BN1 9QN

LOCAL AND GLOBAL INTERPRETATION OF MOVING IMAGES

G. L. Scott
Cognitive Studies Programme

June 1986

Thesis submitted to the University of Sussex for the degree of
Doctor of Philosophy

UNIVERSITY OF SUSSEX

Guy Lindsay Scott D. Phil

LOCAL AND GLOBAL INTERPRETATION OF MOVING IMAGES

ABSTRACT

This thesis is concerned with the theory and practice of processing real time-varying imagery at both local and global levels. Novel algorithms are developed, analysed and computationally deployed for all stages of processing - from obtaining local motion contraints to segmentation and 3D interpretation.

Chapter 1 deals with local motion constraint, culminating in the presentation of an algorithm which derives two, differentially weighted, components of image motion at all points in the image. This is applied to a pair of real images to create a database for evaluating analytic, smoothing and segmentation algorithms in later chapters.

Chapter 2 is addressed to the theory of 3D interpretation of velocity images under perspective projection. Problems of degeneracy, ambiguity, narrow viewing angles and non-rigid motion are explored as is the aperture problem. New algorithms are presented for the "small object" case and for non-rigid motion.

Chapter 3 is concerned with applying the methods discussed in the previous chapter to realistic noisy images. A number of theoretical issues, to do with error minimisation and regularisation, are clarified. A range of algorithms are tested and compared on real and synthetic data.

Chapter 4 addresses locally-based ("smoothing") approaches to optic flow processing. A generalisation of Horn/Schunck and Hildreth smoothing algorithms to the wide-angle perspective case is theoretically derived, and its efficacy computationally demonstrated. A new algorithm, which combines a global assumption of rigidity with extremisation of surface smoothness, is described and tested. The relationship between local and global processes is discussed from a novel angle.

Chapter 5 discusses the segmentation and high-order variation problems and presents a new technique, related to "natural parametrisation", for addressing these. Its relationship to existing techniques of segmentation, integration of local and global processes, edge-detection and shape description is examined.

CONTENTS

LOCAL AND GLOBAL INTERPRETATION OF MOVING IMAGES

INTRODUCTION

This dissertation takes the form of an extended commentary upon the processing of a pair of images to elicit information on motion and structure. In the course of examining the problem I move some distance in the direction of a computational theory of visual motion in which the customary distinction between "local" and "global" processes is eroded. This, it might be said, is the deep theme; though the reader may frequently find it is obliterated by the crunching of numerals and the crackle of algebraic manipulation.

A state-of-the-art motion algorithm with which I am acquainted (Ibison, Zapalowski and Harris 1986) has the following structure:

1) Each of a pair of images is processed to identify and richly characterise small regions of high "interest" (such as corners).

2) A 1-to-1 match is sought, across the images, between these high interest points.

3) The resulting mapping, from one image to the other, is then interpreted in terms of a model of ego-motion in a rigid environment.

4) On the basis of the estimated camera motion the two images are brought into rough "register" (one or both are tranformed so that image-space shifts between matching points are lessened).

5) One-component motion constraints are computed at each pixel after the style of Horn and Schunck.

6) Segmentation into regions is effected by processing of the static image

7) Planes (in 3-space) are fitted to each region on the basis of the global motion computed at step 3 and and local edge-motion data computed for each pixel at step 5.

This algorithm possesses many sophisticated features which I have not mentioned. It is under development and promises to be highly effective in the task for which it is designed (passive navigation by an autonomous vehicle). But it could not possibly be taken as a paradigm for human vision or for any other general-purpose vision system. Note the clean distinctions made between point- and edge-motion, the use of one for ranging and the other for calibration, the purely structural nature of segmentation and purely dynamic nature of motion and depth estimation, the combination of a *global* assumption of rigidity with a *local*

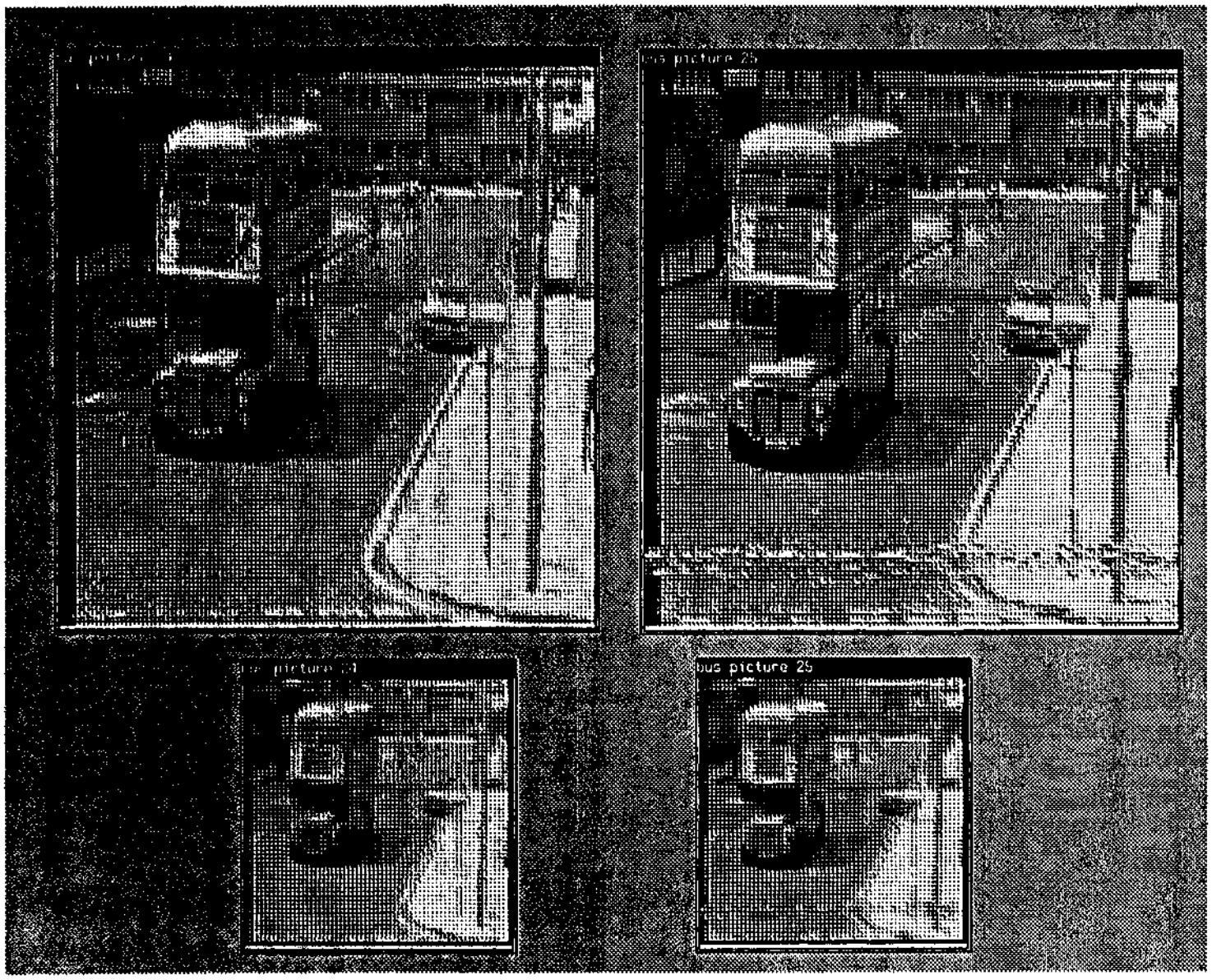


Figure Intro:1 London Bus grey-level image pair: 128x128 (top) and reduced to 64x64 (bottom)

smoothness constraint. This particular configuration of assumptions aad processes can be used only rarely on the real world. In human vision the following states of affairs would be commonplace, (All page references arc to the 1979 edition of Oilman's ""The Interpretation of Visual Motion*).

1) The image *mmmstM* only of curved line* or edges but 3-D motion is *perceived* (p 181).

2) Obj«t» in tlie environment are in relative motion or arc undergoing diitortion.

3) Image motion itself if the only cue to fegateatation (p!34» p 22).

4) Static inforattion is available to determine tie 3-D conformation of tbe scene* In such i situation structural wet may actually determine the motion interpretation - or even override the dynamic evidence fp 184). Set also Ames (1951), Scott and Btuctcm C19S5)

5) Flanarity way apply globally wMle rigidity applies locally or not at all - is In UUmaii'i conveyer belt desnontttiatkm Cp 138).

A number of strategies have been pursued in the attempt 10 Inure computer *vision* programs against *being* totally Cummcxed by an uneontrived glimpse of reality. These include:

- ratnetien of constraint* *rash* as rigidity or planarity to i local Eilt

- the use cf scnsiraints such as ngklly cr ssrsoUiness as extrea&um principles rather than *m* iffis'uufek Uws to which the ir.terfretaticn *must* cenferm

- expanu^Dn cf the *world mode]* to encempass phenomeii& u-hich show higher cr^ern of regularity

S^ih Iin*s cf exploraiAoa may -tivt iscvtd tetsi tbe froatiur cf lit possible a little, t^i the lasaits ^^ sort evident titan tit potential... Mi&& cf tlus ites is devoted to pr::t%n-g fsr lism lfSilf - tbouji I as careful 15 avoid prtautureJ/ itockrifij tiutt *ihty* fcavr beer* reached. OR tit contrary. I bavt put cciattdftable effort aito expaadEng Use ^*ffliicm:Liy* cf current approaches to:

- local motion constraint and local flow recovery.

- tit s^rfc^v angle cues

- vision as an "ill-posed problem"

- high-order interpretation of velocity images

- segmentation with special reference to "intrinsic images"

It is only in the final chapter that ideas ever-so-slightly angled to the major axis of current thought are tentatively aired.

The London bus data

Figures I.I and 1.2 show two successive grey-level images of a London bus approaching a road junction. This imagery was obtained in the course of road-traffic control project and used as data for a model-based vision program reported in Hogg et al (1984).

The two large (original) images are 128x128 pixels with 256 grey-levels. The smaller images are 64x64 reductions of these. The maximum image-motion of any part of the bus is approximately 3 pixels in the larger images. The images were obtained by freeze-framing a standard video film and various forms of corruption have resulted, notably the band of "video shatter" across the bottom of the second frame. There are pronounced specular effects on the roof and front of the bus. The camera data and true motion of the bus do not appear to have been recorded but most of it can be simply and accurately estimated on minimal assumptions. Since the bus is travelling along its length (we hope!) the focus of expansion is coincident with the vanishing point of the longditudinal edges of the bus. (This places it in the vicinity of the top right-hand corner of the image). Combining the known ratio of the length to breadth of a London bus with the image-ratio of these quantities and the location of the f.o.e. we obtain the focal length: the image is 0.55 ± 0.05 focal length units square. That there is no overall jitter, as opposed to local disturbance, is evidenced by nearly perfect in-register correlation at many "high interest" background localities.

The main parameter of interest which we are missing is the speed of the bus (or the product of speed and distance from the camera). The only available check on the accuracy of recovery of this is an estimate of image velocity of parts of the bus, made by hand over an extended sequence of images which includes the pair which we are examining.

Summary of Chapters

Chapter 1: Obtaining local estimates of motion

The starting point of most **computational** visual motion analysis is the estimation, across two or more images, of some constraint on local motion in the image. It is usual to distinguish "gradient'[1] and "correspondence'[1] schemes for doing this.

Gradient schemes are exemplified by Horn and Schunck's (1981) motion constraint **equation.** This gives an estimate of one of the two components of the flow-vector at any

pixel.

Horn and Schunck's assumptions have struck many as unduly restrictive. The assumption that movement is small in relation to the scale of spatial structure, necessary to justify local linear approximation of the image function, is not warranted in many circumstances. Where it does hold noise is liable to present a considerable problem. Some workers have accordingly developed other means, which focus upon edges, of computing theoretically equivalent constraints.

The assumption that the image-intensity at a projected point does not change as the point moves has been relaxed in work done by Cornelius and Kanade (1983). The precursor to the motion constraint equation is treated by them as a constraint *jointly* on the image motion *and* the rate of change of intensity. For recovery of full optic flow this necessitates placing restrictions on intensity changes as well as motions - but it is not clear how this is to be effected in a principled way.

As input to many algorithms for interpreting optic flow images one-component constraints are not sufficient. It is necessary to have estimates of both components of the flow-vector. Intuitively it would seem possible to obtain - at "distinguished points" such as corners - a rather reliable estimate of both components of flow rather than, as one gets with the Horn-Schunck procedure, one very unreliable component. There have been numerous algorithms developed to estimate point-motion. They tend to suffer from a characteristic defect: as a result of filtering out features which are liable to yield spurious "autocorrelation" matches the velocity image may become extremely sparse.

I present and discuss a method of obtaining motion constraint which yields pure point-motion or edge-motion estimates in limiting cases but which generally yields a hybrid type of motion estimate. A similar method has recently been described by Anandan and Weiss (1985), though their philosophy differs from my own in certain respects. The application my "principal axes procedure" to the bus imagery is described and the results displayed at the end of this chapter.

Chapter 2: Scene based interpretation of velocity images under perspective projection

The first stage in interpreting a "raw velocity image" involves either fitting field equations of some sort to all or a portion of the image, or recovering a full flow-field by iterative local operations. In either case the ultimate objective is to determine what is happening in the world, and between the observer and the world. Thus a sound preliminary step in visual motion theory is to study the optics and kinematics of flow-field generation.

The optics of multiple imaging under perspective projection have been well studied by many workers though treatments of the topic date back to Gibson (1950, 1957, 1979). In order to keep this document within a finite compass I concentrate upon the case where

displacement between camera positions is notionally infinitesimal. It makes very little difference, from a practical point of view, whether we use "velocity" models or "displacement" models in interpreting real image sequences. Over image motions of a few degrees of arc trigonometric refinements make no significant difference to results. Where image displacement is sufficiently large to make a difference, constraints on local motion cannot be obtained by the methods discussed in the last chapter. We have departed the territory of the putative "short range" process in human vision (Braddick 1974, 1980).

Most analysis to date treats rigid relative motion (i.e. the scene is not deforming, nor is one scene-object moving relative any other). Six parameters (a 3-vector to specify relative translation of the camera and a 3-vector to specify rotation) are required to describe any rigid motion. Two types of scene have been well studied - those composed of points disposed at arbitrary depths and those in which all visible points are arrayed on a plane.

Longuet-Higgins has derived equations expressing a relation between two-component image motion and image position, for both the general and planar cases. He has developed algorithms for recovering scene and motion parameters from a point-motion velocity image in both cases. I call these LH1 (the general case) and LH2 (the planar case).

Where we are given only an edge-motion field the planar algorithm can be adapted readily. A variant of it has been developed by Buxton et al (1984), but the algorithm for the general case cannot be straighforwardly adapted to cope with the "aperture problem". The assumption of rigid motion is not sufficient to allow determination of both motion parameters and the depths of scene-points. Some reference to other principles is therefore demanded.

This chapter is largely an attempt to establish the natural limits of the "global, analytic, closed-form" approach to optic flow interpretation. I explore problems of degenerate scene-and-motion configurations and of narrow viewing angles and I analyse the difficulties which are encountered in stretching the global approach to cover non-rigid motion and surfaces more complex than planes.

Chapter 3. Computational recovery of scene and motion parameters

In this chapter I compare, computationally, a number of algorithms for the estimation of global parameters of motion from local flow estimates. These include LH1 and LH2 as well as some iterative procedures which seek to directly fit a model of scene and motion to the data (without proceeding via flow-field parameters).

I preface the reporting of computational results with a discussion of error-estimation and of regularisation. With real, or realistic, data the solution is the result of an attempt to minimise a cost-function. In the currently fashionable paradigm of the Ill-posed Problem

such a functional should contain both an appropriate error term and "stabilisation" terms which represent generic constraints - broad presuppositions and prejudices not already incorporated in the interpretation model.

It is not a trivial matter to determine the minimum-error solution in the case of LH1 and LH2 since error terms become compounded with other terms in algebraic manipulation. Further, where regularisation is attempted, it can be difficult to penetrate their closed form to ensure that the stabilisation conditions are satisfied.

These practical difficulties, which are illustrated with the help of data from the bus imagery, point in the direction of algorithms with more exposed workings. I present and demonstrate what, in my opinion, are the two best iterative algorithms (for the general and planar cases respectively). Both these involve the explicit computation of point-depth. This facilitates the bringing-to-bear of certain types of constraint. Both show good convergence properties.

An advantage of an algorithm in which structural and dynamic parameters are kept distinct and iteratively estimated is that one set may be constrained to apply globally while the other varies between regions of the image. With noisy data there is a very marked difference between the accuracy of such a hybrid regional-global process and that of a closed algorithm applied on a purely regional basis.

Chapter 4: Locally-based interpretation of optic flow fields

In two well-known algorithms - due to Horn and Schunck and to Hildreth - optic flow recovery is sought through iterative, locally-based, computations which produce a smooth flow-field. I call these collectively the "MIT algorithms". A good prima facie case can be made for such locally based techniques. They lend themselves to fast parallel computation and seem to offer more opportunities for locating and respecting flow boundaries than do the global methods of analysis which are considered in the last chapter. In this chapter I am concerned to establish the natural limitations of *this* type of approach.

There are a number of obtrusive questions in relation to smoothing which I address in this chapter and the next:

1) Under what circumstances does smoothing yield a good approximation to the veridical velocity image?

2) What is the relationship between smoothing of the flow-field and smoothing of, say, the viewed surface?

3) How are we to relate locally-based smoothing to recovery of global parameters of motion and structure?

4) How are we to integrate segmentation processes with smoothing?

This chapter and the next attempt to confront these questions.

The story relating the analytic properties of velocity fields to the effects of smoothing-based recovery is complex and incomplete but it would appear that gradient-squared smoothing is a theoretically sound procedure where the true flow-field is Cat most) first order in image coordinates. Very crudely: the minimisation of grad-squared over a field tends to produce a situation in which the second differentials of the field are zero - and if the the true field really possesses this characteristic we are on reasonably firm ground. But a perspective flow-field, even in the case of a planar surface, is generally second order.

I show that we can treat the optic flow field in the perspective, planar, case as a field of 3-vectors (the "p-field") which is located in the image plane and is first-order in image coordinates. I argue that the correct way to extend the logic of the MIT algorithms to the perspective case is to recover a field of 3-vectors located in the image plane rather than a field of 2-vectors aligned with the image plane. Computational results on the bus data and on synthetic images suggest that my reasoning is correct.

Image motion, even my extended notion of it, is not in itself particularly "meaningful[11]. Can we extend smoothing approaches to recover properties such as depth or 3-D relative motion? It is quite straightforward to combine a global motion-estimation process with a local "relaxation" depth-estimation process to iteratively derive an interpretion of an edge-motion image and I present an algorithm which illustrates such global/local cooperation. But do we wish to commit ourselves to the a priori assumptions that the scene is globally rigid but only locally smooth? Perhaps we are viewing a forest which is locally rather rough but is smooth at a coarser scale.

A quality singularly lacking in computer vision programs is opportunism. The human visual system does not seem to apply "constraints" as if they were immutable laws. A rule which holds globally in one situation may apply only locally, or not at all, in another. *But,* for all its failings as a pedant, the human visual system is extremely quick to seize its chances for "regulaiisation[11]. It might be likened to an enthusiastic games^player; who now plays football, now tennis, now bridge, now a game they have just Invented and which lacks a name* If we hired a team of observers working in shifts to observe the player and record the laws by wMch they lived what a strange person we would believe ourselves to have discovered!

In the last section of this chapter I consider the relation between local and global processes *in* the light of the need for more *oppoxtunistic" switching between levels.

Chapter 5: A "natural parametrisation" approach to image organisation

I have allowed a cluster of problems to accumulate: the "what scale?" problem, the "which constraints are appropriate here and now?" problem, the "segmentation" problem and the "high-order variation" problem.

A general difficulty with the models which we have is that they tend to invoke simple functional relationships *between observables.* Both analytic and variational algorithms attempt to fit low order functions in which observables (e.g. velocity) depend on other observables (e.g. image coordinates), but the briefest consideration of the process of image formation suggests that such simple interdependency between observables is rare. A far more general model is that of a mapping, with "catastrophes", from "hidden variables" into observables including image coordinates.

Should we not therefore be trying to estimate the hidden variables - the "natural coordinates" - of points in our image? And should we not, further, be controlling the degree of mutual interaction between points on the basis of their proximity in the space of these natural coordinates?

(Perhaps I should say "more natural" coordinates - I do not wish to seem pedantic over a concept I cannot rigorously define. Let us also note here that there is no *a priori* reason for identifying natural coordinates with 3-D Cartesian coordinates. The parametric forms used to sweep out 3-D shapes in computer graphics define coordinate systems which are usually very non-Cartesian yet they are arguably quite natural).

But how can we escape from image space into the more natural spaces of the phenomena being observed? I approach this question via a consideration of the segmentation problem.

In a local-computation algorithm it seems natural to inhibit the mutual influence - the smoothing effect - between neighbouring pixels which do not belong together. To estimate the extent to which two pixels do belong together we must apply a metric which takes into account their separation in at least some salient dimensions (image space, velocity space, intensity space or whatever). This can be regarded as a possible estimate of their "natural separation". But we cannot merely place a "distance rod" between pairs of pixels to indicate the extent to which they are dissimilar and should be inhibited from communicating. This is liable to lead to an incoherent muddle in which pixels fail to fall naturally into groups and clear boundaries do not form. To prevent this I enforce a condition upon the "distance rods" placed between pixels: it should be possible to locate all pixels in a parameter space of two dimensions in such a way that the Euclidean distance between any pair, in that space, is the length of the distance rod between them. This provision, which is simply enforced computationally, has the effect of favouring large groupings and Marr's "continuity of discontinuities". Visually it can be seen as a rubber-sheet distortion of the image space which clusters groups of similar pixels together and attentuates the space between dissimilar groups. Or it can be seen in a complementary way as a distortion of a *parameter space* in the image space. Viewed in this representation the

process gives rise to "meshes" reminiscent of contour representations of surfaces (Stevens 1980; Weiss 1985).

Afterword

A very brief Afterword takes tentative stock of the achievements of this thesis and warns against the dangers of narrow-angle viewing on the part of Vision researchers.

A note on mathematics and program description

Workers in Computer Vision employ a very wide range of mathematical formalisms and, also, of conventions for describing algorithms. With regard to the maths I have tended throughout to to use "longhand" notation (as opposed to compact vector notation for example). I have also tried to give, in parallel with mathematical analysis, intuitively powerful verbal and visual metaphors. In these respects I have been guided by the desire to make this thesis comprehensible to as wide an audience as possible, particularly new graduate students at Sussex - even though this is not its primary business. Also it is important, I believe, to make it plain to ourselves and others that the issues addressed are not esoteric questions relating to a higher reality accessible only to a handful of initiates.

So far as algorithms are concerned I give both "high level" descriptions (e.g. I talk of "carrying out a least square regression" rather than mathematically describe the pseudo-inverse method) and a number of numerical examples which the reader can, if so inclined, work through with pencil and paper. The POP11 code is not given, but is available on request.

Some standard mathematical and computational texts are given in the Bibliography.

Acknowledgements

Vision *is* not a branch of Engineering or Pure Mathematics; it is the study of seeing, and seeing is about giving Hfe to the world and being receptive to the life it alone can give. My wife, an artist, has shown me this repeatly and kept me constantly aware of Wittgenstein's many dicta to the effect that: we intellectuals has better take care that we do not get bewitched, by the charm of our own inventions, into thinking that we understand what we scarcely know.

I owe a special debt to Christopher Longuet-Higgins and to Barnard Buxton, both of whom have given me many hours of their time. I have also profited from the criticisms and encouragement of David Hogg, Aaron Sloman, Chris Thornton, Mike Brady and many others*

The Cognitive Studies department at Sussex is not yet well-equipped as a vision laboratory. If Ben Rubinstein has not worked late into many nights there would be far fewer pictures, and a lot more numbers, in this document.

CHAPTER 1

OBTAINING LOCAL ESTIMATES OF MOTION

1.1 Introduction

Almost all current computational approaches to visual motion involve obtaining some form of *local* estimate of, or constraint on, motion at points in the image. The extremely wide variety of methods employed can be placed in three categories:

1) Those based on a model of the image as a smooth grey-level function without clear cut discontinuities or singularities. The Horn-Schunck motion constraint equation exemplifies this type.

2) Those which seek to determine the motion of edges - based on imagery which has been appropriately processed for edge detection or by simultaneous spatio-temporal processing.

3) Those which aim to determine full motion by such methods as cross-correlation of images or by tracking the motion of points which have been isolated and characterised by appropriate methods.

Methods of the first two types are customarily referred to as "gradient" schemes (Limb and Murphy 1975; Cafforio and Rocca 1976; Fukinuki et al 1976; Fennema and Thompson 1979; Horn and Schunck 1981; Marr and Hildreth 1980; Marr and Ullman 1981; Hildreth 1984). Methods of the third type are referred to as "correspondence" schemes (Ullman 1979; Aggarwal and Duda 1975; Chow and Aggarwal 1977; Barnard and Thompson 1980; Nagel 1982, 1983a, 1983b; Lawton 1983; Spacek 1986).

The last method generates a full estimate of point motion, whereas the first two yield "edge-motion" or "vernier motion" (one component only). Each approach is demonstrably superior to the other two in the context of appropriately selected or generated imagery! However, there are possibilities for unifying these approaches which I discuss and illustrate in this chapter.

It may be that the distinction between the two types of gradient approach is too finely drawn, since some schemes are difficult to place emphatically in one class or the other. But there is quite a marked philosophical, or perhaps temperamental, difference between those who think of an image exclusively in terms of edges and those who think in terms of a continuous image function. This will come to concern us in chapters 4 and 5, so it is as well to take note of the issue here.

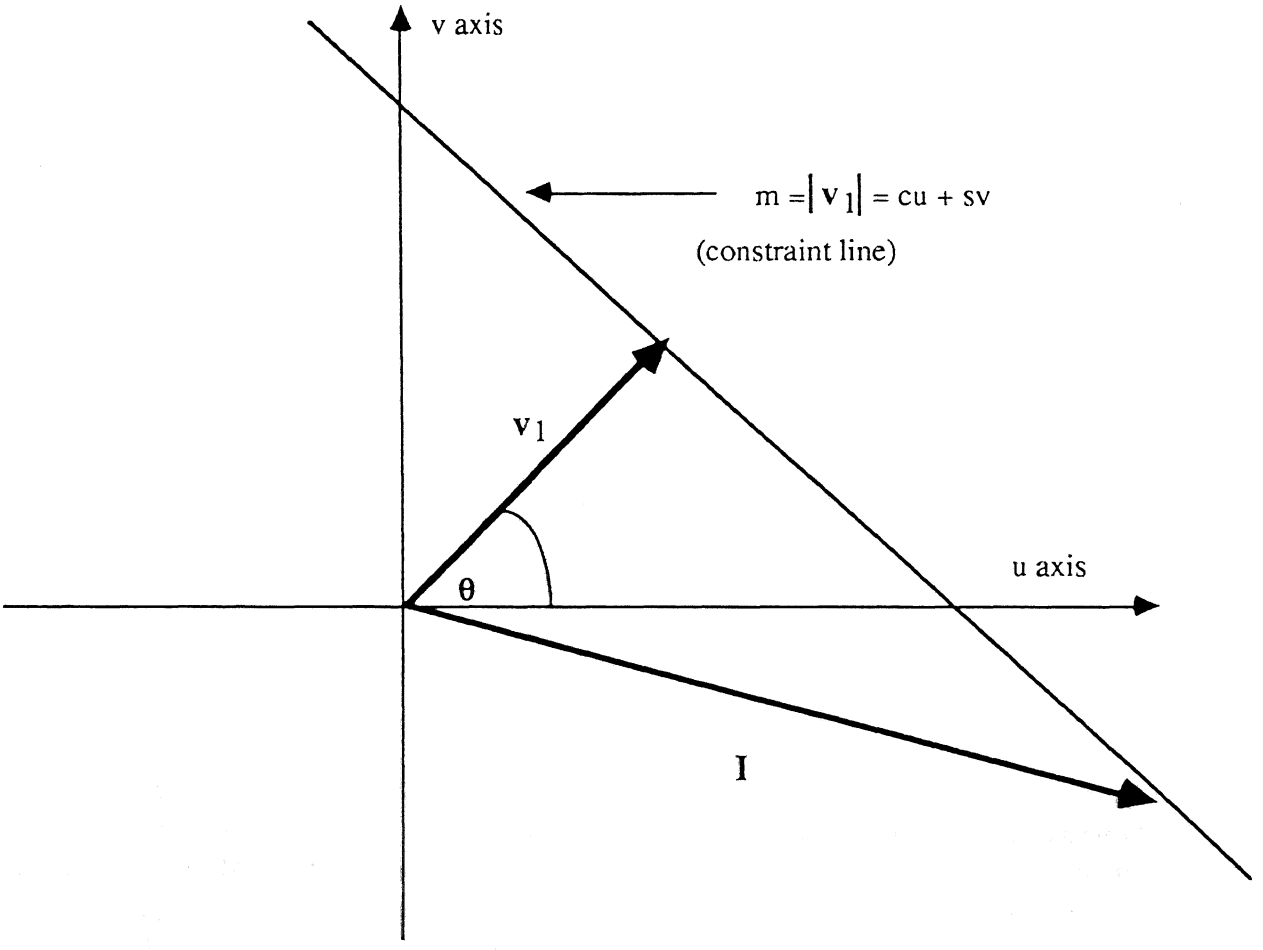


Figure 1.1 Relationship between edge-motion vector ($\mathbf{v}_1$) and point-motion vector ($\mathbf{I}$)

1.2 The motion constraint equation

In their paper "Determining optical flow" Horn and Schunck (1981) introduced a method of estimating a constraint on the possible motion of a point from one grey-level image to another. Their method assumes that the the motion-disparities involved are sufficiently small, relative to the scale of spatial structure for:

- a local linear approximation of the grey-level function to characterise accurately the "matching region" (in the second image) associated with any point in the first image.

If we call the image coordinates x and y and the grey-level intensity or irradiance E then any pixel in the first image can be represented as $\{x_1\ y_1\ E_1\}$. In the second image there will in general be a different intensity E2 at this location $\{x_1\ y_1\ E_2\}$. The "point" at x_1, y_1 in the first image will usually have moved to a different location in the second image.

If expected movement is sufficiently small we can take a region around x_1, y_1 in the *second* image and treat the grey-level function as linear in u,v (where u and v represent small displacements in x and y respectively from x_1, y_1):

$$E = E_2 + au + bv \tag{1.1}$$

If we assume - following Horn and Schunck - that the point does not change its intensity as it moves then we have, defining the locus of possible matches:

$$m = (E_1 - E_2) = au + bv \tag{1.2}$$

In the differential ("instantaneous") formalism $(E_1 - E_2) = -\delta E/\delta t$, $a = \delta E/\delta x$ and $b = \delta E/\delta y$ so that (1.2) becomes:

$$-\delta E/\delta t = u\delta E/\delta x + v\delta E/\delta y \tag{1.3}$$

Clearly u and v cannot be determined from (1.2) or (1.3) alone. What the equation does is to define a straight line in uv space upon which the image motion vector {u v} (I) must lie. It gives us, in other words, one component of the motion. No information about the orthogonal component is gathered under this procedure.

If we divide equation (1.2) through by $a^2 + b^2$ we obtain an "edge-motion" or "vernier motion" constraint in a geometrically more congenial form:

$$M = u\cos(\theta) + v\sin(\theta) \tag{1.4}$$

where M is the magnitude of the one component of motion defined by (1.2) and θ is the (anticlockwise) angle between the x-axis and the direction of that component

(θ = arctan(b/a)). Figure 1.1 gives a visual representation of the relations between the vector **I**, the "constraint line" defined by (1.4) and the edge-motion component v_1.

The quantity $a^2 + b^2$ is the square of the slope of the function (1.2) - in the differential case the gradient-squared $(\nabla E)^2$ - and it is a natural measure of the *strength* or reliability of the constraint. Where the grey level function is flat we expect little information on motion and $(\nabla E)^2$ will be small. In a region where the grey-level function is steep it will be large. $(\nabla E)^2$ exactly represents the "weight" which an equation of form (1.2) will carry in a least squares regression procedure (such as we will discuss at length later) aimed at recovering local or global properties of the optic flow field. It is therefore sound, simple and economical *computational* practice to leave the data in the form of (1.2) and not to convert it to form (1.4).

Horn and Schunck's work followed upon earlier work, largely inspired by attempts to model short-range motion detection in the human system, notably that of Marr and Ullman (1979). Schunck (1983) claims some support for the approach to be forthcoming from the psychological evidence, particularly with reference to the experimental work of Braddick (1974). The motion constraint equation had also been foreshadowed to some extent by workers in the area of image compression (for example Limb and Murphy 1975; Netravali and Robbins 1979).

1.3. Relaxing the constant intensity assumption

There are a number of criticisms that can be made of the method just outlined as a means of obtaining "the opinion of the data" upon the motion taking place between images - though its extreme elegance and simplicity serve to buffer it against these to a great extent.

Clearly the intensity of the projection of a physical point may change over time so the constant-intensity assumption is unwarranted in general.

If we do not make this assumption of constant intensity then from equation (1.1) we obtain:

$$(E_1 - E_2) - (E_1 - F) = au + bv \qquad (1.5)$$

where F is the new intensity of the point. The first LHS term $(E_1 - E_2)$ is the (known) change in intensity over frames at the location x_1, y_1 and the second LHS term represents the (unknown) change in intensity of the (moving) point from one image to the next. Call this latter term H. Then (1.5) is an equation which imposes a single constraint upon the vector {u v H}. It defines a plane in uvH space. I will call motion involving a change of intensity "grey-motion" (a bizarre bit of nomeclature - but we must have words).

Whether we can make good sense of constraints of the form (1.5) computed over an image depends on whether we can extend smoothing or analytical methods to cover intensity variation. It is by no means obvious how we would do this is in the general case. Even where the change in luminance at the projected position of a physical point is due to the changing orientation of a lambertian surface patch relative to the the light source, it is not straightforward to link a "shape-from-shading" analysis to the motion analysis. I am not aware of any algorithm which successfully integrates the two to determine grey-motion. While it is straightforward to extend smoothing approaches to encompass H in addition to u and v the *theoretical* justification for relaxation smoothing of the quantity H is less clear than it is with u and v.

Arguably the most interesting type of intensity change in an image is due to specular reflectance. In this case it is perhaps not even desirable that we discount the change at the earliest stage of processing. By actually tracking the "movement" of specular "features" we may stand to gain much information on intrinsic properties of scene surfaces. So it may be that an algorithm based on the constant luminosity assumption, together with an analysis of specular stereo such as that of Blake (1985), is more useful in a wide class of cases that one which attempts to elicit physical motion by discounting luminance effects.

It is perhaps significant that a recent algorithm which relaxes the constant intensity assumption (Cornelius and Kanade 1983) finds a natural application to X-ray imagery. In this context changes of intensity are generated by changes in the amount of material between the source and sensor (e.g. blood moving into and out of the heart) and have thus a more straightforward meaning than do intensity changes in optic imagery.

1.4 Varying the scale

Under most methods of estimation the linear approximation to the image function is more accurate in the immediate vicinity of the centre of the matching region than it is towards the periphery (typically the influence a point has on the estimate will fall off as a Gaussian from the centre). This suggests that it may be profitable, especially if the process of recovering the flow-field is iterative anyway, to update the motion constraint equation for each point on the basis of the region around the best-estimate-so-far of its shifted position. This might be simply done in one of two ways:

1) By adopting the constraint equation associated with the pixel in which the current estimated position of the point lies. (This will have been computed anyway - so the procedure is fairly cheap).

2) By approximating the intensity variation as a more complex function than a linear one - e.g. a cubic spline or a Fourier series - and deriving the linear constraint equation as the tangent plane to this function at the currently estimated position of the point.

1.6 The motion of edges

The above strictures on the limitations of using higher order qualities of the image do not necessarily apply in the context of *edge-tracking*, where no attempt is made to obtain a dense image field of motion constraints. Hildreth (1982), for example, makes use of the spatial and temporal gradient of the image after convolution with a difference-of-Gaussians to obtain a constraint on the motion of zero-crossing contours, in a manner logically similar to that of Horn and Schunck.

There is a widespread belief - based upon some evidence - that what counts in an image *is* edges. The linear approximation method of Horn and Schunck blurs sharp edges (approximating a step by a slope) thus effectively rendering the actual location of the edge less certain. If there are intensity changes between frames then matters are made far worse. Intuitively an edge is an edge - it has a definite location and orientation and it remains itself under changes of intensity. The problem of obtaining local constraints on motion might therefore be addressed by convolving successive images to locate either continuous zero-crossings (Marr and Hildreth 1980) or edges derived from a directionally selective operator (e.g. Canny 1983) and then tracking them from one image to the next. Marr and Ullman (1981) proposed that the human system effectively does the former by means of a filtering process involving both spatial and temporal derivatives. A computational development of this idea is the spatio-temporal operator developed by Buxton and Buxton (1983). This is designed to operate across a "stack" of images to locate and determine the characteristics of the spatio-temporal surface swept out by an edge over time. Marr and Poggio (1979) proposed a stereo scheme with coarse-to-fine matching of zero-crossing contours which Grimson (1981) implemented. Waxman's (1984b) ideas concerning "contour evolution" are natural extensions of these concepts.

Edge-tracking clearly has advantages in localities where edges are clear-cut and where the displacement is sufficiently small for there to be no ambiguity about how the edge-contour evolves in time. But where the alleged edges are zero-crossing contours tracing a locus of weak inflection points in a (necessarily somewhat arbitrarily) smoothed grey-level image the method is suspect - since there is no strong probability of invariance of "edgeness" across time in this case. Such unstable points may be removed by response-strength or scale-space filtering for example but, with some images, this would leave us looking a bit empty-handed.

There is also the point (with one eye on how nature does these things) that visual motion is a very powerful breaker of camoflage - it can act to destroy what appear to be *edges* in a static view. In fairness, however, this observation can be used to criticise the present state of any approach to obtaining local motion constraint.

1.7. Point-tracking

Horn and Schunck's model of a visual image is one in which intensity varies in a smooth, wave-like way in all directions. Indeed the best way to show off the power of their motion constraint equation is to generate a synthetic image function which varies sinusoidally in both the x and y dimensions (see for example Negahjdaripour and Horn 1985b). However edges have a sort *mf* Jtybrid nature - they are smooth and continuous in one direction and "singular[11] in the orHmgonal direction. But images may also be thought of in terms of points; discrete features (which might have properties other than location) disposed across the image plane like stars or, since we are in the motion business, planets.

In "sparse random dot" movies such as Ullman's rotating cylinder (Ullman 1979 p 134) each frame is composed of small distinct spots of light. It is clear that motion fusion is readily established in the human system across a pair of such images involving disparities considerably larger than the average distance between spots *within* a single frame. (In the particular case of the rotating transparent cylinder the flow-field is two-valued and also highly non-uniform - *so* there are no simple tricks, like conventional correlation, which will dispose of the problem).

In the case of sufficiently dense random dot movies or stereo pairs it may be possible to achieve something by blurring the image and applying Horn/Schunck type methods or extracting zero-crossing contours and applying edge-tracking techniques. But attempts to treat sparse random dot images in either way seem doomed. In the region of a singular pixel both the maflon constraint equation and the zero-crossing contour generated by the point are assfely sensitive to the smoothing scale and thus tend towards "meaninglessness".

Furthermore, even with *dense* images it seems intuitively that there are many singular or "distinguished" points which - provided they can be isolated - could *he* tracked from *one* frame to the next to provide a strong (two-component) constraint on local motion. It is precisely in the region of such features - comers and T-junctions for example - that the methods we have been diK»^tag perform most badly. The motion constraint equation tends to fail because the assumption of linearity is unwarranted and edge-detection schemes (based as they are on the assumption that there is only one edge in a vicinity) tend to smooth out more aimpkx structure.

Early approaches to computational koage matching and the theory of the human system tend to lean heavily on grey-liswcl cross-correlation (e.g, Reichardt 1961; Leese et

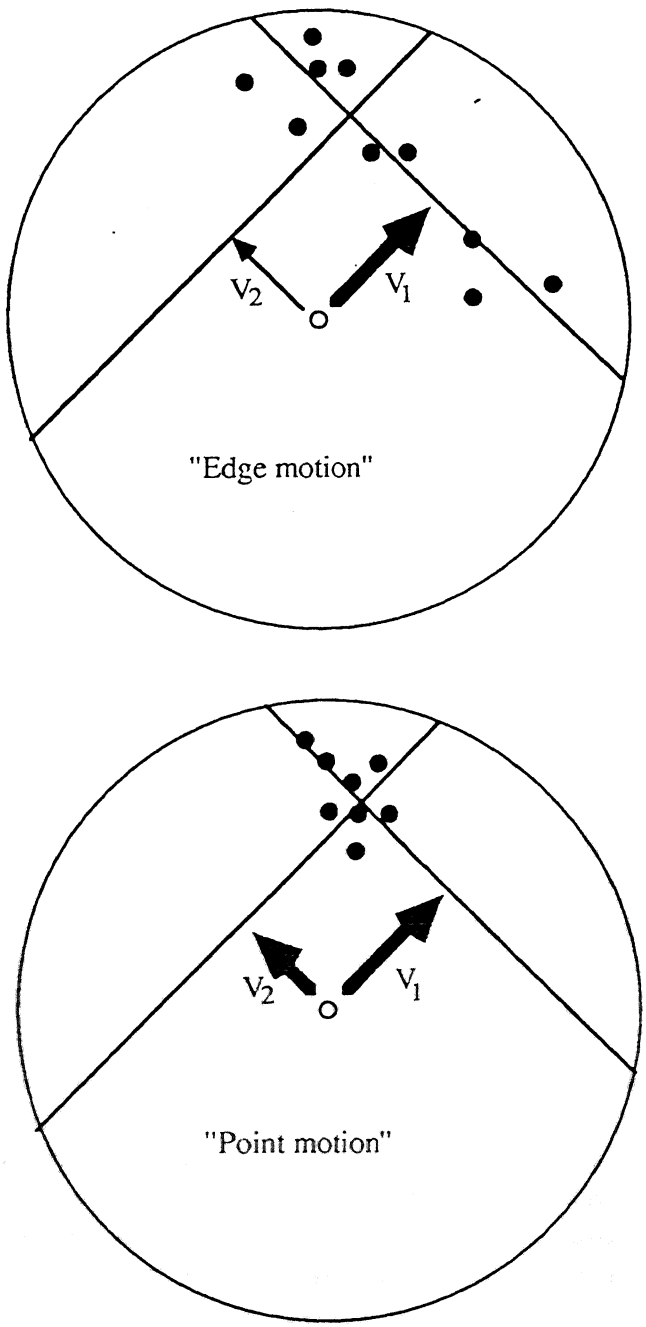


Figure 1.2 "Edge" and "point" motions obtained from principle axes procedure

al 1970). However this yields a unique match only where the image *autocorrelates* weakly when it is offset. Suppose a patch A of the image correlates well with a nearby patch B (as would be the case in a region of uniform grey-level or along an edge). Call the corresponding patches (in image 2) A' and B'. We can anticipate that A will cross-correlate well with B' (a false match) as well as A'. I call this effect, in later discussion, the "autocorrelation effect". All this suggests that points which are likely to yield a strong unique match, points which have "interest", could and should be identified before matching.

(There is a slight confusion of current terminology in relation to interest measures. Some people talk of autocorrelation "maxima" and others of "minima" to refer to the same property. I tend to prefer the latter since it is precisely the *lack* of correlation between a patch and its neighbours which defines interest.)

There has been an enormous amount of work done on the isolation and "rich description" of singular features and on their tracking (Potter 1975, 1977; Ullman 1979; Chow and Aggarwal 1977; Nagel 1982, 1983a, 1983b; Barnard and Thomson 1980; Medioni and Nevatia 1984; Shah and Jain 1984). Contemporary work in Britain includes that of Ibison and Zapalowski (1986) and that of Spacek (1986). The former follow in the tradition of selecting auto-correlation "peaks" or "minima" (by my convention) in each image as distinguished points. They use an operator based upon an analysis due to Forstner (1984). This yields quite a rich *characterisation* of each high-interest point, which facilitates matching, in addition to identifying it. Spacek's approach is to process images to obtain estimates of edge curvature - as well as location and orientation - which are as reliable as possible. High curvature points ("knots") along edges may then be selected for matching. This has similarities to the method described by Lawton (1983). In following this route Spacek is extending the use of curvature knots in static shape description to motion interpretation. Attneave (1954) originally drew attention to the salience of curvature points in human perception and Brady and Asada (1984) developed methods of automatically extracting these to form a "curvature primal sketch".

1.8 Unifying point and edge methods by obtaining two weighted components of motion

In this section I introduce an approach to the problem of obtaining local motion constraint which to some extent unifies Horn/Schunck, edge-tracking and point-tracking approaches. Some of the key concepts are present in (Anandan and Weiss 1985). Their treatment and its differences from my own are discussed.

Consider the problem of obtaining motion constraint on a straight line of finite length which is displaced slightly from one image to the next. An edge-tracking procedure, and the Horn/Schunck procedure in an equivalent grey-level image, should give us rather good estimates of edge-motion in the *central* region. But outside this region, in the vicinity of the end points, such techniques are liable to yield a good approximation to nonsense.

Conversely a point-tracking algorithm (given a suitable feature extractor) would accurately match the line-endings but give us highly indigestible lists of possible matches in the central region.

Suppose we are given a matching region containing a number of points each with the same matching strength and asked to say something computationally useful about the location of the correct match. There are a number of strategies for guessing this:

- pick a point at random
- pick the point closest to the centre
- pick the point closest to the "centre of gravity"
- pick the c. of g. itself

What we should be seeking to maximise is the confidence we have in our choice. On conventional statistical logic the right measure of confidence is the expected error in our choice. Unless we have some notions about the a priori probability distribution of disparity the correct choice is the last - the centre of gravity. There may of course be no point actually at the c of g. - in which case there is certainly a discrepancy between our estimate and the truth - but the *expected* discrepancy is minimised.

We can decompose our c of g. estimate of local motion into two orthogonal components by computing the principal axes - PM and Pm (major and minor) - of the group of points (figure 1.2). The two axes can be regarded as defining two orthogonal edge-motions with differential associated confidence. The first moment (the "spread") about either axis is a measure of the expected error in assuming that the motion lies on that axis. In other words the procedure provides us with two motion constraint equations - in one of which we have ''m^Yi-miim, and in the other minimum, confidence.

Where the points in the matching region are strongly Hneated (e.g. figure 1.2.a) we obtain one very reliable component of motion (defined by the major axis) and one very "fuzzy" component (defined by the minor axis). This tends towards the condition of pure edge motion though note that, since the region is bounded, the line is not infinite in extent and there is thus some confidence associated with the minor component. In recovering full motion this has the effect of preventing the estimated flow component parallel to the edge "rushing off to infinity*. It thus acts as a regularlser on the magnitude of local motion Gee Chapter 3).

Where there is only one point, or a closely spaced cluster of points as In figure 1.2.b, we obtain two reliable orthogonal components of motion viz. we effectively have full point motion. For point-patterns which do not lie at the extremes of the line case or the point cms© (or the null uniform distribution case) we merely have two "quite reliable[11] motion components V| and v_2 with associated moments or expected error e^ and e_2-

For a window containing discrete points we can substitute one with matching

strength varying from pixel to pixel. Anandan and Weiss, in their algorithm, compute such a field (they call it the "error surface") by correlating a patch of the grey-level image around the reference point in image 1, with the patch around each pixel in the matching region in image 2. I use their technique in processing the London Bus imagery and will describe and discuss it further below.

If we were to use similarity of pixel intensity as a basis for computing matching strength, in an area of a grey-level image where the intensity varies linearly, then application of the principal axes procedure is very nearly equivalent to the original Horn/Schunck procedure. There will only be pixels of high matching strength along the isophote of the same intensity as the reference pixel. This will result in a high-confidence motion constraint equation - the equation of the major axis - virtually identical to the Horn/Schunck equation. But if the reference pixel is "distinguished" (e.g. is bright in a sea of dark) then we will obtain two high-confidence motion components by my procedure - while the Horn/Schunck procedure will yield an inaccurate estimate of only one.

1.9 In defence of my version of the "principal axes procedure"

In Anandan and Weiss's algorithm it is assumed that there is a well-defined pit in the error surface. By "pit" is meant a strong clustering or lineation of pixels of high matching strength in the matching region which can be identified as "the best match". A procedure with affinities to my principal axes procedure is then applied by them *in the immediate vicinity* of the pit. It is only similar - not identical - since they recover the principal curvatures of the error surface in the pit region. The directions of these play the same role as my principal axes. They use the magnitudes of the two curvatures as estimates of reliability. (The use of the curvatures to define directions and as estimators of reliability seems somewhat counter-intuitive but appears to be the differential equivalent of my treatment of the problem).

Now this seems to me very difficult to justify theoretically. If the "error surface" does actually represent a probability distribution over the matching region then it is invalid to select the point or contiguous region of maximum probability density as the "best estimate" and even less valid to judge the reliability of the estimate on the basis of local behaviour of the probability function. (This implies a cost function which charges for small mistakes but not for big ones). This is not to say that such an initial choice would be wrong in a stochastic or other iterative algorithm in which the estimate may be revised in the course of computation. But where the aim is to summarise - in the form of weighted motion constraint equations - all the information present in the error surface, it seems that a maximum likelihood estimate should be made over the whole surface. If the error surface only shows a single pit - if there are very low matching strengths elsewhere - then it makes little difference whether the computation is performed locally or globally. But if the pit is not unique then Anandan and Weiss's method might yield motion components with high confidence while

mine will correctly reflect the high level of uncertainty associated with the motion of the point in question.

It might be said, against my version, that it is intrinsically conservative as a consequence of taking into account all spurious claims to affinity throughout the matching region. Since the window is symmetrical, and "spuriousness" tends to be arbitrarily disposed, it is indeed the case that underestimation of the magnitude of motion is more likely than overestimation by my method. But such a conservative bias cannot be avoided simply by opting for a single point or cluster of points of high matching strength. Statistically (over the image as a whole) it will still be present as a consequence of opting for the *wrong* point or cluster in some windows. The "inertia effect" comes with the problem - not the method.

And perhaps a touch of conservatism is not such a bad thing in local motion estimation. Schunck (1983 p59) had to introduce a "maximum velocity heuristic" in a smoothing process in order to prevent pathological behaviour on the part of the velocity field in the vicinity of boundaries.

My procedure does not eliminate the "autocorrelation effect". It will be seen that it may give rise to non-zero local estimates of flow in the absence of any movement (i.e. from an image to itself). But where autocorrelation components are present the confidence in at least one of the components of flow is appropriately reduced. We may still recover one component with great confidence - e.g. that orthogonal to a well-demarcated edge - whereas we would deprive ourselves of this with a pure point-matching algorithm.

1.10 Application of the principal axes procedure to the London Bus images

To obtain a first batch of data I applied the principal axes procedure with a 3×3 mask-size and a 5×5 matching window on the 64×64 imagery.

First, the "mismatch" between the 3×3 region around the reference point (in the first image) and the 3×3 region around each of the 25 points in the matching region (in the second image) is computed as:

$$MM = \sum_{i,j=-1,0,1} (E_{x+i,y+j} - E_{x+dx+i,y+dy+j})^2$$

$E_{i,j}$ is the intensity associated with pixel (i,j)

x, y are the pixel coordinates of the reference point

dx,dy are the "offsets" of the pixel for which matching strength is being computed

dx = -2,-1,0,1,2 dy = -2,-1,0,1,2 defines the matching region

i = -1,0,1 j = -1,0,1 defines the mask

The term "correlation" is used somewhat informally to describe this process. Anandan and Weiss (1984) argue the case for using this particular means of assessing matching strength. In essence the justification for summing the squares of grey-level differences, rather than absolute values, is that it yields a measure which is rather sensitive to changes in *relative* pixel brightness but fairly insensitive to uniform brightening or darkening.

The 3 x 3 region around the pixel (11, 20) in image 1 has the grey-level values:

y=			
19	10	23	161
20	9	24	166
21	9	31	171
	x= 10	11	12

(This region straddles a portion of the left vertical edge of the bus).

The 7×7 region around (11, 20) in image 2 has the values:

y=							
17	11	13	28	161	201	205	145
18	10	13	43	184	216	206	118
19	13	13	58	203	236	231	146
20	43	18	60	194	228	203	96
21	76	26	65	187	246	219	99
22	94	31	76	203	236	221	108
23	90	43	111	217	234	207	97
x=	8	9	10	11	12	13	14

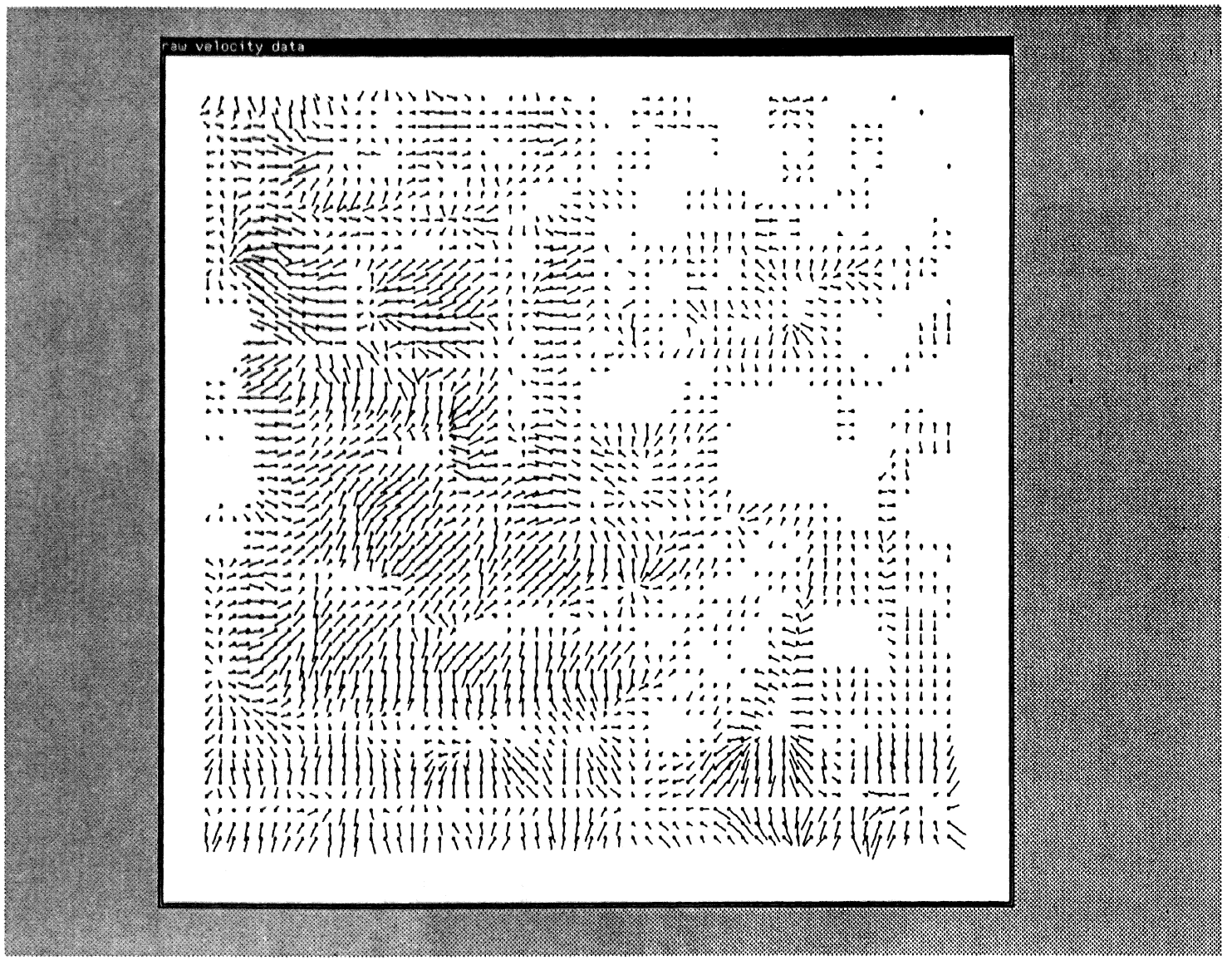


Figure 1:3 Raw 2-component velocity field from the 64x64 Bus pair (magnitude < 0.3 pixel unmarked)

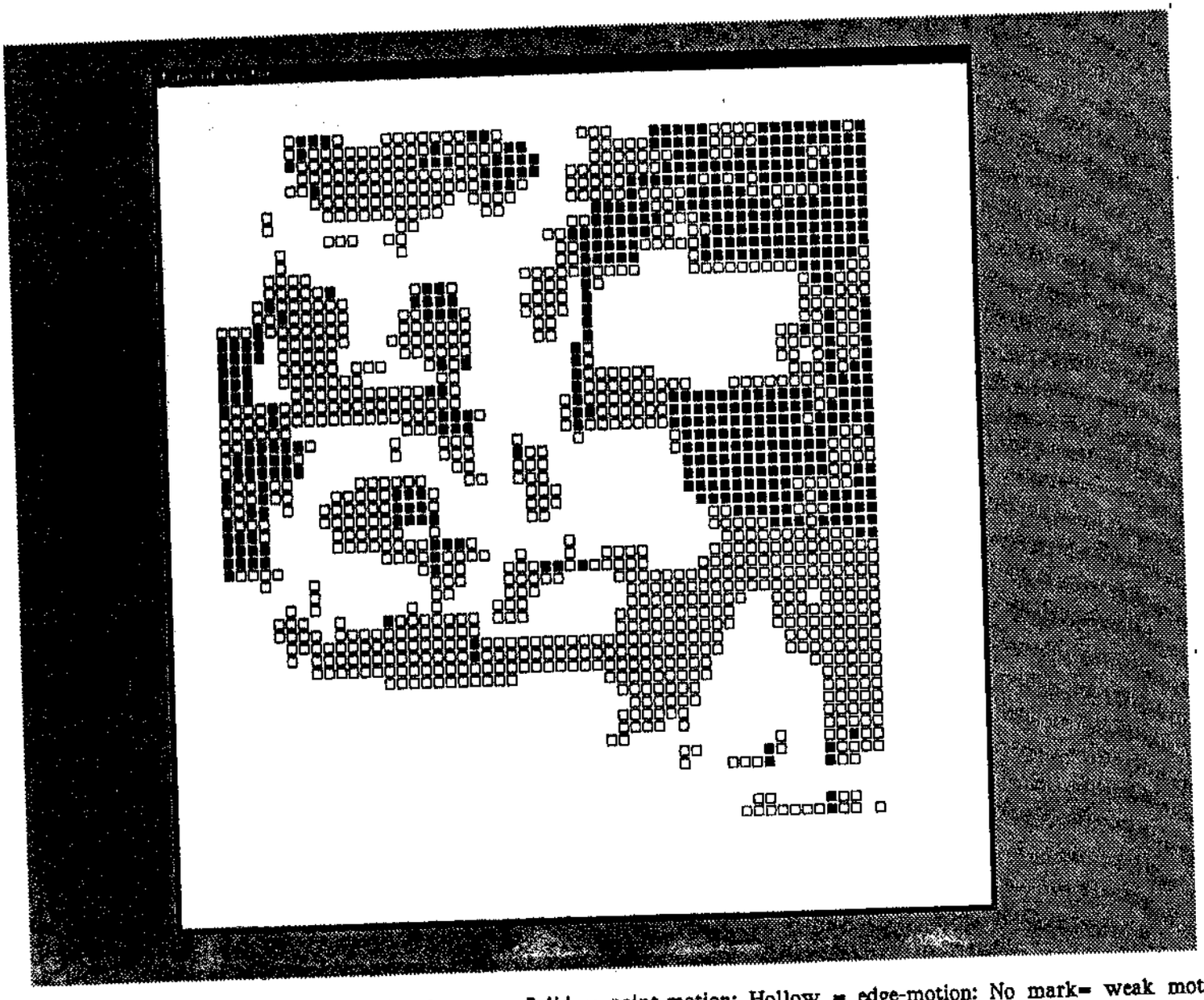


Figure 1:4 Local motion information by type. Solid = point-motion: Hollow = edge-motion: No mark= weak motion

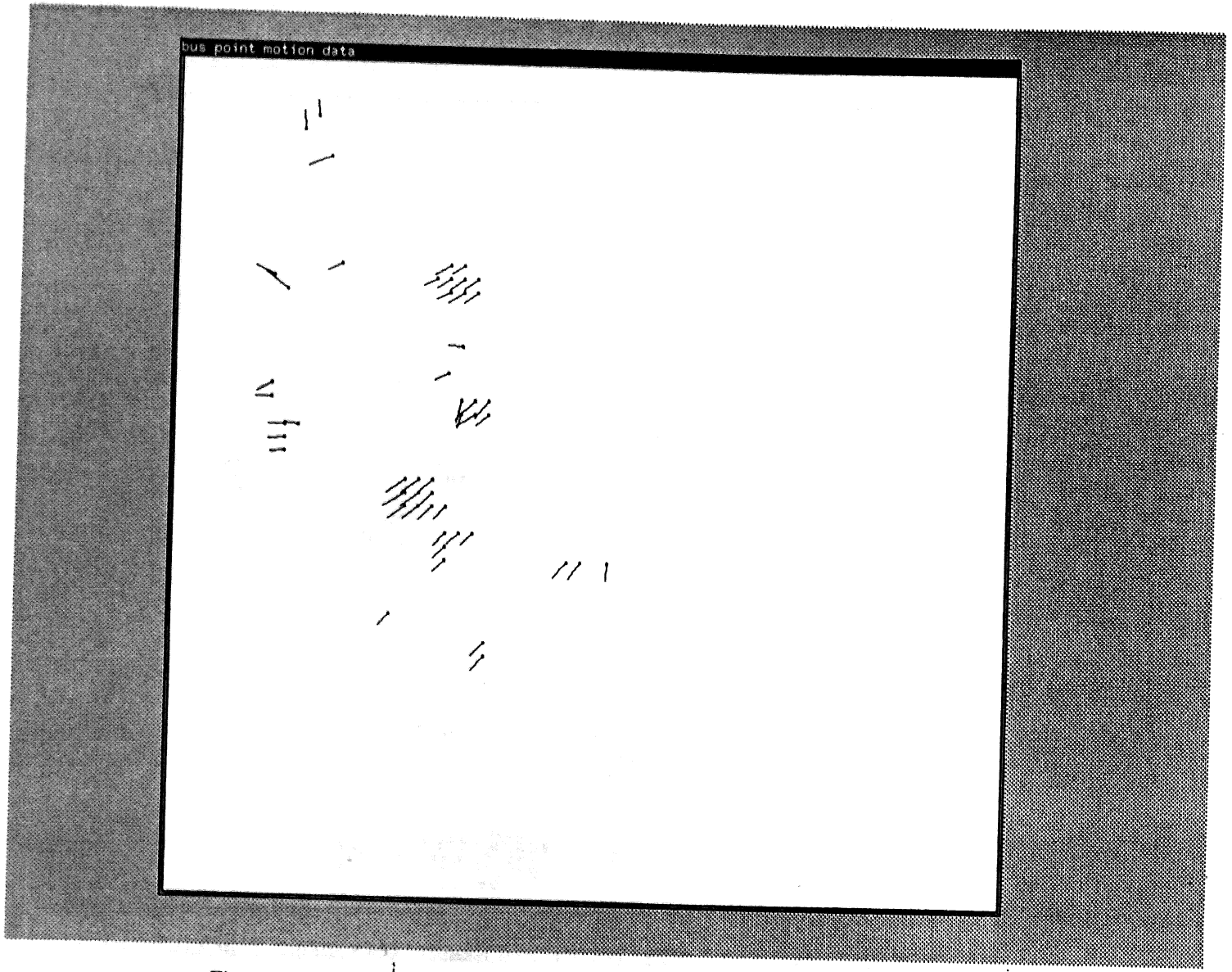


Figure 1:5 The 50 strongest point-motion vectors associated with the Bus

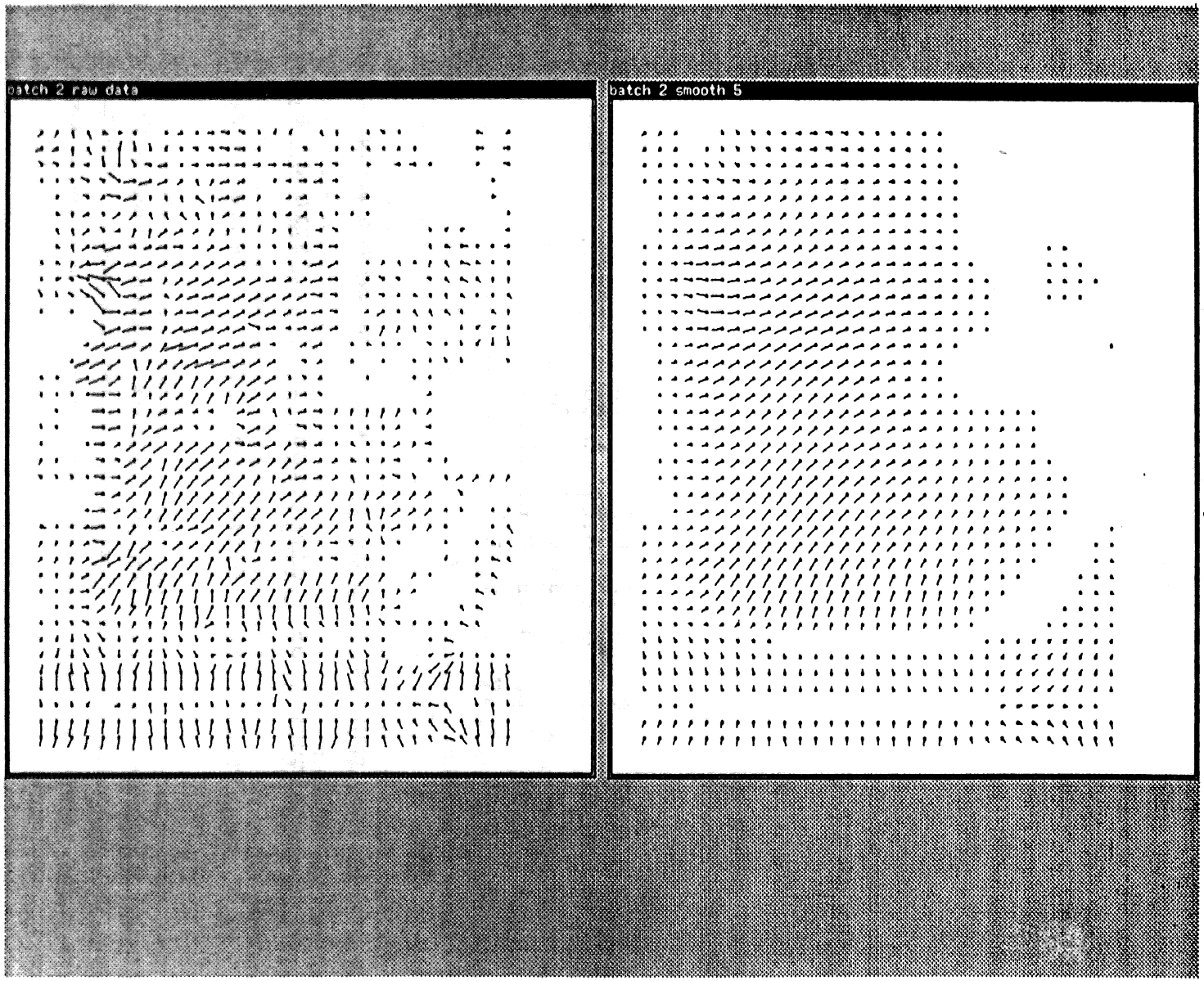


Figure 1:6 128x128 raw velocity field (see text)

Figure 1:7 After weak smoothing by Horn/Schunck

The values of MM computed for each of the 25 "offsets" are as follows:

dy-	y·					
-2	18	46150	2504	86434	208771	240292
-1	19	39471	4930	101796	230413	255722
0	20	38896	6860	108771	244941	276164
1	21	42259	8466	110633	243583	274430
2	**22**	39260	16925	127545	255796	278735

x-	9	10	11	12	13
dx	-2	-1	0	1	2

I have been unable to derive a principled formula for converting MM into a measure of matching strength. In the present case I employed the heuristic formula:

W - 20000/(100 + MM)

This yields the following "matching strength surface" in the current case:

dy-					
-2	0.43	7.68	0.23	0.1	**0.08**
-1	0.51	3.98	**0.2**	**0.09**	**0.08**
0	0.51	2.87	0.18	**0.08**	**0.07**
1	0.47	2.33	0.18	**0.08**	0.07
2	0.51	1.17	0.16	0.08	0.07

dx-	**~2**	**-1**	**0**	**1**	**2**

The principal axes procedure yields, for this configuration, the two motion components:

v_1 : magnitude 0.98 direction 180° **"spread"** e_1 0.38

v_2 : magnitude 0.66 direction 270° **"spread**[11] e_2 1.81

Direction *ïn* measured clockwise from the positlve-x direction. Spread is the first moment about the appropriate axis. Hie alignment of the two components with the x- and y- *Mxm kx tbh* cample ·« largely Incidental. The edge of the bus *is* almost precisely vertical **ind** the arroi^wirftce *M* too coarsely sampled to give us fine directional acuity. Also, the program which I used has a slight tendency to "default* principal axes to align with the x- and *y-mxm* as a consequence of stabilisation against arithmetic errors which would otherwise **arise** with a rotttionally symmetric distribution of matching strength.

The true motion of pixel (11, 20) is about u- -1.1. v- 0.3 versus the estimate of u- -

0.98, v= -0.66 which we obtain by the principal axes procedure. The error in v is due to a spurious tendency for the edge to "autocorrelate" strongly as it is moved upwards along its length. This appears to be due to "specular motion" rather than quantisation error or noise since a number of close neighbours show the same tendency. In any case this instance of the autocorrelation effect - if that is a correct characterisation of it - will not trouble us unduly since its presence is faithfully reflected in the high spread associated with v_2. To all intents and purposes we have obtained only one constraint on motion for the pixel (11, 20).

To translate spread into a measure of confidence - a "weight" to be associated with the component - I use the formula:

$$WW = 1/(1 + 5 \times \text{spread})$$

The stabilisation of the divisor is necessary to offset the tendency for quantisation effects to lead to spuriously low estimates of spread in some cases.

The procedure just described was carried out for each image position (more than three pixels from the boundary). The results are displayed in various ways in figure 1.3. to 1.5

Figure 1.3 shows the estimated 2-component vector at each point in image 1. Motions of magnitude less than 0.3 of a pixel-width have not been marked. Although at first sight the data looks extremely noisy, recall that some vectors have very little weight associated with them. Some do aspire to the condition of reliable "point-motion" vectors, some merely to the condition of "edge-motion" vectors and some aspire to neither condition - having two unreliable components. Figure 1.4 gives some indication of the relative distribution of the three types. A solid square indicates a vector having both components with WW > 0.2 and a hollow square indicates a vector having only one component which satisfies this condition. Absence of either mark indicates that both components are weak. Figure 1.5 shows the best 50 "point motion" vectors from the region of the image occupied by the bus and its shadow. These constitutes the "bus point motion list" which I use as data to compare algorithms in chapter 3. There are a few "specular" rogues but most of the vectors which survived the filtering relate to the physical motion of the bus.

Unlike the Horn/Schunck method the principal axes procedure thrives on detail at a finer scale than the magnitude of motion. Local motion estimates become more accurate as resolution is increased and the "error surface" more precisely defined. Some of the data which I use in the last chapter is derived from carrying out the procedure at 128×128 resolution with a 5×5 correlation mask over a 9×9 matching region. Figure 1.5 shows the result of "lumping" the 128×128 data into 3×3 "big pixels" and computing the pseudo-intersection of the 18 weighted constraint equations contained in the big pixel. (A vertical strip at the right of the image was excluded to reduce the computational load). Figure 1.6 shows the results of smoothing this 5 times with Horn and Schunck's smoothing procedure (which will be described in chapter 4). The vectors associated with the bus are quite well

aligned with the focus of expansion and the apparent motion induced by the band of video noise is accurately reflected by the vectors associated with that. In short, the 132x132 data is fairly good and not nearly as bad as it would look in some representations.

For maximum. acuity correlation-based techniques can be applied at "sub-pixel[11] resolution. For example we could create 256x256 images from the 128x128 by interpolation of grey-level values and apply the principal axes procedure to these. Anandan and Weiss, and Ibison and Zapalowski, do this with their correlation and auto-correlation techniques respectively.

However, for our purposes the 64x64/3x3/5x5 data, rough and ready though it may be, is suitable. There is no reason for being kind to the algorithms for interpretation which will be tested in chapters 3 and 4. In view of recent evidence (Todd 1985) that humans can make good sense of moving images with a signal-to-noise ratio of 0.15 we should perhaps not demand (though it is always welcome in practical applications) great accuracy from local motion processes!

1.11. A note on alternatives to the motion constraint approach

The principal axes procedure is intermediate in its scope between "differential[11] methods of obtaining motion constraint, such as Horn and Schunck's, and methods aimed at solving the "correspondence problem proper" of which numerous examples exist in the area of stereo vision (Barnard and Fischler 1982; Pollard, Mayhew and Frisby 1985; Prazdny 1985). The difficulty which stands in the way of adapting stereo algorithms to motion fusion is that the latter is a far less constrained problem. The epipolar constraint which is commonly enforced in the context of stereo - on the basis that relative camera positions are known - must be relaxed in most interesting motion interpretation contexts. Also, since imaging is simultaneous in stereo there is no problem of the object deforming, as there is in time-varying imagery even (one might say *especially/)* over small disparities. The human eye can see a circle deform smoothly into a square and sometimes prefers to see figural disruption involving smaller amounts of movement to rigid motion involving larger motions. Relevant psychological evidence includes the observations of Kolers and Pomerantz (1971) and of Navon (1976). The best know example in the computer vision literature is probably Ullman's "broken wheel" (Ullman 1979 p. 22). To complicate matters the percept may depend critically upon the interstimulus interval. The simplest and best known demonstration is Temus* configuration (Temus 1967).

In the stereo case there is considerable empirical support for the notion of a "disparity gradient limit* which can be applied as a basis for neighbourhood support computations (Burt and Julesz 1980; Mayhew and Frisby 1982a)» In motion fusion it as yet no clear, empirical psychopliysical evidence for an equivalent principle. So there seem to be limited possibilities for importation from the stereo field* I suspect that the possibilities for trade in the other dtrtctlon are also limited. A recent attempt to apply ideas developed for motion to stereo (Etstmmi and Waxman 1985) seems to offer less computational advantage than the

disparity gradient limit.

Many workers find local motion estimation, by any method, to be a distastefully messy, exhausting and inaccurate business and there have been a number of treatments of "motion without correspondence" (Jain 1983; Kanatani 1985; Koenderinck and van Doorn 1976b; Waxman 1984b). But the practical value of these numerous and varied ideas remains to be demonstrated. In principle it is of course possible to take complex structural qualities - such as the analytic parameters of a curve or an integral around a contour - and deduce much about the deformation of the image space from the way in which these high-order descriptors change. But to do so we need a good, solid understanding of what the correct descriptors are *and* a practical way of obtaining them reliably from real, noisy images. As the state of theory stands I can see no possibility of using these notions in anger (on real and realistic images) or even in mild irritation (as we might say in respect of the processing of synthetic imagery). On the other hand there is little doubt that the human visual system is capable of effecting fusion and inferring shapes and motions over very large image-distances (Wertheimer 1912; Kolers 1972) and in extremely noisy conditions (Todd 1985). I am particularly interested in an analysis of beta motion due to Foster (1978) who treats the problem as that of interpolating, by variational means, a parametrised spatio-temporal surface between observed cross-sections. His treatment has very close formal similarities to Weiss' (1985) recent approach to shape-from-contour, in which parametrised (spatial) surfaces are interpolated between extremal or intersection contours. This hints at possibilities for an integrated theory of spatial and temporal structuring which could resolve the dilemma over "what comes before what".

CHAPTER 2

SCENE BASED INTERPRETATION OF VELOCITY IMAGES UNDER PERSPECTIVE PROJECTION

2.1 Introduction

A comprehensive review of the mathematical analysis of visual motion is beyond the scope of this thesis. Such an undertaking would involve very many references - much water having passed under the bridge since Nagel's (1978) review with its mere 150. It would also have to draw attention to the high frequency of duplicated effort which lurks beneath a bewildering variety of formalisms, coordinate systems and recombinations of ideas.

In this chapter I offer a selective treatment of the imaging process and of image-interpretation which takes as its point of departure some of the work of Longuet-Higgins, whose analysis of perspective projection forms the theoretical basis for much contemporary endeavour in the visual motion field - though there exist analyses which are closely related in some respects (for example Tsai and Huang 1981, 1982, 1984a, 1984b; Maybank 1985a).

Longuet-Higgins employs a planar camera-model and pinhole-centred coordinates. Some workers continue with the long tradition of spherical projection (Maybank 1984) - and with alternative means of articulating the geometry of the perspective, rigid motion case (Ibison and Zapalowski 1985a, 1985b; Kitahashi and Hiroyuki 1985).

In particular I will consider two of Longuet-Higgins' algorithms - which I call LH1 and LH2. These are addressed to the case of an object of arbitrary shape and a plane respectively and invoke *instantaneous* motion. They may both be regarded as "velocity versions" of models which permit finite displacement between the two viewpoints (Longuet-Higgins 1981, 1986a). Much analysis of moving imagery involves finite displacement - for example Ullman's (1979) treatment of rigid motion under orthographic projection in which three independent views are required to determine the structure of the object. The finite displacement model may be more appropriate in the case of stereo vision. Indeed LH1 has its origins in an analysis of the stereo problem (Longuet-Higgins 1982a, 1982b; Mayhew 1982; Mayhew and Longuet-Higgins 1982).

Historically there are many analyses aimed at understanding and using the "motion parallax effect". Helmholtz (1910) realised that, like stereo, it could be used in the determination of depth. Gibson (1950, 1957, 1979) made much of the presence, in the time-varying optic array, of information regarding both the scene and the observer's motion. His insights appear to have little obvious applicability in computer vision (and it would be surprising if they did have, since the deliberate direction of his enquiry was orthogonal to the

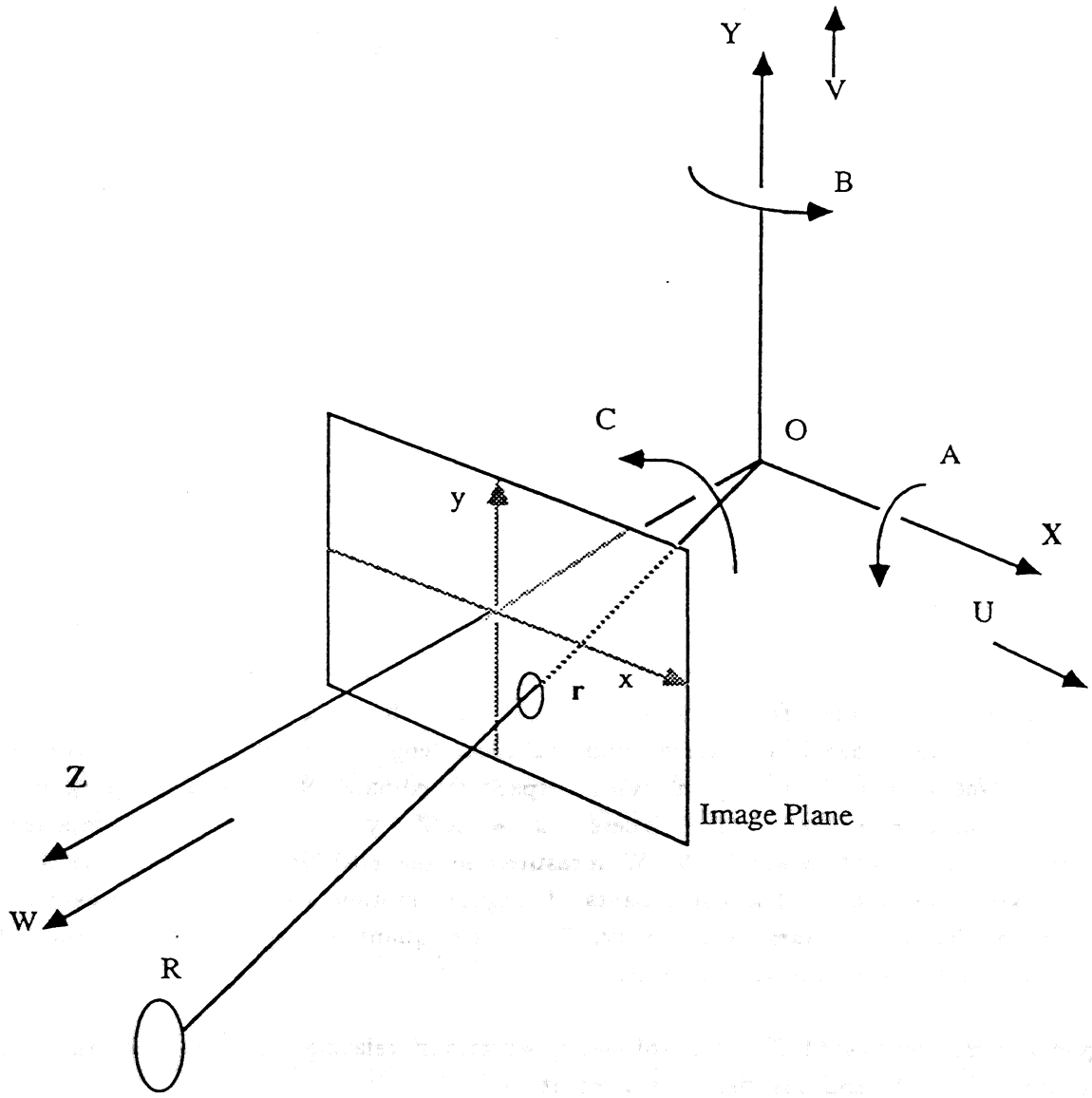


Figure 2.1: Coordinate system

(after Longuet-Higgins and Prazdny)

computational approach). An early mathematical treament for application to photogrammetry was that of Thompson (1959) but most analyses with direct computational implications date from the seventies (Lee 1974, Nakayama and Loomis 1974, Braunstein 1976, Koenderinck and van Doorn 1976, Ullman 1979, Longuet-Higgins and Prazdny 1980; Prazdny 1980).

In the early part of this chapter I describe LH1 and LH2 and an analytic solution to the aperture problem due to Buxton et al (1984). In the latter part I focus on the problem of modifying these algorithms to cope with a variety of realistic complications. These are: a narrow viewing angle, non-rigid motion, and objects which are smooth but more complex than planes.

2.2 Coordinate system

The instantaneous relative motion between a camera and a rigid object requires 6 parameters to specify it. Three of these relate to translational motion and three to angular motion. I follow quite closely the notational and other conventions of Longuet-Higgins and Prazdny (1981) so that (figure 2.1):

> A 3-D Cartesian coordinate system has its origin at the camera "pin-hole". The Z-axis runs directly away from the camera (so that the Z-coordinate of a point is its depth). The image is uninverted and the focal length is 1, so that the image plane is the plane Z = 1. Thus a point with 3-space coordinates R = {X Y Z} projects to the image-point r = {x y 1} where x = X/Z, y = Y/Z. The components of translational motion are U, V, W measured in the positive direction of the X, Y and Z axes respectively. The components of angular motion are A, B, C measured about the X, Y, and Z axes respectively. These six quantities specify the motion of the *camera* relative to the environment.

Longuet-Higgins derives (1981) the following equations relating: the image coordinates, the image motion vector, and the depth of a point:

$$u = (Wx - U)z - B(1 + x^2) + Cy + Axy \qquad (2.2.1a)$$

$$v = (Wy - V)z + A(1 + y^2) - Cx - Bxy \qquad (2.2.1b)$$

where lowercase "z" is the reciprocal of the depth Z. u and v are the x and y components respectively of the image motion, as in chapter 1. A novel derivation of these equations is given in chapter 4.

Immediately apparent in (2.2.1) is the so-called "depth-scale ambiguity" involving the depth Z and the translational motion vector {U V W}. If we multiply the velocity {U V W} by a scalar - and multiply the depth at each point by the same scalar - we leave the image motion unchanged. Intuitively: we cannot tell from kinematic considerations alone whether

we are moving slowly relative to a small, close object or fast relative to a large, distant one. Note however that the rotation vector does not interact in any way with depth; angular velocities not being affected by scale.

The depth-scale ambiguity encompasses what we might call the "back-to-front" ambiguity. We can multiply both depth and translational motion by a *negative* number without affecting the flow-field. Geometrically this exactly reverses the direction of translational motion and "mirrors" the scene about the plane Z « 0 - so that a point which was in front of the image plane moves behind it and *vice versa.* Of course a point which is behind the image plane cannot physically project to the image, however wide the viewing angle. But the algebra is. so to speak, unaware of this assymmetric fact of life. So long as the back-to-front ambiguity involved in an interpretation of an image is such that either *all* points lie in front of the camera, or they all lie behind it, there is no difficulty in disambiguating the situation by appeal to the "visibility constraint". But where the image is corrupted the interpretation may be such that scene-points lie partly in front of, partly behind the image plane *whichever* way we try to resolve the ambiguity. This problem receives attention in Chapter 3.

The back-to-front ambiguity can give rise to confusion when it is compounded with other types. Longuet-Higgins (1984) has established that in the case of a rigid moving plane there is four-fold ambiguity in interpretation whereas Maybank (1984, 1985b) has proven that there is a maximum of three interpretations in a class of cases which includes the rigid moving plane! Longuet-Higgins is including the back-to-front ambiguity and Maybank is not - so Longuet-Higgins means "two-fold" in Maybank's sense and Maybank means "six-fold" in Longuet-Higgins* sense.

2-3 The general case - LHI

In an algorithm. which I call LHI Longuet-Higgins eliminates the the unknown depth Z (or inverse depth z) from the two equations (2.2.1a) and (2.2.1b) to obtain:

$$x^2Q_6 - xyQ_5 + y^2Q_4 - xQ_3 - yQ_2 + Q_1 - v(U) + u(V) + (vx\text{-}uyXW) - 0 \qquad (2.3,1)$$

where % - (BV + CW)
Q_5 - (AV + BU)
Q4 - (AU + CW)
Q_3 - (AW + CU)
Q_2 - (BW + CV)
Q_t - (AU + BV)

The quantities x. y, u. v are observables at any data point. From eight general points therefore we can obtain eight equations of form (2.3.1) which can be solved to yield the *r€itios* of the nine quantities Q_6, Q_5,....W. Let us call these quantities, somewhat loosely, the "field parameters*. We cannot obtain the absolute values of all nine of them since

equation (2.3.1) is homogeneous. (This homogeneity is a reflection of the depth-scale ambiguity). The simplest way of obtaining a solution - other than the trivial solution in which all nine parameters are zero - is to fix the value of one parameter at, say, 1 and solve the resulting non-homogeneous equations by the usual method. If there are only eight points this will yield the same ratios regardless of which parameter is fixed (unless the true value of that parameter is zero in which case the resulting set of equations is degenerate). But if there are more than eight points and/or the data is corrupted by noise there is nothing straighforward about obtaining good solutions to (2.3.1)!

The appellation "field parameters" for Q^.....W is loose because equation (2.3.1) defines only one component of the motion vector at any point in the image. This is the component orthogonal to the translational component of that point's image motion. The other component can only be determined if a depth is associated with the point.

U, V and W are recovered directly (up to the depth-scale ambiguity) as field parameters. The angular motion parameters A, B, C may be straighforwardly recovered from these and any three of the remaining field parameters Q_6.....*Qy* There are three more field parameters than motion parameters - nine versus six (or eight versus five if the depth-scale is arbitrarily fixed). The field parameters are thus not independent. The ratios between six of them suffice to determine all that can be determined about the motion. The remaining three might be regarded as a check on the rigidity assumption.

If the data is perfect this interdependence of field parameters causes no difficulty. However if the data is corrupted by noise we have a problem. Eight points will always yield a "perfect* solution to the flow-field equation (2.3.1) but this solution may not be compatible with the assumption of rigid motion. We may recover one estimate of angular velocity from Q_6, Q_5 and Q4 and a totally different estimate from Q_5, Q_3 and Q_2 for example. Likewise a simple least-squaxes regression (LSR) performed on a set of more than eight equations of form (2.3.1) will treat the field parameters as independent - in general yielding "nonsense" values which are not in agreement concerning the values of the angular motion parameters. In both cases some iterative and/or exploratory procedure (typical of non-linear equation-solving) is indicated and such procedures cannot in general be guaranteed to find the best solution at an acceptable cost.

A further problem with LH1 is that it is degenerate in a non-trivial class of scene configurations - despite the fact that there may be sufficient information in the image to determine the motion and (thence) the scene. I return to this problem in section 2.6 below.

2.4 A note on ego- and world-motion

The coordinate system is chosen for analytical tractability and implies nothing about the coordinate system in which a particular "experience" is best described. If relative rotational motion is due to camera movement then our camera-centred coordinate system might be a natural one in which to specify this rotation. But if the camera is static and a distant object is spinning about an axis which it contains then this axis may be a more natural reference-axis in describing its motion. In such a case, describing the rotational motion with reference to camera-related axes involves invoking large amounts of rectilinear "pseudo-motion" to cancel out the swing of the object around the camera which the rotational parameters imply.

Restle (1979) describes a demonstration in which a circling dot changes its perceived sense of rotation - from clockwise to anticlockwise - as a consequence of a change in the motion of its environment. This striking illustration of the human eye's predilection for "natural coordinate systems" is discussed in Scott and Buxton (1985).

If a sensor is not translating (though it may rotate) in a fixed environment through which a single rigid object is translating then the flow-field which arises will be interpretable by LH1 *provided* we regard z as a parameter ("pseudo-depth") which scales rectilinear motion in a way not simply determined by depth. This is the situation which obtains with our London Bus imagery. In such a case the points making up the environment will be seen at "pseudo-infinity" - since they do not give rise to any rectilinear image-motion component - whereas the points making up the moving object will be seen at the correct *relative* depths. If a number of objects are moving parallel to one another - but at differing speeds - then the flow-field will likewise be of the type required by LH1. But relative depths between objects will not be obtainable. Geometrically speaking: there is a single focus of expansion in all these cases.

Thus, quite apart from the various problems of ambiguity and ill-conditioning that we shall encounter below there are ambiguities in the "scene-semantics" (e.g. the physical meaning of the quantity z) which cannot be resolved without stepping outside the kinematic domain.

2.5 The planar case - LH2

Another approach adopted by Longuet-Higgins (1984) to the elimination of the non-observable z or Z is to introduce the assumption that the object is planar. In the coordinate system used the equation of a plane may be written: $LX + MY + NZ = 1$ where L, M and N may take on any real values. Since $x = X/Z$ and $y = Y/Z$ we may write:

$$Lx + My + N = 1/Z = z \qquad (2.5.1)$$

and hence convert equation (2.2.1a) to:

$$u = (Wx - U)(Lx + My + N) - B(1 + x^2) + Cy + Axy \qquad (2.5.2a)$$

which is an expression containing only observables and global parameters of position and motion.

Treating equation (2.2.1b) in the same way and multiplying out the bracketed terms we obtain the field equations:

$$u = - P_1 - P_3x - P_5y + P_7x^2 + P_8xy \qquad (2.5.3a)$$

$$v = - P_2 - P_4x - P_6y + P_7xy + P_8y^2 \qquad (2.5.3b)$$

where

$P_1 =$ UN + B
$P_3 =$ UL - WN
$P_5 =$ UM - C
$P_2 =$ VN - A
$P_4 =$ VL + C
$P_6 =$ VM - WN
$P_7 =$ WL - B
$P_8 =$ WM + A

If we have perfect information on four points we can solve a set of equations of form (2.5.3a) and (2.5.3b) for the eight field parameters P_1, P_2, P_3 etc. The underlying scene-and-motion parameters U, V, W, L, M, N, A, B, C can be derived from these (with the qualification imposed by the depth-scale ambiguity) by a procedure which involves solving a cubic. It turns out that there is a four-fold (or two-fold!) ambiguity which will be discussed in the next chapter. With noisy data we are in a much better position (theoretically) than we were in the non-planar case since the flow-field parameters are truly independent - their number faithfully reflecting the 8-parameter family of discriminable situations.

2.6 More on the degeneracy of LH1

LH1 breaks down, trivially, if there is no translational motion or if there are insufficient data (fewer than eight points). In these respects it is no different from any other structure-from-motion algorithm (though some require less data). However it also suffers from a more idiosyncratic form of degeneracy in which a set of equations of form (2.3.1) will not yield a solution even though full information regarding the motion and the 3-D configuration of the scene may be contained in the data.

A conceptually simple way of proving that a set of linear equations is "degenerate" is to show that there are two distinct solutions to them (in which case there will be an infinite number since any linear combination of the two solutions is also a solution).

It is easy to see that LH1 breaks down in the planar case. Equations (2.5.2a) and (2.5.2b) are blunt statements of the fact that the two velocity components u and v, individually, are linear combinations of x, y, x^2, y^2 and xy. So we have two possible solutions to (2.3.1):

	Q_6	Q_5	Q_4	Q_3	Q_2	Q_1	U	V	W
Solution 1:	P_7	P_8	0	$-P_3$	$-P_5$	P_1	0	1	0
Solution 2:	0	P_7	P_8	$-P_4$	$-P_6$	P_2	1	0	0

Clearly these solutions are not, under any circumstances, the same. (There is in fact multiple degeneracy in the planar case but a single infinitude is all we require for purposes of the present discussion). There have been a number of attempts to exhaustively establish the conditions both for degeneracy of the 8-point algorithm (in both its versions) and for degeneracy of the problem of making a rigid interpretation of the data in general. (See for example Tsai and Huang 1984b). A general result concerning degeneracy of LH1's parent "8-point algorithm", which assumes discrete displacement between pinholes, has been obtained by Longuet-Higgins (1984). It is: the algorithm is degenerate if and only if all scene points and *both* pinholes lie on a quadric.

This result unifies, for the discrete case, what previously appeared to be a variety of degeneracy conditions involving:

a) "intrinsically degenerate" scenes

b) particular scene and camera configurations

b) image configurations

c) an inadequate number of data points

All these conditions are subsumed in the one "quadric degeneracy" condition. In the case of

a planar surface a quadric (in the form of a pair of planes) can always be constructed by adding a second plane which passes through both viewpoints. Since it is possible to construct a quadric through any nine points a scene in which there are fewer than seven points must satisfy the quadric degeneracy condition. And so on. It is mathematically a very neat result.

Longuet-Higgins extrapolates to the velocity case to obtain the result that LH1 is degenerate if and only if:

a) All scene points and the pinhole lie on a quadric

and

b) the motion of the camera is tangent to the quadric.

I will not reproduce his original proof, or show an alternative one here but I do wish to offer a clarification of the tangency condition in the situation in which the pinhole is at a point at which the quadric *has* no unique tangent. The general equation for a quadric is:

$$a + bX + cY + eZ + fX^2 + gY^2 + hZ^2 + kXY + mXZ + nYZ = 0 \qquad (2.6.1)$$

The pinhole of the camera is at (0, 0, 0) and by hypothesis lies on the quadric so $a = 0$. Making this substitution and dividing through twice by Z we obtain:

$$(bx + cy + e)z + (fx^2 + gy^2 + h + kxy + mx + ny) = 0 \qquad (2.6.2)$$

The equation of the tangent plane to the quadric at the pinhole is:

$$bX + cY + eZ = 0 \qquad (2.6.3)$$

This defines a unique tangent plane unless it is the case that $b = c = e = 0$, in which case the tangent plane is undefined. If this *is* the case then (2.6.2) becomes:

$$fx^2 + gy^2 + h + kxy + mx + ny = 0 \qquad (2.6.4)$$

viz. all points project onto a conic in image space. Conversely *if* all points lie on a conic in the image we can deduce that the camera is at a "tangent-less" point on at least one quadric passing through all scene points and the pinhole (there may be more than one quadric through all points including the pinhole). (2.6.4) trivially implies that all points lie on the quadric:

$$fX^2 + gY^2 + hZ^2 + kXY + mXZ + nYZ = 0 \qquad (2.6.5)$$

which has no defined tangent at the origin. (2.6.5) defines either a cone with its apex at the origin or a pair of planes whose line of intersection passes through the origin.

If image-points lie on a conic we can deduce that we have degeneracy. We cannot have zero translatory motion so there must exist a solution in which it is not the case that U = V = W = 0. This is distinct from the solution to (2.3.2) suggested by (2.6.4) in which it is the case that U = V = W = 0:

Q_6	:	Q_5	:	Q_4	:	Q_3	:	Q_2	:	Q_1	:	U	:	V	:	W
f	:	-k	:	g	:	-m	:	-n	:	h	:	0	:	0	:	0

The "conic" condition makes no reference to the motion or the depths of the points. So motion can be in any direction. There cannot in general be another quadric, with a defined tangent, through all scene points and the origin since the depths of each scene point are arbitrary and will, together with the pinhole, generally determine a unique quadric.

So to Longuet-Higgins' degeneracy conditions for the velocity case we should add the clarification that, where the tangent is undefined, the motion may be in any direction.

2.7. Irreducible degeneracy and ambiguity

The degeneracy of LH1 does not necessarily imply that the problem of interpreting the velocity image is "intrinsically" degenerate. There may be enough information in the data to enable us to determine the *five* degrees of freedom in the motion, but we may not be able to do so by proceeding via the fiction of a flow field with *eight* degrees of freedom.

LH1 involves "linearising" non-linear equations by multiplying them out and gathering terms to obtain equations linear in new parameters (which are combinations of the original parameters). This is a type of procedure which is well known to give trouble if the data are not perfect. We can clarify the fundamentals with a simple example: that of the two line problem.

I am given a set of data points and I am told that they all lie on a pair of straight lines in 2-D. I am not told the parameters of the lines. So for each data point one or the other, or both, of these equations holds:

$$ax + by + 1 = 0 \qquad (2.7.1)$$

$$a'x + b'y + 1 = 0 \qquad (2.7.2)$$

where a, b, a', b' are not known.

There are a variety of iterative and exploratory procedures I might use to partition the data into two sets and, at the same time, determine the values of a, b, a', b'. But I am keen on a closed-form solutions, so I proceed by writing:

$$(ax + by + 1)(a'x + b'y + 1) = 0 \qquad (2.7.3)$$

which is an algebraic ORing of the two equations - for each point either one holds and/OR the other. (Whereas to AND them I would square them, add them, and equate to zero - a form we will encounter later).

I now multiply out (2.7.3) and gather terms:

$$\underset{R_1}{(aa')x^2} + \underset{R_2}{(bb')y^2} + \underset{R_3}{(ab'+ba')xy} + \underset{R_4}{(a+a')x} + \underset{R_5}{(b+b')y} + 1 = 0 \qquad (2.7.4)$$

I can now derive from the data a set of equations of form (2.7.4) which I solve for the *five* parameters R_1.....R_5 by a simple closed procedure. From these I derive values for the *four* parameters of the original model. If the data are perfect, if I am given at least five points, and if they really do lie on *two* distinct lines this is fine. But if these conditions are not fulfilled then I have problems:

1) If the data only lie on one line then a set of equations of form (2.7.4) is degenerate. I can find an infinite number of solutions (corresponding to the infinite number of second lines which I can hallucinate). The image is too simple for the model!

2) If the data are noisy then from five points (or by least-squares regression over more) I may obtain values of R1.....R5 which are not consistent with their being composed of four underlying parameters in the right combination. So far as any linear solution procedure is concerned R1.....R5 are independent and equation (2.7.4) defines a general *conic*. The "solution" to a two-line problem may, then, be an ellipse!

I could make a virtue of necessity and decide that what I really wanted in the first place was a general conic-finder but this is allowing the solution to dictate the problem - and would not be of practical interest in a world which consisted entirely of straight bits.

There is no sure-fire way of solving the two-line problem with noisy data, short of an exhaustive search over every single possibility. In the case of the more complex problem to which LH1 is addressed such an exhaustive search (which would have to be over a 5-dimensioned space) is out of the question. So linearisation of the essentially non-linear continues to hold its appeal.

A virtue of LH1 from the theoretical point of view is that it does define conditions under which the data itself cannot be degenerate or have more than one interpretation. Any solution to the *problem* is a solution to the LH1 flow-field equation, so when the latter yields a unique solution that is the *only* solution to the problem. This is important because it means we can confine our search for intrinsic degeneracy or ambiguity to "quadric" situations. This search does not appear to be complete but there are a number of firm

results:

1) In the case that the viewed surface is planar there are at most two solutions, provided that the image is not such that the flow-field equations (2.5.3) are degenerate. (This would be the case where, for example, there were four data points but three of them were colinear c.f. Longuet-Higgins 1984 p 167). Further, a flow-field cast by a rigid moving plane cannot be interpreted as being cast by any other rigid body (Maybank 1984, 1985b).

2) In the more general case of an analytic flow-field there are at most three solutions for the motion (Maybank 1985), subject to the same proviso concerning uniqueness of the field parameters as above. These cases involve special combinations of motion parameters and parameters of the quadric surface. As Maybank points out, the chief interest of such shape-and-motion configurations is that, in the presence of noise, there will be difficulties with unambiguous interpretation of the flow-field when the veridical situation merely *approximates* to an ambiguous one. Both the planar and Maybank-type ambiguities can be resolved if the flow-field is followed over time.

3) Five "arbitrary" points in combination with "arbitrary" motion determine a unique interpretation consistent with rigid motion.

2.8 The aperture problem in the two cases

The fact that we may know only one component of motion at a point creates what is traditionally known as the "aperture problem" (Wallach 1976; Fennema and Thompson 1979; Horn and Schunck 1981; Marr and Ullman 1981; Marr 1982; Adelson and Movshon 1982; Hildreth 1984). Much early work on the problem made the assumption that velocity is constant over (some relevant portion of) the image (Limb and Murphy 1975; Fennema and Thompson 1979; Thompson and Barnard 1981; Marr and Ullman 1981). However this assumption does not in general hold in the class of cases covered by LH1 and LH2.

That the full image-motion vector at any point should be decomposed into an "x" component and a "y" component is of course arbitrary. We could specify it just as well in terms of any vector basis e.g.

$$v_1 = uc + vs \qquad (2.8.1a)$$
$$v_2 = -us + vc \qquad (2.8.1b)$$

where c and s are the sine and cosine of the (anticlockwise) angle between the positive-x direction and the v_1 direction. We may thus take any pair of equations such as (2.2.1) or (2.5.3) which have the form:

$$u = F(x,y)$$
$$v = G(x,y)$$

and rotate the basis to produce two orthogonal linear combinations containing exactly the same information:

$$v_1 = uc + vs = cF(x,y) + sG(x,y) \quad (2.8.2a)$$
$$v_2 = -us + vc = -sF(x,y) + cG(x,y) \quad (2.8.2b)$$

In the case of edge motion we may select our basis so that:

$$v_1 \text{ (known)} = cF(x,y) + sG(x,y) \quad (2.8.3a)$$
$$v_2 \text{ (unknown)} = -sF(x,y) + cG(x,y) \quad (2.8.3b)$$

We have effectively only one equation - (2.8.3a) - relating to each point. From the discussion of the last chapter it will be seen that F(x,y) and G(x,y) may be substituted directly into the Horn-Schunck motion constraint equation to obtained a (scaled) version of (2.8.3a). From the principal axes procedures of myself or Anandan and Weiss we would obtain, by substitution, two equations of forms (2.8.3a) and (2.8.3b) - the first with maximum confidence associated and the second with minimum confidence.

Carrying out the transformation (2.8.2) in the planar case upon the flow equations (2.5.3a) and (2.5.3b) we obtain:

$$v_1 = cu + sv \quad (2.8.4)$$

$$= -P_1c - P_2s - P_3cx - P_4sx - P_5cy - P_6sy + P_7(cx^2 + sxy) + P_8(cxy + sy^2)$$

v_1, c, s, x and y are observables at a data point and so, from eight general points, we have sufficient information to solve for P_1.....P_8. We can then seek to calculate the scene parameters from these precisely as before.

This is similar to the "3-D solution to the aperture problem" first described by Buxton et al (1984).

Note that a (planar case) full vector field can be regarded merely as a special case of a vernier field - it is a field in which there "just happen" to be two orthogonal vernier vectors associated with each point. Any program which handles the general vernier case automatically handles the full-field case without modification.

If we effect the transformation (2.8) upon equations (2.1a) and (2.1b) we obtain the single valid equation:

$$v_x \text{ « } cu + sv \qquad (2.8.5)$$

$$- (W(xc+sy) - Uc - Vs)z + A(cxy + s(l + y^2)) - B(c(l + x^2) + sxy) + C(yc - xs)$$

But this contains the unknown z. We cannot solve a set of equations of form (2.8.5) simultaneously to determine both the global parameters of motion and the depth map - though, given one, we can determine the other. LH1 relies upon there being *two* components of image-motion, and only one depth, associated with each image point.

If two points are very close in image space we might hope that they differ little in inverse depth. From two closely spaced (but differently oriented) vernier vectors we could interpolate a point-motion vector with a confidence dependent upon the separation.

If we have two arbitrarily oriented vernier vectors "at a point" (though they will not necessarily be, literally, at a point if we are interpolating) then our estimate of the full vector is:

$$u \text{ «• } (ms^* - m^*s)/(cs^* - c''s) \qquad (2.8.6a)$$

$$v - (me^{\bullet} - me^{\bullet})/(cs^{a} - c^{\bullet}s) \qquad (2.8.6b)$$

where m, s and c are the magnitude and sine and cosine of orientation respectively of the first vector and m\ s* and c* the corresponding quantities for the second vector, {u v} is the intersection in velocity space of the constraint lines for the two edge-motion vectors. Expressions (2.8.6a) and (2.8.6b) are merely the solutions to the pair of constraint equations:

$$m \text{ -» } cu + sv$$

$$m^{\bullet} \text{ « } c''u + sV$$

If we wish to interpolate between more than two edge-motions then we may take the pseudo-intersection in velocity sptace viz. the least squares solution to the set of more than two constraint equations.

'Our confidence in a point-motion estimate obtained by such simple means is, first, dependent upon the angle between the two edge-motion vectors (under conditions of additive Gaussian noise). The quantity (cs* - s'c) reflects this confidence (being 1 when the two edge-motions are orthogonal and zero wlien they are parallel). Care must be taken to ensure that the estimate carries an appropriate amount of weight in any subsequent computations.

In addition, if we arc interpolating the full vector between two separated points, then we should further reduce our confidence in our estimate according to their distance apart. Of course, we ran the risk of there being a flow boundary between our data points and this *Is*

liable to generate nonsense. A Hough transform type of approach to finding clustered intersections in velocity space has been explored by Schunck (1983, 1984) - but we will come to the segmentation problem later.

"Interpolation[11] of full vectors is intrinsic to smoothing approaches to recovering the optic flow field. Thus a velocity field which has been smoothed is suitable input to LH1. Further analysis of smoothness assumptions, smoothing algorithms, and the interpretation of smoothed fields, is contained in Chapter 4.

2.9 Narrow viewing angles in the general case

Both LH1 and LH2 involve recovering components of the flow field which are second order in image coordinates. While this is a realistic proposition in the case of a wide-angle image (for which the rigidity assumption holds) or in the case of perfect data it is very unrealistic in the case of narrow-angle imagery which is even mildly corrupted by noise. The interpretation of narrow-angle imagery is a question of much interest because, in real situations, it is frequently the case that the rigidity or planarity assumptions may be reasonably applied within *regions* of the image but not over the image as a whole.

Taking the general/point-motion case (LH1) first: for small angles of view we may delete the first three terms from equation (2.3.2), since they are second order in image coordinates. We remain with:

$$- x(AW + CU) - y(BW + CV) + (AU + BV) - v(U) + u(V) + (vx - uyXW) - 0 \quad (2.9.1)$$

Since we originally had three more field parameters than motion parameters we can still recover the motion - up to the depth-scale ambiguity - from the ratios between the six parameters in (2.9.1) provided we can estimate them reliably. Can we do so?

This depends on whether u, v and (vx - uy) show sufficient variation, independent of each other and of x and y, for a set of linear equations of the form of (2.9.1) - with one parameter fixed - to be weE-conditioned. The conditions of the required independence are that inverse depth, z, varies sufficiently (in a manner uncorrelated with x and y) over the image and also that the effect of rectilinear motion upon image-motion is sufficiently large relative to angular motion. We can see this readily by removing the second order terms from equations (2-2*1):

$$XL - (Wx - Vh - B + Cy \quad (2.9.2a)$$

$$v - (Wy - Yh + A - Cx \quad (2.9.2b)$$

A1» (with terms that arc second order in image coordinates removed)

$$vx - uy - (Uy - Vx)z + Ax + By \quad (2.93)$$

If z does not vary then u, v and (vx - uy) are each linear in x and y and a set of equations of the form (2.9.1) becomes degenerate. Likewise if z varies but in a manner strongly correlated with x and y then we can predict degeneracy. Substituting Lx + My + N for z and removing second order terms we obtain from equation (2.9.2a):

$$u - (WN-UL)x - (C - UM)y - (UN + B) \quad (2.9.4a)$$

Similarly for equations (2.9.2b) and (2.9.3). Thus if the points lie on a plane a set of equations of the form (2.9.1) is degenerate.

So long as variation in xz and yz is significant we do not have degeneracy, but as variation in z diminishes to the point where it is comparable with variation in x and y so does the significance of these second-order terms and we lose the requisite independence of our variables. One way of seeing this is to replace the term z in equations (2.9.2a), (2.9.2b) and (2.9.3) by (1 + d), where "l" represents our choice of mean inverse depth and d is the difference of z from this mean inverse depth. I call d the "relief term[11]. If we assume that d is small then, with second order terms in xd and yd removed since d is small:

$$u - (Wx - U)(l + d) - B + Cy \quad - \quad Wx - (U + B) - Ud + Cy \quad (2.9.5a)$$

$$v - (Wy - V)(l + d) + A - Cx \quad - \quad Wy - (V - A) - Vd - Cx \quad (2.9.5b)$$

$$vx-uy - (Uy - Vx)(l + d) + Ax + By - \quad (U + B)y - (V - A)x \quad (2.9.6)$$

Inspection of equations (2.9.5a), (2.9.5b) and (2.9.6). reveals a satisfying way of showing that the *problem* has become degenerate and of characterising the information we have lost. Given *any* solution to the flow-field: U, V. W, A, B, C and a set D of values "d" for each point I can construct a new solution as follows:

1) halve the values of all relief terms d in D
2) double the magnitude of U and V
3) adjust A and B to maintain the values of (U + B) and (V - A)

Recall that we have already *resolved* the depth-scale ambiguity - by setting mean depth to 1, The "relief-scale[1] ambiguity concerns the extent of depth *variation, given* the overall depth-scale, and an unresolved inverse relation between it and the magnitude of the component of translation parallel to the image plane. Analogously with the depth-scale case the relief--scale ambiguity encompasses a mirror-reflection ambiguity in which the *signs* of the reEef terms and the direction of {U V} are inverted. In this "inside-out" ambiguity a configuration of points which is convex may be seen as concave, for example, and the component of translation parallel to the image plane reversed. A pair of inverse interpretations cannot be disambigu&ted on the basis of a "visibility constraint" in this case, however.

We could not distinguish in a "small object" situation - however we actually chose to solve

for scene and motion parameters - between the following types of situation:

a) We are travelling slowly and "deep" (U and V small in relation to W) and viewing a bust of the queen.

b) We are travelling fast and "shallow" (U and V large in relation to W) and viewing the queen's head as embossed on a coin (in bas-relief).

c) We are travelling "in reverse" relative to the first two situations and viewing the inside of a mask of the queen (the inverse of a) or an engraving of her (the inverse of c).

The existence of the one-parameter family of rectilinear motion directions in the narrow-angle case has been discovered by Maybank (1985b) in applying and analysing his perspective motion algorithm.

Let us note that variation of the term d independent of x and y is a requirement for the "small object" approximation. If the object is planar we lose still more information.

There are a number of ways of solving for such quantities as we can determine in the small object case. One way is simply to fix two of the parameters in equation (2.3.1) - for example V and W - and bear in mind that the ratio between them is arbitrary when interpreting the solution. However there are methods which promise to be more accurate and easier to stabilise computationally. I outline two related algorithms, SO1 and SO2. (These are implemented in the next chapter - see section 3.8).

SO1 works as follows:

1) From equations of form (2.9.6) we obtain values for (B + U) and (V - A). Two equations are the minimum required in theory but an LSR over all data points is advisable with noisy data.

2) Writing R for our estimate of - (B + U) and S for - (V - A), we resolve equations (2.9.5a) and (2.9.5b) to eliminate d. Gathering terms we obtain a "small object" version of equation of the LH1 flow-field equation:

$$xq_2 + yq_1 + (v - S)(U) - (u - R)(V) = 0 \qquad (2.9.7)$$

where $q_2 = (CU + WV)$ and $q_1 = (CV - WU)$

3) From estimates of the ratios $q_2 : q_1 : V : U$ derived from a set of equations of form (2.9.7) we have the direction of the vector {U V} and we may immediately derive W and C:

$$W - (Vq_2 - U_{qi})/(U^2 + V^2)$$

$$C - (Vq_x + Uq_2)/(U^2 + V^2)$$

In all discussion of narrow angle approximations it is implicitly assumed that our narrow viewing cone is in the region of the principal point x«0, y«=0. If the viewed region is distant from the principal point then it is necessary to effect a transformation of the image coordinates and velocities to simulate a "swing" of the camera to centrally position the image- This is a simple operation (particularly if the image is in angular - i.e. spherical retina - form from the start). Even if the image is roughly central we may gain some accuracy by centering it precisely.

SO2 works as follows:

1) The image is tranformed so that the coordinate system has its origin at the centroid of the data points.

2) Summing equations (2.9.5a) over all N points and dividing by N we obtain:

$$\bar{u} = W\bar{x} - (U + B) - U\bar{d} + C\bar{y}$$

where the horizontal bar indicates the average value. Due to the tranformation we have effected, and the fact that d is variation either side of mean depth, it is the case that:

$$\bar{x} = \bar{y} = \bar{d} = 0$$

so that

$$R « - (U + B) - ¥$$

in this coordinate system. Similarly by summing equations (2.9.5b) ova: all points we obtain:

$$S - - CV - A) - \bar{v}$$

3) This enables us to proceed with the solution of equations of form (2.9.7) as in SOI. We must remember that all estimated quantities are in terms of the tr&nsfoimed coordinate system and must be themselves retransformed if we want the information in terms of the original system.

Intuitively R and S are "fixation* terms - they define the *rate* at which we would have to swing the camera, during imaging, to keep the projection of the centroki stationary, (v - S) and (u - R) are thus "velocities after fixation* and are zero at the point of fixation itself. In both cases we fixate the principal point x •• 0, y •« 0 but in the SO2 case we have first

swung the camera to make this coincident with the centroid of the data points.

The minimum number of points required by SO1 or SO2 is 4. This is not immediately apparent but is more obvious if we consider a "4-point algorithm" due to Longuet-Higgins (1986b). This algorithm, which is not suitable for application to more than four points without modification, works by transforming the coordinate system, as in SO2. But one of the data points is explicitly chosen to be the "centroid". For this point, in the transformed system, $x = y = d = 0$, $u = R$, $v = S$. Clearly we cannot use this point again in solving equations of form (2.9.7) since all the relevant quantities x, y, d, (u - R) and (v - S) are zero. Three more general points are required for this.

A tempting application of a small object algorithm is to apply it, over narrow viewing angles, repeatedly or simultaneously at locations dispersed across the image. (Spherical projection is more helpful to the imagination here). The results might then be integrated - after all had been converted to the same coordinate system - to obtain a global intepretation of motion. The fixing of the depth-scale is arbitrary for each application of a small object algorithm and the problem of integration is less straighforward than it might seem at first sight. However, it is a promising line of enquiry, particularly where there are computational possibilities for much pre-processing at the local level. Ullman (1979 p 160-168) describes in outline a "polar-parallel" scheme which is similar to this save that it is the "orthographic approximation" which is applied at local level. It does not appear to have been implemented.

2.10 Narrow viewing angles in the planar case

In the planar, narrow-angle case we have no surplus of flow-field parameters over scene parameters and the loss of the second-order terms from equations (2.5.3) thus opens up a two-parameter ambiguity (in addition to the depth-scale ambiguity) in the scene and motion parameters. The situation may be interpreted thus: the relief-scale ambiguity is still with us *and* we have an uncertainty regarding the direction of {U V} which we did not have in the small-object case.

The "planar patch" problem has been thoroughly investigated by Koenderinck and van Doorn (1976a) and I here merely adapt results obtained by them in a differential formalism to the formalism and terminology we are using.

Equations (2.5.3) become:

$$u = -P_1 - P_3x - P_5y \qquad (2.10.1.a)$$
$$v = -P_2 - P_6y - P_4x \qquad (2.10.1.b)$$

where $P_1 = UN + B$
$P_2 = VN - A$
$P_3 = UL - WN$

$$P_4 = VL + C$$
$$P_5 = UM - C$$
$$P_6 = VM - WN$$

We define:

$$a = -(P_4 + P_5) = -(UM + VL)$$
and $$b = -(P_6 - P_3) = UL - VM$$

Call the angle between {U V} and, say, the x-axis θ and that between {L M} and the x-axis ϕ. Then:

$$\tan(\theta + \phi) = (\tan(\theta) + \tan(\phi))/(1 - \tan(\theta)\tan(\phi))$$

$$= (V/U + M/L)/(1 - (V/U)(M/L))$$

$$= (UM + VL)/(UL - VM) = -a/b \qquad (2.10.2)$$

viz. $(\theta + \phi)$ is invariant - fixed by the data - though θ and ϕ individually are not. The natural geometrical interpretation is that the *bisectors* of the two vectors {U V} and {L M} are fixed. In visualising the ambiguity regarding the *direction* of {U V} - and {L M} - we may imagine the directions of these two vectors as aligned with two halves of a pair of scissors which may open and close though the axis of symmetry is fixed. (The bisectors are in fact the directions of maximum and minimum "compression" of the image - a and b being the two components of deformation).

Regarding the *magnitudes* of {U V} and {L M} we can rapidly show that the product of the two is invariant:

$$(a^2 + b^2)^{1/2} = ((UM)^2 + (VM)^2 + (UL)^2 + (VL)^2)^{1/2}$$

$$= ((U^2 + V^2)(L^2 + M^2))^{1/2}$$

$$= |\{U\ V\}|\ |\{L\ M\}| \qquad (2.10.3)$$

So, choice of a direction for {U V} determines the direction of {L M} and choice of a magnitude for {U V} determines the magnitude of {L M}. These are our two ambiguities. The latter - the magnitude ambiguity - can be seen as the relief-scale ambiguity:

$$e = -(P_6 + P_3) = 2WN - (UL + VM) = 2WN - \{U\ V\}.\{L\ M\} \qquad (2.10.4)$$

("." indicates scalar product)

Setting N = 1 to resolve the depth-scale ambiguity we have:

$$W - (e + (UL + VM))/2$$

It is clear that W is invariant over choice of magnitude of {U V} since (2.10.2) and (2.10.3) together tell us that (UL + VM) is determined solely by the choice of the direction of {U V}. Given any choice of this direction, W is determined, but the relative magnitudes of {U V} and {L M} may vary. Thus the plane may tend to fronto-parallel and motion to "shallow" - l{L M}l small, I{U V}I large - or the plane tend to highly slanted and motion to "deep". This is a description of the relief-scale ambiguity in the planar case.

There are two relations of some interest which are invariant *given the* direction of {U V}.

$$c - (P_5 - P_4) - VL - UM - 2C \; - - \; \{U\ V\}x\{L\ M\} - 2C$$

("x" here indicates the cross-product of two vectors)

The scalar product of {U V W} and {L M N} gives us the inverse of "time to impact":

$$\{U\ V\ W\}.\{L\ M\ N\} \ll UL + VM + WN \ll -e + 3WN$$

So C, the component of rotation about the optic axis, and the "time to impact" are fixed once the direction of {U V} has been determined. If the camera really *is* on its way to impact with the visible planar patch (rather than an extrapolation of it) then U, V « W and the inverse of time to impact is approximately WN or e/2 (from 2.10.4 above). This result has inspired Maybank (19&6) to devise a "motion without correspondence" algorithm which derives time to impact from the change over time in the *area* subtended by a small object.

2.11 On living with ambiguity

It can hardly be said that we fail in the narrow-angle case unless we are so unrealistic as to demand accurate determination of the scene and the motion from a single narrow optic flow image, the rigidity (and perhaps pknarity) assumption and the laws of optics. 'Consider the following experiment on a human being:

> The subject is drugged and placed in an aircraft which climbs to altitude and performs a flick roll (a generalised form of spin which is not necessarily about a vertical axis). They are awakened and permitted a narrow-angle view through a porthole for 1/I6th of a second. They are thai hooded and asked to estimate ptjameters of angular and rectilinear motion as well as the momentary orientation of the ground plane relative to the Hue of sight- (Care has been taken to select an acrobatically inexperienced subject and to disable all their non-visual sense organs).

Even assuming collaboration in this bizarre experiment it is unlikely that their account will be dcwc to the truth of the situation. Nonetheless, even though it is obviously true

that no instant of perception can be meaningful in total isolation (whatever that may mean!), there is an obvious sense in which "time deep" experience is composed of "instants". It is thus of acute interest to determine what information (what constraint on experience-over-time) is in principle accessible at any moment.

The relief-scale ambiguity is closely allied to the "registration problem" in stereo vision. Given two halves of a stereogram I can fuse them by crossing my eyes. The actual vergence angle between my eyes bears no relationship to the vergence angle appropriate to the stereo pair, of course, and I must somehow infer the latter if I am to unambiguously perceive a 3-dimensional scene. With perfect data I can, in theory, compute the vergence angle (corresponding to the parameter B in the motion case) on the basis of a geometric analysis which takes into account the *vertical* shifts in point-position between the two images. But vertical disparities are second-order and tend to be tiny (except in very wide-angle stereo pairs). And in general they can be eliminated, by suitable choice of scene or by corruption of the images, without affecting my ability to form an interpretation. It follows that I must be bringing in other structural considerations of some sort to select - from the one-parameter family of optically allowable scene interpretations - the "best".

Helmholtz (1925) noted that the eyes do not appear to act in concert as a pre-calibrated range-finder. Blank (1978) reports results which strongly suggest that the stereo system does not use proprioceptive information (or the vertical disparity effect) in determining vergence.

In humans the situation with moving imagery is similar to that with stereo. Images sequences can be constructed in which the second-order flow effects essential to a purely kinematic, rigidity-based, recovery of scene and motion are eliminated. This does not prevent the human visual system from making interpretations which are firmer than can be justified on the grounds of the rigidity principle and the laws of optics.

In a computer vision application the relief-scale ambiguity remaining after an application of one of the algorithms described might be removed in a number of ways:

1) The task may be one of model-matching, in which case the relief-scale parameter joins parameters of orientation, scale etc which have to be instantiated.

2) Something may be known about the motion in advance - either in precise terms or in the form of a probability distribution on the basis of which a maximum likelihood estimate can be made. Advance information may derive from such sources as the performance envelope of the relevant device or, more immediately, from preceding images in the flow sequence.

3) Any of a bewildering variety of "generic" flow-field, structural, motion, or structure-motion constraints may be brought in to regularise the interpretation. This might be construed as deployment of a "default" context appropriate in the absence of sufficient situation-specific data. While regularisation on the basis of such principles

as compactness, symmetry or General Viewpoint is constantly discussed in the field of static imagery it receives scant attention in the area of moving imagery. I can only attempt to redress this imbalance to a very small extent in the course of further discussion.

2.12 Non-rigid motion

The eight degrees of freedom in equation (2.3.1) prompt us to enquire whether we should allow eight degrees of freedom - rather than five - in the motion of scene points. We might, perhaps, thus make a virtue of necessity (notwithstanding my remarks in section 2.7!) The equations which describe rigid motion in 3-space are:

$$\begin{aligned} dX/dt &= -U - BZ + CY \\ dY/dt &= -V - CX + AZ \\ dZ/dt &= -W - AY + BX \end{aligned} \qquad (2.12.1)$$

If we allow *shear* in 3-space - as well as rotation and translation - equations (2.12.1) would take on the more general first-order form:

$$\begin{aligned} dX/dt &= -U - bZ - aY \\ dY/dt &= -V - dX + cZ \\ dZ/dt &= -W - fY + eX \end{aligned} \qquad (2.12.2)$$

where a, b, c, d, e, f can take on any real values. The corresponding generalisation of equations (2.2.1a) and (2.2.1b) - whose derivation I will not show here, are:

$$u = (Wx - U)z - b + ay - ex^2 + fxy \qquad (2.12.3a)$$

$$v = (Wy - V)z + c - dx - exy + fy^2 \qquad (2.12.3b)$$

And from these we can derive the corresponding form of equation (2.3.1):

$$\underset{Q_6}{x^2(Ve + Wd)} - \underset{Q_5}{xy(Vf + Ue)^2} + \underset{Q_4}{y(Wa + Uf)} - \underset{Q_3}{x(Wc + Ud)} - \underset{Q_2}{y(bW + aV)}$$

$$+ \underset{Q_1}{(Vb + Uc)} + u(V) - y(U) + (vx - uy)W = 0 \qquad (2.12.4)$$

Unfortunately the nine parameters Q_6.....Q_1, V, U, W are not independent. It is easily shown that:

$$Q_2VW - Q_1W^2 + Q_3UW - Q_6U^2 + Q_5UV - Q_4V^2 = 0 \qquad (2.12.5)$$

There is thus a one-parameter family of motions involving unconstrained shear of the object/scene corresponding to any solution of the flow-field equation (2.12.4) which conforms to (2.12.5). A similar ambiguity arises if we generalise 3-D motion to include linear "divergence" terms (making dX/dt a function of X etc) rather than shear terms. In fact a full first-order generalisation of - to allow unconstrained shear and divergence - gives us a 12-parameter family of scene-motions from which we can derive a flow-field equation of the familiar "LH1" form with 9 terms and 8 degrees of freedom since condition (2.12.5) continues to apply. We might be tempted to place some restriction upon the motion to give us an 8:8 correspondence (or a 7:7 correspondence if we fix the depth-scale ambiguity) - but any such restriction is bound to very arbitrary. A more attractive alternative is to allow the full 12 degrees of freedom in the scene and "regularise" the interpretation of the flow parameters by minimising computed deformation of the scene. Adding dilation terms g, h and i to (2.12.2):

$$\begin{aligned} dX/dt &= -U - bZ + aY + gX \\ dY/dt &= -V - dX + dZ + hY \\ dZ/dt &= -W - fY + eX + iZ \end{aligned} \qquad (2.12.6)$$

The corresponding flow-field equation is:

$$\underset{Q_6}{x^2(Ve+Wd)} - \underset{Q_5}{xy(Vf+Ue+W(h-g))^2} + \underset{Q_4}{y(Wa+Uf)} - \underset{Q_3}{x(Wc+Ud+V(g-i))}$$

$$- \underset{Q_2}{y(bW+aV+U(i-h))} + \underset{Q_1}{(Vb+Uc)} + u(V) - v(U) + (vx-uy)W = 0 \qquad (2.12.7)$$

From U,V,W and Q_1.....Q_6 we obtain six linear equations involving the nine unknowns a, b, c, d, e, f, g, h, i.

$$Q_6 = Ve + Wd \quad \text{etc.} \qquad (X.1.....X.6)$$

To these we add six further "rigidity" conditions:

$$\alpha a = \alpha d,\ \alpha b = \alpha e,\ \alpha c = \alpha f \qquad (X.7,\ X.8,\ X.9)$$

$$\beta g=0,\quad \beta h=0\ ,\ \beta i=0 \qquad (X.10,\ X.11,\ X.12)$$

where α and β are "weights" reflecting our confidence in the assumptions that there is no shear and that there is no dilation respectively. If we perform a least squares regression over the twelve equations X.1 to X.12 we obtain a solution for the nine motion parameters

which minimises:

$$|\text{prediction error}|^2 + a^2 (|a-d|^2 + |b-d|^2 + |c-f|^2) + 0^2 (g^2 + h^2 + i^2) \quad (2.12.8)$$

Note that our six "data" equations are not independent (condition 2.12.5). We effectively have only five. We have (U, V, W being determined) nine scene parameters to estimate. Regularising on either shear or dilation alone (each equivalent to adding three equations) is thus not sufficient. If we set *a* » /3 then we will recover the interpretation which is close to the "minimum divergence" member of the one-parameter family of "shearless" interpretations. Similarly if we set /3 » a we will recover the "divergence free" interpretation which has the minimum shear. If we set both *a* and 0 very high - relative to the weight accorded to the first six equations - our least squares regression over equations XI to X12 will return with the best possible estimate of the parameters of a *rigid* motion.

I cannot imagine that there is any possibility of this algorithm being used seriously on real imagery- The deformation (of the whole scene) permitted may be mathematically natural but it is empirically rather rare. I have given the analysis mainly as an illustration of regularisation and a reminder that the rigidity constraint is not an immutable law of nature.

The deformation for which we allowed above is globally organised deformation. This picture is very different from rigidity-maximising approaches which assume deformation to be statistical (equivalent to noise added to 3-space motions) or "rubbery" (equivalent to a sort of cumulative 3-space noise). An approach appropriate to such a situation is that of Ullman (1979, 1983b) whose algorithm may be pictured as attempting to minimise the energy expended in stretching or compressing springs connecting points making up the object. An alternative measure of deformation of a point-object is my "alpha code" (Scott 1983, 1984). All triplets of points are examined to generate "alpha facts" of the form: AC > AB. A "unit of deformation* is counted for each such inequality that changes sense between images. The method is surprisingly sensitive in defining configurations and in detecting deformation though it has been known for some time in the field of Multi-Dimensional Scaling that ordinal information on the relationship between points in a space may serve to rather tightly define the actual configuration (Shepard 1966; Coxon 1982).

Allowing global first-order deformation of the scene is also unlike local-rigidity approaches which assume hinged objects or which invoke more general segmentation into dense, perfectly rigid, gubpazts- Hoffman's (1980) method of recovering biological motion from images of lights attached to joints (Johansson 1973) assumes hinged motion together with planar motion of complete limbs. Most optic flow segmentation algorithms (see Chapter 5) implicitly or explicitly strive to decompose a non-rigid image into sub-images to which the assumption of rigidity (and therefore the sort of analysis we have conducted earlier in this chapter) may be applied.

2.13 Smooth but non-planar objects

In deriving LH2 a linear expression (Lx + My + N) is substituted for inverse depth in equations (2.1). A natural development is to substitute a second-order expression - thus allowing more degrees of freedom in the scene structure. Waxman and UUman (1984), Eastman and Waxman (1985) and Negahjdaripour and Horn (1985a) have pursued this "quadric" extension of the logic of LH2. Maybank's (1985) treatment of the Taylor series expansion of the flow-field is in similar pursuit of high order surface structure.

Strictly speaking a quadric patch will in general give rise to an inverse depth field of infinite order - but this may be reasonably approximated to second order.

Waxman's approach is to recover all flow-field components up to second order and then to recover from these the motion parameters and the scene parameters (which now include "curvature[11] terms as well as orientation terms). His method of effecting this recovery is similar to those originally developed by Longuet-Higgins for LIE and by Longuet-Higgins and Prazdny (1981).

In addition to recovering higher order flow-field terms (and interpreting them as due to complex structure though they could as well be due to deformation) Waxman seeks to extend his analysis over a protracted period of time to obtain and interpret changes in image velocity over time.

But the segmentation problem (in either its spatial or temporal guise) tends to put obstacles in the way of "going global" by the straightforward method of raising the order of polynomials composed of observables. In its present form Waxman's algorithm, and its relatives, are probably most interesting for the approach to vision which they exemplify: an approach biased more towards a direct, high-order interpretation of the sensory array than a painstaking "stitching together" of local interpretations.

CHAPTER 3

COMPUTATIONAL RECOVERY OF SCENE AND MOTION PARAMETERS

3.1 Introduction

This chapter is focussed on the practical application of the theoretical analysis of the last. In both the planar and general cases we have a choice of solution procedures which range from fast, closed methods to exhaustive search over the solution space. We rapidly discover that the first type, exemplified by LH1 and LH2 and variants thereof, present us with problems in conditions of noisy data. It is in general not possible to acquire the "least squares" solution - on any reasonable noise model - by a straighforward application of these algorithms.

There are cases in which it is clearly desirable to regularise the solution by various means, but it can be difficult to penetrate the closed form of some of the algorithms discussed in the last chapter. Even a simple stabilising measure such as preventing the inferred depth of a point in space from being negative is hard to bring to bear when point-depth has been algebraicly expelled from the model!

I use real data from the bus imagery, as well as some synthetic data, to explore the effectiveness of a number of computational procedures. The results tend to support the case for iterative algorithms with "exposed workings" - either instead of or in support of closed-form algorithms.

3.2 Error-handling in the planar case - a cautionary tale

Obtaining a solution to an overdetermined set of equations in conditions of corrupt data involves defining a cost function which may be composed of:

- an error term

- other "regularisation" terms

The model of image corruption most commonly employed in generating and interpreting synthetic imagery is one of additive, Gaussian noise applied to the local (point- or edge-motion) flow vector. The most common measure of "cost" applicable to this model is the sum of the squares of the imputed difference between the data and the predictions of the interpretation.

The Gauss-Markov theorem, which proves that the "ordinary least squares (OLS) estimator"

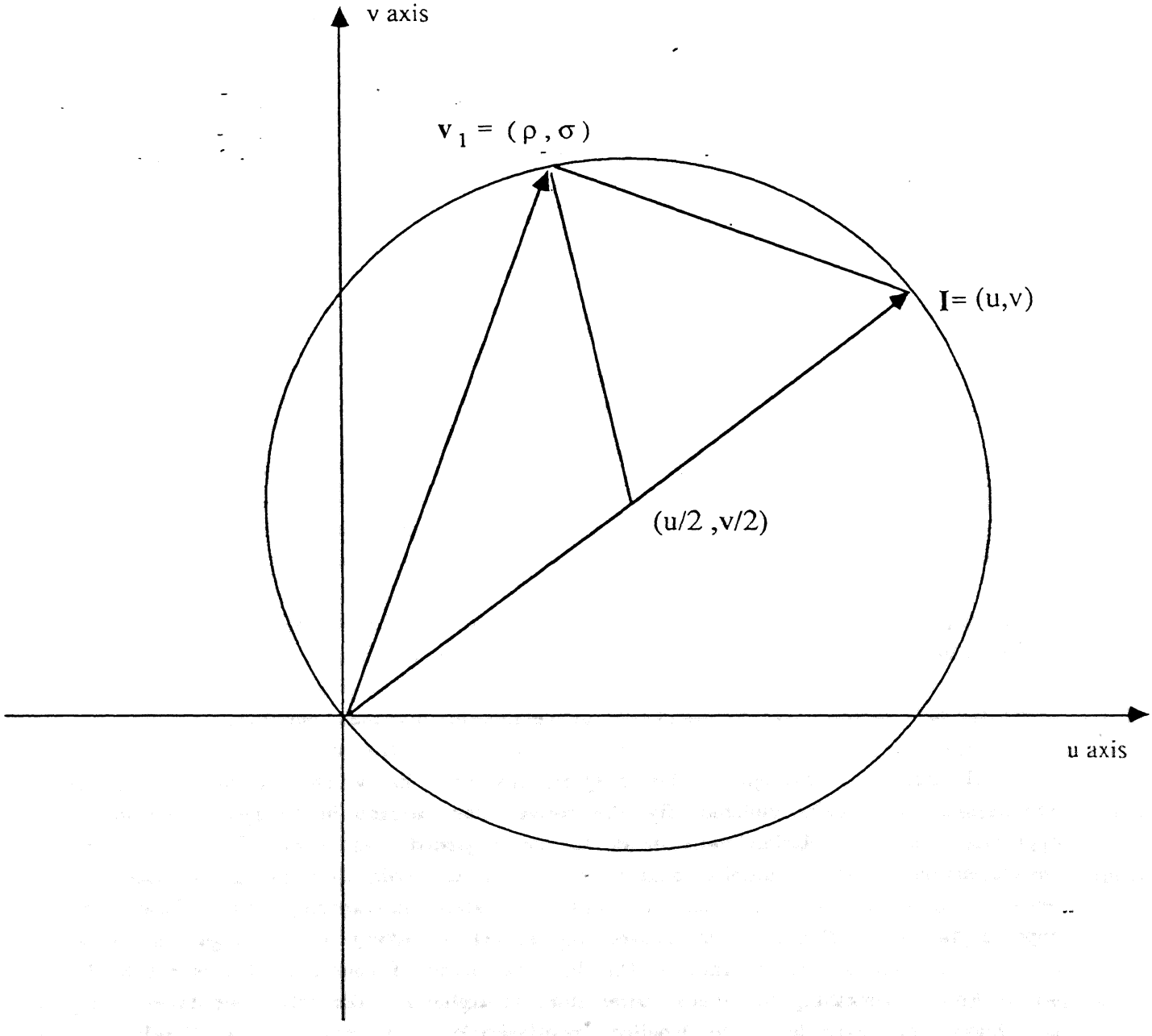


Figure 3.1 Relationship between edge- and point-motion for analysis of error (see text)

is the "best" invokes a number of assumptions (see Bibby 1977 p 44):

A: The values of the dependent variable(s) (in our context the image flow vector component or components) are in fact generated by the linear model which we are fitting.

B: The values of the independent variables are known without error

C: Nothing is known *a priori* concerning the values of the parameters of the model.

D: Each disturbance has the same variance.

E: The disturbances are uncorrelated with each other.

F: The disturbances are statistically independent of each independent variable

G: We require estimators of the model parameters which are linear in the dependent variables

H: We require our estimator to be unbiased (equal to the true value "on average").

I: We are willing to judge the quality of our estimators by their variances.

We will encounter, throughout this chapter, instances in which one or more of these assumptions are rudely violated. By the conventions current in Computer Vision action appropriate to a violation will tend to be regarded as either "error-handling" or "regularisation". If, for example, assumption D does not hold then the appropriate step of rescaling equations to ensure that it does (equivalent to carrying out a "weighted" as opposed to an "ordinary" least-squares regression) is likely to be regarded as good, conventional statistical error-handling. On the other hand, if condition C does not hold - if we *do* known something the model parameters in advance - the action we take to deploy this knowledge is liable to be labelled "regularisation". I adhere fairly closely to this distinction despite misgivings.

The remainder of this section is concerned to highlight the "error-handling" problem with the particularly striking example, encountered in practice, of synthetic edge-motion/planar-object images subject to additive Gaussian noise.

Figure 3.1 shows a full motion vector **I** and an edge-motion vector v_1 which represents its component in one direction. The circle represents the locus of possible edge-motion vectors corresponding to **I** and has radius $|I|/2$. The geometrical interpretation of the original (Buxton) solution to the aperture problem is thus:

Vj has components p , cr. The centre of the circle is u/2, v/2.

$(p - u/2)^2 + (cr - v/2)^2 \sim III^2/4$

$\rightarrow \quad p^2 + cr^2 - pu - (Tv + u^2/4 + v*/4 - 11\text{^}/4 \qquad (3.2.1)$

$\rightarrow \quad Ivjl^2 \ll pu + crv$

which is analytically equivalent to the solution described in the last chapter if we note that p/lvjl — cos(0), cr/lvjj — sin(6). However, when I empirically compared (in 1984) the original Buxton algorithm, based on the formulation just given, with one based on my own formulation I noted .that the former was very much less accurate than mine and embarked upon an analysis of the error terms.

Consider the case in which v_x is subject to (additive) noise. We can confine our attention to the component of this (call it *e)* which is *normal* to the circumference of the circle - since the component tangent to the circle does not affect the accuracy of (3.2.1). The RHS of (21) should now read (ill + eXHI + €) which, ignoring e^2, is $III^2 + 2III6$ viz. the expected error is proportional to the magnitude of the point-motion vector I at the point. Points with large true motion thus exert an undue (and baleful) influence upon the result of a least squares regression unless the data associated with them is scaled down appropriately. We do not know this magnitude III initially but Ivjj is a good estimator for it, the *expected* value of III/lvjI being the same at each point. My equations are thus unbiased (properly scaled) for LSR, given the information initially to hand.

Once a regression has been run we can form direct estimates of I at each point. We can now run a second regression - with the data scaled according to our estimates of III, instead of by tv^l. We can then run a third regression - based on our new improved estimates of L This sort of iterative procedure is characteristic of fitting a circle or ellipse by LSR (for a brief account see e.g. Ballard and Brown 1982 pp 487-488).

The typical effect on the edge-motion version of LH2 of these error-handling precautions is, in the case of fairly *small* amounts of noise, to reduce error in recovering flow-field parameters by a factor of about ten. (Buxton et al 1984). The argument just outlined is thus- more than a statistical quibble. It applies not merely to LH2 (in the edge-motion case) but to ail situations in which only one component of full motion is known *and* the noise model is appropriate. These would include smoothing algorithms such as those of Horn and SchxmcJt and of Hitdrdh which are discussed in the next chapter.

Let us[1] note, however, that the noise model just discussed is *not* appropriate to the case of Imagery - real or synthetic - from which motion constraints have been obtained by the Horn and Schundk ©r by tbe principal axes procedure. The noise story in the Horn and Sehunck case is complicated - and 1 am not satisfied that 1 fully understand it. The motion constraint equation is computed* let us recall, by fitting a linear function to the grey-level surface in tbe viciaity of each pixel in the second image:

$$E = E_2 + au + bv \tag{1.1}$$

Here u and v (the displacements from the reference pixel) are correctly treated as errorless independent variables (condition B above) and E as the "errorful" dependent variable. If we grant the explicit linearity assumption (A above) and the implicit assumptions that E is subject to additive Gaussian noise for which the various independence conditions (D to F) hold then it seems reasonable to determine a and b by a least-squares procedure. But when we move on to the motion constraint equation and to analytical and/or smoothing procedures for determining full motion we find ourselves violating the conditions required by the Gauss-Markov Theorem.

$$(E_1 - E_2) = au + bv \tag{1.2}$$

$(E_1 - E_2)$ - which is effectively the dependent variable in ordinary least squares treatments - has, to be sure, additive Gaussian error independent of its magnitude associated with it. But we are now taking a and b (perhaps in combination with image coordinates if we substitute analytic expressions for u and v) to be explanatory (independent) variables. But we know perfectly well that condition B - error-free knowledge of the independent variables - does not apply since we have just computed a and b by statistical means. Error in (1.2) is thus bound up as follows:

$$(E_1 - E_2) = (a + \epsilon_1)u + (b + \epsilon_2)v + \epsilon_3$$

$$= au + bv + (\epsilon_1 u + \epsilon_2 v + \epsilon_3)$$

where ϵ_3 is the error in $(E_1 - E_2)$, and ϵ_1 and ϵ_2 are the errors in a and b respectively. *Qualitatively* the picture is similar to that for the simpler noise model discussed above: the greater the magnitude of true motion {u v} the greater the expected error in satisfying the constraint equation. But it is more difficult to "reweight" the constraint equations appropriately. We require some estimate of the relative variance of ϵ_1, ϵ_2 and ϵ_3. This problem requires a rather sophisticated treatment and is not of immediate concern to us, since our local flow information is not derived by the Horn/Schunck procedure.

Because my principle axes procedure addresses the expected error in each constraint directly it does not appear to suffer from the problem of "compounded error terms" in the same way as the Horn/Schunck constraint equation. I believe it is sound to use our motion constraint data "as are" in least squares estimation.

There is a double caveat contained in the tale told in this section:

(1) Derive the right noise model

(2) Use fitting methods appropriate to that model

It may be difficult to comply with either of these requirements but it is obviously

important to do so. When first investigated empirically LH2 appeared considerably less robust than it actually is.

Note: LH2 invokes a two-stage process - recovery of field parameters from image data, followed by recovery of motion and scene parameters from these. It can be simply shown that, where the translational motion vector {U V W} is nearly parallel to the surface normal {L M N}, the estimate of these two vectors is very sensitive to error in field parameters. This sensitivity does not appear to be the fault of the *algorithm* itself - since it is empirically evident in algorithms based on numerical methods. At the opposite extreme, where motion is parallel to the plane, flow-field parameter error leads to comparable scene parameter error.

3.3 Error in the general case

In the case of LH1 we have two serious problems in noisy conditions. First we have the fact that are eight degrees of freedom in the linear equations for the flow-field (versus the five which the model allows). A good fit to the flow-field does not, therefore, necessarily imply a good fit to the model of rigid relative motion. The second problem relates to the manner in which data-error is "bound up" in equation (2.3.1). It is helpful at this stage to briefly remind ourselves of the two simplest ways in which error can appear in linear equations and of the appropriate method of least-squares solution in each case.

Error may be associated with a *dependent* variable which is to be "explained" as a linear function of a number of independent variables which are free of error. Here we have a set of equations of the form:

$$y + \epsilon = ax_1 + bx_2 + cx_3..... \tag{3.3.1}$$

where ϵ is the error in measuring y and x_1, x_2, x_3 etc are "errorless". Where ϵ is Gaussian the least-squares regression produces the maximum likelihood estimate of the parameters a, b, c etc. as those values which minimise:

$$\Sigma(y - ax_1 - bx_2 - cx_3....)^2 = \Sigma(e)^2 \tag{3.3.2}$$

where the sum is over all data points and e is the estimated value of ϵ for each observation. The second straightforward case is the one in which error is associated with "dependent" variables in a set of homogeneous linear equations. Here we have:

$$a(x_1 + \epsilon_1) + b(x_2 + \epsilon_2) + c(x_3 + \epsilon_3)..... = 0 \tag{3.3.3}$$

where ϵ_1, ϵ_2, ϵ_3 etc are independent errors in observing each of the quantities x_1, x_2, x_3 etc. These are assumed to be Gaussian and to have the same expected magnitude. We have of course the trivial (and only exact) solution to a set of such equations: a = b = c =...= 0, but this is of no interest. We want the direction of the *unit* vector {a b c ...} which

minimises the imputed error in the data:

$$\Sigma (e_1)^2 \quad + \quad \Sigma (e_2)^2 \ldots\ldots \tag{3.3.4}$$

where e_1, e_2 etc are our estimates of actual error ϵ_1, ϵ_2.... Summation is over all data points.

If we fix one of the terms in this equation, for example if we set a = 1 we obtain:

$$b(x_2) + c(x_3)\ldots. = - x1 - \epsilon_1 - b\epsilon_2 - c\epsilon_3.. \tag{3.3.5}$$

If we now attempt to solve for b, c etc by least-squares regression we obtain the solution which minimises the value of:

$$\Sigma (e_1 + be_2 + ce_3\ldots.)^2 \tag{3.3.6}$$

which has the same expected value as:

$$\Sigma (e_1)^2 \quad + \quad b^2 \Sigma (e_2)^2 + c^2 \Sigma (e_3)^2 \ldots$$

since the cross-terms arising when (3.3.6) is multiplied out are expected to disappear (the errors being assumed to be independent and of average value zero).

Clearly this is not the same quantity as (3.3.4) - since it is a function of the solution parameters. It "pays" a least squares regression, in this situation, to underestimate b, c etc. since this tends to reduce the *apparent* error (3.3.6). To put it another way: we have not fixed the magnitude of the vector {a b c....}; only the magnitude of one of its components. The "apparent error" in a set of homogeneous equations is a function of the magnitude of the solution vector (recall that if the vector is null the apparent error is zero). So a least squares regression works to reduce the magnitude of the solution vector. The norm (3.3.4) is the sum of squares of the perpendicular distance, in observation space, from the observations to the solution plane. What we seek, in effect, is the minor principal axis, through the origin, of the "cluster" of data points in $x_1x_2x_3$.... space. The solution plane which minimises the norm (3.3.4) is that normal to the minor axis. There are no closed-form methods of obtaining this beyond four dimensions - since the characteristic equation of an N×N matrix must be solved (to yield the eigenvectors which are the axes) for an N-dimensional cluster.

The error in the LH1 flow-field equation (2.3.1), alas, is bound up in a manner which is suitable for neither simple least squares regression nor for a straightforward principal-axis type of treatment. In fact LH1 appears to be entirely refractory to accurate closed-form solution. We will see below that, even in the simplified case in which pure translation is assumed, there may be large discrepancies between the true minimum-error solution and that discovered by a closed-form procedure. The largest discrepancy factor I have heard quoted - for the discrete displacement version of LH1 during empirical trials at GEC Hirst

Research Laboratory - is 10^{13}. Even if this number is exceptional we have cause to attend to the boring statistical details!

3.4 Regularisation

The mathematical paradigm of the "ill-posed problem" has a long history in other fields but has been brought to bear on computational vision quite recently (Poggio and Torre 1984; Poggio 1985). An ill-posed problem as defined by Hadamard (1923) is one which fails one or more of the following tests:

- it has a solution which exists
- that solution is unique
- that solution depends continuously upon the data (i.e. it is robust against noise)

If a problem is ill-posed then, according to the theory, it stands in need of "regularisation" to render its solution existent, unique and stable. The commonest and simplest form of regulariser or "stabiliser" is a norm (usually quadratic to ensure uniqueness) which the solution is to minimise. The ubiquitous sum of error-squared is thus a regulariser. A simple least-squares regression turns a problem whose solution does not exist (an overdetermined and inconsistent set of equations) into one whose solution is unique and stable (at least in the technical sense that the solution is a continuous function of the data).

But in practical terms, in many applications, a least-squares solution may be highly unstable. It may even be inapplicable because there are fewer equations than unknowns. (The problem of determining A and B from the single equation A + B = 10 is of course ill-posed). To derive satisfactory solutions we need to "import" more information. Where this outside help takes the form of the introduction of generic constraints then we would regard it as regularisation. The simplest form of generic regulariser is a (weighted) sum of squares of the parameters of a polynomial - such as the Tikhonov stabiliser (Tikhonov 1963; Tikhonov and Arsenin 1977). The "meaning" of such a stabiliser varies with context. In section 2.10 I introduced such a stabiliser as a measure of departure from rigidity. I will give an example later in this chapter of such a stabliser deployed to implement a priori assumptions of a conservative "general viewpoint" type.

It is unclear to me whether a hard-and-fast distinction between generic, as opposed to context specific, constraints can be upheld. It is also hard to know whether some basic measures which may be taken to restrict the solution space deserve the appellation "constraint" or "regulariser" at all. Take for example the restriction that the depth of a visible point must be positive. With perfect data LH2 gives rise to at least two solutions (out of four) which are optically impossible. With noisy data both LH2 and LH1 may give only optically impossible solutions. In such a case we are naturally tempted to iteratively push the solution to a state in which the laws of optics are not violated. Is this

regularisation or is it merely taking into account, albeit in a belated and messy fashion, a feature of our model which the algebra did not want to digest?

Ambiguity is a persistent feature of human perception - at least in impoverished situations. Humans appear to live quite happily with the depth-scale ambiguity in the absence of scaling cues and there are innumerable examples of both static and moving imagery which causes "flipping" from one interpretation to another. What is the justification for an insistence upon *uniqueness* of a solution in the case of a computer vision program? Would a computer program which suffered from the Necker cube illusion stand in need of further regularisation? This is a quite fundamental issue. There are many situations in which different basic constraints or principles are brought into conflict with each other. In a rotating cylinder display there is a conflict between the tendency to see a planar configuration of points and the tendency to see a rigid configuration. Humans tend to see one *or* the other - they do not appear to see the mixed interpretation which would correspond to minimising a sum of quadratic planarity and rigidity "norms". The evidence points to a strategy which necessarily leads to ambiguity: given an overdetermined set of conditions which cannot be satisfied, find a subset which *can* be satisfied. This is somewhat like minimising a non-elastic norm (e.g. the sum of the square-root of error) over a set of equations. The solution will not, in general, be unique.

In practical terms the problem confronting us in the context of interpreting optic flow imagery is one of finding plausible ways of restricting the solution space without becoming excessively narrow-minded about the situations which might arise in the world! I will occasionally mention and illustrate various possibilities for improving upon solutions without insisting on their theoretical status.

Whatever our means of improving upon interpretations, and whatever name we choose to call them, we are faced with the task of bringing them to bear on the process of recovering scene and motion parameters. This is not necessarily straighforward with algorithms such as LH1 and LH2 in which structural parameters (e.g. point depths in LH1) have been eliminated or (e.g. the surface normal vector in LH2) compounded in a complex way with other unknown quantities. The need for integration of information from a variety of sources pushes us in the direction of algorithms with exposed workings.

3.5 Applying LH2

With LH2 Longuet-Higgins discovered that, in solving for the *scene* parameters, given the flow-field parameters, there exists (in additional to the depth-scale) a two-fold ambiguity in that:

> given three vectors {U V W}, {L M N} and {A B C} which are a solution to the equations relating scene/motion and flow-field parameters, there exist three other sets of vectors which are also algebraic solutions. These are:

$$\{L\ M\ N\},\ \{U\ V\ W\},\ \{A'\ B'\ C'\}$$

in which the translational motion and the normal to the plane are interchanged (the rotational vector is different) and:

$$\{-U\ -V\ -W\},\ \{-L\ -M\ -N\},\ \{A\ B\ C\}$$

$$\{-L\ -M\ -N\},\ \{-U\ -V\ -W\},\ \{A'\ B'\ C'\}$$

These are the "back-to-front" versions of the first two. Now two of the four solutions can always be immediately rejected if the data are perfect since they imply negative depths for visible points. Longuet-Higgins found that under certain plausible circumstances, a third solution could also be dismissed as spurious on the same grounds. This arises as follows:

An indefinitely extended plane, not lying orthogonal to the line of sight, viewed in perspective projection, gives rise to an horizon (which may not be in the image if the angle of view is restricted). Visible points (if the planarity assumption holds) may not lie "above" the horizon in the image. A point lying on the plane projects to image coordinates x, y and the following equation holds:

$$Lx + My + N = 1/Z$$

Clearly it must be the case that $Lx + My + N \geqslant 0$ for a point (x, y) to be visible. $Lx + My + N = 0$ defines the horizon. It cannot be the case that all points are visible under two interpretations which are related by a change of sign unless all points lie on the horizon (in which case we would not have got to this point with LH2).

If the planar normal and translational vector are transposed we obtain the "pseudo-horizon":

$$Ux + Vy + W = 0$$

which is the intersection of the image plane with the plane through the pinhole and normal to the direction of motion. So long as all visible points do not lie on one side of this pseudo-horizon (as long as they are not all ahead of, or behind, the camera in respect of its direction of travel) we are able to reject the remaining spurious interpretation.

Though with perfect data there must exist at least one interpretation which complies with the visibility condition it is possible, with error in the data, to arrive at a situation in which all four putative solutions are optically impossible. If there are points on or close to the horizon this might be a very powerful source of regularisation - particularly in preventing excessive slant estimates where the plane is truly quite sharply slanted to the viewer.

It is straightforward to check for invisible points once a putative solution for scene

parameters has been obtained. (This should be done routinely to eliminate spurious solutions in any case). The problem is how to *enforce* the visibility condition efficiently where there is error in the data. The plane parameters appear in the field equations only in combination with motion parameters - making it difficult to influence their values by "biasing" the regression for the field parameters.

One might envisage an iterative search for the best "visible" solution which would involve:

1) Estimating field parameters

2) Solving for scene parameters

3) Identifying "invisible" points

4) Adjusting scene parameters to produce visibility

5) Recomputing field parameters from these and repeat step 1 using these adjusted values to "bias" the regression.

This is a terribly clumsy and heavy-handed approach which I have not attempted to implement.

For the moment I let the horizon problem stand as an illustration of how difficult it is in general to bring in outside information to help regularise the two-stage recovery of scene parameters in LH2 - even where such outside help is purely a matter of elementary optics.

Now, returning to the first stage of LH2 we note that simple least-squares estimates of flow-field parameters may run riot in the presence of small amounts of noise where the viewing angle is narrow. We may adopt a variety of stabilisation approaches which are analogous to general viewpoint, smoothness or other "conservative" principles. There is indeed no reason why such principles should not be important in the interpretation of multiple imagery - in view of the fact that they are absolutely *crucial* in the case of static imagery.

The simplest and commonest form of stabiliser used in the case of a linear model is one which is composed of a sum of the squares of the parameters being sought. In the specific case of the first tage of LH2 this would mean that we seek the solution which minimises:

$$\Sigma\, |E|^2 \quad + \quad \Sigma\, |\alpha_i P_i|^2 \qquad i = 1 \text{ to } 8 \tag{3.5.1}$$

where the P_i are the flow-field parameters and the α_i are weights associated with each of them. The first sum is over all data points and the second over all flow-field parameters.

What is the "meaning" of the weights α_i in (3.5.1)? We might derive and justify them by an argument such as this: In a noisy and/or edge-motion image we are required to estimate

the true, full motion. In keeping with the principle that interpretations should be no more energetic or complex than the data forces them to be we want to infer a true flow-field which involves a relatively small amount of image motion. Motion everywhere in the image is aspiring, on this argument, to the stodgy condition of no motion. We can incorporate such inertia in a least-squares regression procedure simply by adding false data appropriate to zero motion. The amount of such data which we added - the weight which we according the zero-motion hypothesis - would reflect our lack of confidence in the data. Given this it is straightforward to derive the functional of form (3.5.1) which a least squares regression over both the genuine and the phony data minimises.

It is difficult, with synthetic imagery, to assess the value of stabilisation. Implicit in any stabiliser is a notion of the a priori probability distribution of scene-and-motion configurations. If synthetic images are generated in accordance with the appropriate probabilities then the stabiliser will, obviously, tend to work. Recall that my procedure for estimating constraints on local motion is inherently conservative. (Overall the 64x64 data underestimates the magnitude of image motion by about 25 percent and the 132x132 data by between 15 and 20 percent). I would certainly not be inclined therefore to add stabilising terms, of the sort just discussed, to a computation over the bus data.

It seems that the human visual system may have recourse to "principles of good motion" in some situations. Scott and Buxton (1985) draw attention to a class of moving displays in which humans see the translational motion vector and the surface normal as either orthogonal or parallel to each other - despite the fact that the imagery allows arbitrary relative orientations. The effect appears homologous to the tendency for straight lines in static images to be seen as orthogonal in 3-space. It is difficult to guess how and why such regularisation - if that is what it is - comes to be applied. It may be that humans have acquired through experience a disposition to see structural and motion axes as simply related in the absence of any firm indication that they are *rwt* so related.

Whatever the case it is not possible to enforce such a "good motion" constraint directly in LH2 - though it is straightforward enough in the iterative procedure "N&H" which I will describe later.

3*6 LH1 *in* the case of pure translation.

To simplify the discussion of certain issues I first assume that there is no rotation and that we know there is not. Equation (2.3.1) becomes

$$- v(U) + u(V) + (vx - uyXW) - 0 \qquad " \qquad (3.6.1)$$

which defines the direction of image motion at any point, directly toward or away from the **focus** of expansion at fU/W **V/W}»** but not its magnitude (which depends on z).

If the data is free of error there are many ways of solving a **set** of equations of form

(3.6.1) to give us the direction of {U V W}. If there is error in the data we likewise have a choice of solution methods but they will each tend to yield a different result.

With noise terms ϵ_1 and ϵ_2 added to u and v respectively we have

$$-(v + \epsilon_2)U + (u + \epsilon_1)V + (x(v + \epsilon_2) - y(u + \epsilon_1))W = 0 \qquad (3.6.2)$$

Gathering terms we have a total error, per equation, of:

$$\begin{aligned} F &= -vU + uV + (xv - yu)W \\ &= U\epsilon_2 - V\epsilon_1 + W(y\epsilon_1 - x\epsilon_2) \\ &= -(U - Wx)\epsilon_2 + (V - Wy)\epsilon_1 \end{aligned} \qquad (3.6.3)$$

We can set U = 1 or V = 1 or W = 1 and solve for the other two components. Suppose we set U = 1. From equation (3.6.3):

$$uV + (xv - yu)W = v + (\epsilon_2 - V\epsilon_1 - W(y\epsilon_1 - x\epsilon_2)) \qquad (3.6.4)$$

Clearly V and W act as "amplifiers" of error and a least squares regression will therefore tend to underestimate these two parameters. Similarly if V or W are fixed the estimated motion will be biased in the Y or Z directions respectively.

Bruss and Horn (1983) describe a closed-form method, very closely related to a principal axis transform, of solving for U, V and W subject to the constraint:

$$U^2 + V^2 + W^2 = 1$$

This would be entirely appropriate if the model were, say:

$$-(v + \epsilon_1)U + (u + \epsilon_2)V + (w + \epsilon_3)W = 0$$

with u, v and w and ϵ_1, ϵ_2 and ϵ_3 satisfying the appropriate independence and variance conditions. But this is not the model. Their procedure minimises the sum of squares of F and this is not the same thing as minimising the sum of squares of estimated error in the data. The measure F, for any point, will tend to be small if the two quantities (U - Wx) and (V - Wy), by which the true errors ϵ_1 and ϵ_2 are multiplied, are small. Bruss and Horn's procedure thus tends to underestimate these quantities over the data as a whole. Geometrically, the focus of expansion moves toward the centroid of the data in image space.

(Bruss and Horn also discuss a version of their procedure in which, effectively, the magnitude of all image motions are normalised to 1. In an image which contains motions close to "null" motions - such as our bus images - this is a formula for unmitigated

disaster).

The correct estimate of the inaccuracy of a data-point under conditions of additive Gaussian noise is (as Bruss and Horn were fully aware) the quantity E:

$$E = (-vU + uV + (vx - uy)W)/J = F/J$$

where $J = ((Wx - U)^2 + (Wy - V)^2)^{1/2}$

The expected magnitude of E is independent of x, y, U, V and W and is the same as the expected magnitude of ϵ_1 and ϵ_2.

(Geometrically E is the component of velocity orthogonal to the line joining the point to the focus of expansion. It is the "unexplained component" of image motion since the component of motion directly aligned with the focus of expansion can in principle be completely explained by assigning an appropriate depth).

Empirical results with the bus data entirely support the above analysis. The 50 "point-motion" vectors associated with the bus and shown in figure 1:5 were used as data and the vector {U V W} estimated by a variety of methods. The magnitude of {U V W} was adjusted to give the best fit to the hypothesis that z = 1 for all points. The average value of the E-norm, per point, was computed for each result. (The E-norm is the sum over all data of the square of the quantity E). The number of points "invisible" (with negative imputed depth) was calculated. Table 3:1 shows the outcome. "LH1-U" means LH1 applied with U fixed, "LH1-V" with V fixed, "LH1-W" with W fixed. B&H is Bruss and Horn's algorithm which minimises the F-norm. VP refers to the estimated vanishing point - which we presume to be coincident with the focus of expansion. The last row of estimates was obtained by an exhaustive search to minimise the E-norm.

TABLE 3:1

Source	U	V	W	f.o.e	E-norm	Invisibles
LH1-U:	1.17	-0.40	-3.15	{-0.37 0.13}	0.49	2
LH1-V:	0.46	-0.95	3.36	{ 0.14 -0.28}	0.32	2
LH1-W:	-0.42	-0.39	8.1	{-0.05 -0.05}	0.65	8
B&H:	-0.42	-0.39	8.1	{-0.05 -0.05}	0.65	8
VP:				{ 0.26 -0.24}		
Search	0.63	-0.78	3.62	{ 0.17 -0.22}	0.25	0

The LSR with W fixed and the Bruss/Horn procedure here give identical results to two decimal places, but in general they do not. In this particular case the data points are clustered rather close to the centre of the image so that the image coordinates tend to be small. It therefore "pays[11] the Bruss and Horn procedure to impute a high relative value to W in minimising the F-norm.

With the exception of the LSR with V fixed (and this may be accounted a fluke) the results obtained by closed-form methods are much worse than the result obtained by search. The optimum result was, however, rapidly obtained by an iterative algorithm (GLSl) which I describe later.

3.7. LH1 with rotation

Here is an example of the performance of LH1 with a synthetic, noisy, wide-angle image with considerable depth variation:

Data-points: 20
Semi-angle of view: 30 degrees
Inverse depth range: 0.5 to 1.5
Noise: 13 percent Gaussian added to image vectors

The rectilinear motion is normalised as previously to give best fit to z=l for all data. The rotational motion parameters are recovered from the flow-field parameters by a method described below.

TABLE 3:2

Source	U	V	f	A	B	C	(V-A)	(B+U)	E-norm
Verid	**0.08**	-0.6	**-0.8**	-0.82	-0.52	0.42	0.22	-0.46	0.38
LH1-U	**1.12**	**-1.81**	-0.21	**-1.91**	**-1.45**	0.41	0.10	-0.33	5.52
LH1-V	**0.09**	**-1.06**	-0.69	**-1.21**	**-0.61**	0.42	**0.18**	-0.42	0-47
LH1-Y	0.01	-0.38	-0.74	-0.59	-0.42	0.42	0.21	-0.41	0.69
GLSl	0.08	-0.77	-0.73	-0.95	-0.5	0.42	0.18	-0.42	0.32

***GLS1"** refers to the iterative algorithm described below. Attempting to fix U. whose true value **is almost zero,** leads to very high error - **ten** times as much as fixing V or W. **Fixing** any of **the** translation components **leads** to a bias in the direction of **that** component. **Although the** V~fixed and W-fixcd solutions differ in certain respects note that they agree

well between themselves and with both the veridical situation and the minimum-error solution regarding certain key quantities:

1) The magnitude of W (given a mean depth of 1).

2) The direction of the vector {U V} (but not its magnitude)

3) The value of the rotation component C.

4) The values of the quantities (V - A) and (B + U) which show far less variation than their components.

From the analysis of section 2.7 these are precisely the quantities which we would expect to be most stable.

In LH1 we obtain an estimate of {U V W} directly from the flow field and obtain a further six quantities, any three of which serve to give an estimate of the rotation vector. In a narrow angle view we might be more inclined to trust the three quantities associated with first and zeroth order terms - (AW + CU), (BW + CV), (AU + BV) - and in a very wide angle view the three second-order associated terms. Rather obviously we might decide to "blend" all six terms in an LSR - with the refinement of relative weighting determined by angle of view if desired. This simple step can make a spectacular difference to recovery of the rotation vector. But without doubt the soundest method of estimating {A B C}, given an estimate of {U V W}, is to insert the estimated values of the latter terms in equation (2.3.1) and, after some reorganisation, carry out an LSR for A, B and C. Rewriting appropriately we have:

$$A(-xyV + y^2 U - xW + U) + B(x^2 V - xyU - yW + V)$$

$$+ C(x^2 W + y^2 W - xW - yV)$$

$$= v(U) + u(V) + (vx - uy)(W) \qquad (3.7.1)$$

A set of these should be scaled correctly - divided by J - so that we minimise the E-norm. This is the method employed to derive the results given above.

3.8. The narrow angle case

LHl, with rotation permitted, was tested on two groups of bus data:

- the collection of 50 point-motions from the bus (Dl)

- Dl augmented by 50 "background", null-motion vectors (D2)

Dl is "small object" data. The angle subtended by the bus, and the depth variation associated with the bus is small relative to the level of noise. D2 is "high depth variation" but relatively narrow angle data - the effective depth of the background points being infinite.

In the D2 case all methods which were applied (the Bruss and Horn method is not directly applicable where rotation is allowed) produced essentially the same results as in the translation-only case. This is because the static background effectively removes the possibility of any rotation and returns us to the translation-only case. An observation of possible interest is that it seems to take relatively little background data to eliminate the rotation components in the case of exhaustive search or my iterative algorithm whereas LHl seemed to need "swamping" with null-motion data points dispersed across the field of view before it would stop inferring any angular motion.

Where the data was Dl alone neither LHl nor my iterative algorithm GLSl did at all well so far as the wide-angle terms were concerned (the latter took more than 100 iterations to settle), though the "small object" terms W, *C,* (V - A) and (B + U) were relatively stable as anticipated. LHl produced one solution with an error lower than that discovered by my iterative algorithm GLSl but this turned out to imply 10 of the 50 points being behind the camera. (My algorithm explicitly computes depth and is not permitted to allow points negative depth).

SOI and SO2 (the small object algorithms described in section 2.9) were applied to Dl with quite gratifying results. These algorithms are very fast to apply and to "optimise" because:

1) They do not attempt to estimate all five parameters of motion (let alone the additional three fictional field parameters involved in LHl). They address themselves only to the "small object* quantities.

2) Only the one parameter family of directions of the vector {U V} needs to be explored to obtain the minimum error solution to a set of homogeneous equations of form (2.9.7).

Both algorithms involve a three stage process:

1) Estimation of the ^fixation* terms (V - A) tod (B + U)*

2) Solving for the parameters of equation (2.9.7) - effectively three in number.

3) Computing W and C from these.

They differ in that SO2 changes the coordinate system in stage 1 whereas SO1 does not. So the two following estimates of the fixation terms from the D1 data are not directly comparable:

	B + U	V - A
SO1:	0.41	-0.39
SO2:	0.82	-0.67

S02 switched the coordinate system to have its origin at the centroid of the data points: x = -0.10, y = -0.03. If we take the motion parameters:

$$U = 0.63 \quad V = -0.67 \quad W = 3.62,$$

from table 3:1 to be veridical (and A = B = C = 0) then it is clear that SO1 has underestimated both fixation terms. The "veridical" motion parameters become, in the coordinate system which SO2 is using:

$$U = 0.96 \quad V = -0.66 \quad W = 3.56$$

so its estimate of the fixation terms is fairly accurate.

One may think of the difference in the fixation terms arising because SO1 is "tracking" the origin, while SO2 is "tracking" the centroid of the data points which is to the left of the origin and further from the focus of expansion - so it is moving faster.

To solve a set of equations of form (2.9.7) we fix one component of the vector {U V}. This does not have to be in the direction of the X or Y axis - it is straightforward to transform the equations so that a component in any direction may be fixed. When the fixed component is roughly orthogonal to the true vector {U V} the results are appalling, as we would expect, but over a range of 30 degrees either side of the true motion they are stable. I used a measure of error which is equivalent of the E-norm for this case - the sum of prediction error squared when the parameters have been normalised so that the magnitude of {U V} is 1 - and searched over the direction of {U V} for the best solution in both cases. The fixation terms were converted back into the original coordinate system and the error "Te" (in degrees) in the estimated direction of {U V} calculated.

	W	C	(B + U)	(V - A)	Te
SO1:	4.17	-1.06	0.41	-0.39	21
SO2:	4.22	-1.08	0.82	-0.67	30

Both algorithms infer an anticlockwise rotational component about the optic axis (which my iterative algorithm also insisted was present). The estimated values of W are gratifying in the light of the noisiness of the data.

Further investigation on real data is indicated but my first impression is that these algorithms show considerable promise.

3.9 Iterating and searching over the translational motion

We may seek to obtain optimum recovery of all motion parameters by a wide variety of iterative procedures designed to bring us to a least-squares interpretation of the flow-field which is consistent with the assumption of rigid motion.

There is a very large number of possible ways of setting about such a constrained regression and I have implemented many of them with sometimes satisfactory results. However, note that the theoretical justification for equation (2.3.1) is that it allows a "one-shot" solution of the problem. If we are going to employ numerical solution techniques then it is not necessarily the best policy to apply them in the context of LH1. An algorithm which iteratively derives motion parameters (and point depths) from the original field equations (2.1.1) is discussed below. This algorithm, in my experience, produces the most consistent results.

With all constrained regressions there is the danger that sufficient degradation of the data may lead to a situation in which iteration does not lead to the least-squares solution. In such a case, however, it is generally true that the initial estimate is very wide of the mark. In many situations in which narrow-angle imagery is to be interpreted (e.g. where it is reconnaissance imagery) a considerable amount is known about the motion a priori. This could be used to prevent a "wild" initial estimate.

The optimum solution to the (LH1) case can be guaranteed by a sufficiently fine-grained search over the two-parameter family of translational motion directions. Given {U V W} - that is to say given the focus of expansion - a least squares regression for {A B C} over a suitably weighted set of equations of form (3.7.1) will yield the best estimate.

3.10 Explicit depth computation - the general case

In algorithm which I call GLS1 we seek an iterative solution to a set of equations of the form (2.2.1) as follows:

1) Some initial estimate of motion parameters is made

2) z (for each point) is explicitly estimated on the basis of these parameters and the two equations for the point

3) New motion parameters are estimated on the basis of these z estimates

4) Return to step 2 unless no significant change

(Alternatively we may start with an estimate of the depth field)

Equations (2.2.1) are:

$$u = (Wx - U)z - B(1 + x^2) + Cy + Axy \qquad (2.2.1a)$$

$$v = (Wy - V)z + A(1 + y^2) - Cx - Bxy \qquad (2.2.1b)$$

Introducing shorthand expressions Ω_1. Ω_2. T_1. T_2 for the rotational and translational parts of these we have (for each data point) two conditions to satisfy:

$$u = T_1 z + \Omega_1 + \epsilon_1 \qquad (3.10.1a)$$
$$v = T_2 z + \Omega_2 + \epsilon_2 \qquad (3.10.1b)$$

where u and v are the two components of the full vector and ϵ_1 and ϵ_2 represent error in the data and/or error in the current estimates of T_1. T_2. Ω_1. Ω_2. u and v are observables and *given* any set of global motion parameters we may calculate T_1. T_2. Ω_1. Ω_2. If the parameters are correct, and the image is noise-free, then (3.10.1a) and (3.10.1b) will yield the same value for z. If the parameters are incorrect then it is unlikely that (3.10.1a) and (3.10.1b) will yield the same value of z - in which case we should adopt the least squares estimate of z which is:

$$(T_1(v_1 - \Omega_1) + T_2(v_2 - \Omega_2))/(T_1^2 + T_2^2) \qquad (3.10.2)$$

This is likely to give "wild" values for z only in the region of the currently estimated focus of expansion (where $T_1 = 0$ and $T_2 = 0$) for the obvious reason that we are close to "dividing zero by zero". This will not destabilise the global process. On the contrary it may tend to do the opposite (slow it down unduly) for the following reason:

The depth of a point at the f.o.e. has no effect upon the flow field but the *estimated* position of the f.o.e should be shifting as we iterate. A point may acquire

a ridiculous depth value when coincident with the estimated f.o.e. which, fed back into the global process, prevents the f.o.e from moving. If a point has an astronomical inverse depth - depth near zero - then the only position of the f.o.e. consistent with this and the data is precisely at that point. This can be guarded against by simply adding a stabilising term to the divisor of (3.10.2) (which may be reduced to a very small value as we approach the correct solution). This defaults a point at the currently estimated f.o.e to infinity ($z = 0$).

We apply the visibility constraint in the most natural way: if the estimated inverse depth of a point, calculated from (3.10.2), is negative then we set it to zero.

Once we have a set of estimates for z we can substitute these back into (2.2.1a) and (2.2.1b) and carry out a simple regression for U, V, W, A, B, C.

With good data this algorithm is strongly convergent provided the initial estimate of motion parameters is not too far wrong. Where it "gets off on the wrong foot" it may converge to a spurious solution which might be related to the dual solutions which exist in in degenerate situations (see below).

This algorithm has two advantages over treatments based on algebraic elimination of depth:

a) it can handle the "degenerate" cases in which the quadric condition is fulfilled.

b) it requires only five data points (each with two vector components associated) to produce a solution

It has the disadvantage that - even in the case of perfect data - it may not converge to the right solution. We can predict this theoretically and demonstrated it empirically as follows:

In the case that the object being viewed is planar and all points lie to one side of the "phony horizon" we know that we have two possible solutions - involving a "switch" between the rectilinear motion vector and the normal to he plane. Suppose we set up such a planar case. My algorithm will converge to one or the other solution (depending upon starting estimates of the motion parameters). Suppose we pick the solution to which it does *not* converge as "veridical" and very slightly disturb, at random, the depths of points on the plane. An errorless dual solution now does not exist since the object being viewed is not planar. But since we have disturbed point depths by an "infinitesimal" amount my algorithm will still converge to the phony, errorful dual.

A similar argument can be applied to the ambiguous situations analysed by Maybank (1985b) which arise for particular configurations of points *in conjunction* with particular motion parameters ("structure-motion coincidences" in the terminology of Waxman and

Ullman 1984). An interesting theoretical and practical question is how many false solutions GLS1 might converge to in the the case of any one image. I suspect (though I emphasise it is only a suspicion and perhaps an optimistic one at that) that local optima, when they exist, are indeed related to the planar and Maybank ambiguities - and that they therefore cannot number more than three (or six if we take into account a reversal of the sign of the translational motion). But I have not yet given the matter the close attention which it deserves.

For the synthetic case shown in Table 3:2 the estimated parameters and E-norm at each stage are shown below. The starting assumption was that z = 1 everywhere:

TABLE 3:3

Iteration number	U	V	W	A	B	C	E-norm
1	0.3	-0.68	-0.67	-0.97	-0.74	0.44	0.65
2	0.13	-0.74	-0.66	-1.0	-0.57	0.44	0.36
3	0.09	-0.74	-0.67	-1.0	-0.53	0.44	0.33
4	0.08	-0.74	-0.67	-0.99	-0.51	0.43	0.33
5	0.08	-0.74	-0.67	-0.98	-0.51	0.43	0.33
6	0.08	-0.73	-0.68	-0.97	-0.5	0.43	0.33
7	0.08	-0.73	-0.68	-0.96	-0.5	0.43	0.32
8	0.08	-0.73	-0.68	-0.95	-0.5	0.43	0.32
9	0.08	-0.73	-0.68	-0.95	-0.5	0.42	0.32
10	0.07	-0.73	-0.68	-0.95	-0.5	0.42	0.32

(The translational motion vector is here normalised to a magnitude of 1. When it is renormalised to give a best fit to the hypothesis than z=1 for al points it acquires the slightly different values shown for GLS1 in Table 3:2)

3.11 Explicit depth computation - the planar case

Recalling the analysis of the aperture problem in Chapter 2 we may write, in the planar, edge-motion case (for an edge oriented at an angle θ to the x-axis):

$$v_1 = (W(cx+sy) - Uc - Vs)(Lx + My + N)$$

$$- B(c(1 + x)^2 - sxy) + C(cy - sx) + A(cxy + s(1 + y)^2) \quad (3.11.1)$$

where $s = \sin(\theta)$ and $c = \cos(\theta)$.

Rather than multiply out this equation to obtain the field equation (2.8.4) we may try to solve a set of such equations by an iterative procedure such as:

1) Given an initial estimate of {L M N} solve, by simple least squares, the resulting linear equations for {U V W} and {A B C}.

2) Substitute these values into (3.11.1) and solve the resulting linear equations by simple least squares for {L M N}.

3) Return to step 1 (unless no significant change).

(we can start with velocity estimates if we have them)

This algorithm is used by Negahjdaripour and Horn (1985) and I will call it N&H.

At the very least we know a priori that z is positive (the visibility constraint). This makes it the obvious term to fix initially. In the absence of any other bias or expectation we set $z = 1$ everywhere (i.e. $L = 0$, $M = 0$, $N = 1$) and proceed with our alternating process.

We have the immediate advantages over LH2 that:

- any expectations which we have (over and above the certainty that depth is positive) can be incorporated from the outset and

- if the solution moves off into "invisibility" we simply drag the offending point or points back into visibility (by not allowing z to be negative).

I have experimented with my own version this algorithm, which I developed independently. Some points regarding Negahjdaripour and Horn's version and their account of its performance:

1) They state that the process always converges - though it may sometimes take hundreds of iterations to do so. I find this hard to substantiate - I have encountered examples in which 200 iterations yield a state which is not obviously converging to any of the possible solutions. It is of course impossible to prove empirically that the process would not, given sufficient time and sufficient decimal point acuity, eventually arrive at a solution. Perhaps the best statement of the facts would be that for practical purposes the algorithm is not always convergent. (One reason why my version does not always converge is that it may be attempting to settle on a spurious

solution which involves invisible points. In such a case the application of the visibility constraint prevents it from arriving. My comments above, however, relate to implementations in which this constraint was not enforced.)

2) They "unpack" the regression computations in order to take advantage of the fact that many of the terms involved at each iteration are unchanged. This is a practical implementation detail with which I have not concerned myself.

3) They note that it is possible to solve alternately for U, V, W, A, B, C and L, M, N, A, B, C (solving for the rotation parameters at *each* step rather than every alternate step). They state that this procedure converges more rapidly than the UVWABC/LMN alternation. In my experience this algorithm is liable to converge to a spurious bi-stable condition in which there are two sets of A, B, C parameters - one associated with the UVWABC step and the other with the LMNABC step. They have not stated what measures they take to prevent this happening.

4) They do not reweight their data in the light of up-to-date estimates of the full vectors at each point or implement any regularisation measures such as I have discussed.

To summarise my experience of this algorithm briefly:

Where the initial normal vector {L M N} is within about 60 degrees of its true value and the angle between the plane vector and the translation vector {U V W} is substantial the procedure almost always converges strongly (though perhaps to an alternative solution in which the normal and translation vectors are interchanged). We can be very confident of good convergence on at least *one* process if we set up five processes with initial plane vectors {1 0 0}, {0 1 0}, {-1 0 0}, {0 -1 0}, {0 0 1} - a total ignorance "shotgun" strategy. The final stable states appear to be *identical* to those yielded by LH1 with equivalent error-handling measures. This is true even with very noisy data.

Where the angle between {L M N} and {U V W} is small convergence can be painfully slow (hundreds of iterations) even with perfect data and from a near-correct starting position.

Rate of convergence is (as we would expect) a function of noise and conditioning of the equations. In wide-angle, accurate images half a dozen cycles may suffice to achieve a good approximation to the best answer - whereas we may have to iterate tens of times to achieve a solution within 10 percent of the "correct" (LH1) solution with a narrow-angle, noisy image. But this faithfully reflects the fact that one solution (in the converging sequence) is about as good as any other in these circumstances.

Note that there is no theoretical reason why we should not adapt this algorithm to allow a second or higher order inverse depth field - in which case we would have an iterative equivalent of Waxman's algorithm.

An obvious development of N&H – given its "exposed workings" – is to cross-constrain the flow-field interpretations provided by it at different moments in a flow sequence. The simplest assumption one can make about the parameters in the context of a flow sequence is that the motion vectors remain unchanged throughout. The orientation and depth of the plane relative to the camera (i.e. the plane vector) must change in a manner consistent with the motion. It is relatively easy to enforce – or regularise on the basis of – such assumptions in a computation in which the scene parameters appear explicitly.

3.12 Exposed workings and the intrinsic image paradigm

Another attraction of an algorithm in which structural and motion parameters are exposed is that it enables us to enforce the rigidity constraint across orientation or occlusion boundaries.

As Barrow and Tenenbaum (1978) pointed out boundaries in an image usually mark discontinuities only in certain intrinsic image qualities – they are seldom discontinuities in every intrinsic quality. A natural image is not an arbitrarily assembled montage of independent mini-images. In the optic flow image cast by a rigid assembly of planes there will be image discontinuities in the surface normal vector but not in the motion vector. However each of the flow field parameters $P_1 ... P_8$ in LH2 contains a surface normal component, so all of these will tend to be discontinuous at structural boundaries.

The problem of *how* to segment is addressed in Chapter 5. Here, let us assume we are given the segmentation of a multi-facetted picture and wish to recover the (global) motion and (regional) plane parameters for each region. If we use LH2 we are forced to treat each region as an independent image. If we use N&H, by contrast, we only need to do this so far as the planar normal is concerned. The fact that the polyhedron is rigid means that we can still treat the image as a single entity when estimating motion parameters.

If there are N different planes visible then interpretation by LH2 involves estimation of 8N notionally independent parameters. Using N&H we require to estimate only 5 + 3N. Further, if we know the planes are facets of a polyhedron then we have further constraints (the continuity everywhere of the depth field) which effectively reduce the number to 7 + N or fewer.

The benefits to be gained from decoupling estimation of surface from motion parameters in the case of multiple planes moving rigidly may be readily demonstrated with synthetic data.

An edge-motion image was generated as follows:

Global motion parameters:

U - 0.27 V - 0.89 W - 0-36 A - -0.52 B - 0.35 C - -0.78

Plane parameters:

1st quadrant	L *	-0.49	M -	-0.12	N -	0.86
2nd quadrant	L -	-0.49	M -	0.46	H -	0.86
3rd quadrant	L -	0.3	M -	-0.12	N -	0.86
4th. quadrant	L -	0.3	M -	0.46	N -	0.86

(this is a four-sided pyramid with the apex aligned with the optic axis the 1st quadrant is that with x > 0 and y > 0 and the others numbered anticlockwise).

10 points were randomly placed in each quadrant and 5 randomly placed on each line of intersection (the four "arms[11] of the coordinate system). Edge orientations for each point were chosen at random and an edge-motion calculated. Additive Gaussian noise of c. 15 percent was introduced. Three methods were used to recover estimates of the motion and planar normal parameters.

Error in recovery was assessed by three measures:

ER1: The angle between the true and estimated direction of translation

ER2: The angle between the true and estimated planar normal

ER3: The ratio of estimated to true magnitude of the cross-product of translatory and planar normal vectors. (If translation velocity is normalised this represents the error in estimated perpendicular distance to the plane).

Method 1: Apply LH2 to each set of 20 coplanar points.

In the cases of quadrants 1 and 2 there was no doubt about which of the four algebraic solutions offered by LH2 was best. In the cases of quadrants 3 and 4 however the most accurate solutions (in terms of ER1. ER2 and ER3) involved 40 percent and 25 percent poiats with negative depths. I have given the error measures for both the "invisible[11] and the only 'Visible* solution for quadrants 3 and 4:

	ERl (degrees)	ER2 (degrees)	ER3
Quadrant 1:	51	45	2.23
Quadrant 2:	23	28	1.41
Quadrant 3: invisible	52	41	0.27
visible	85	58	2.05
Quadrant 4: invisible	52	30	0.67
visible	147	90	0.68

Method 2: Apply N&H with *global* estimate of motion parameters but separate estimates of planar normal for each coplanar group of points.

To all intents and purposes the procedure had converged by the tenth iteration but was continued until the 25th.

	ERl (degrees)	ER2 (degrees)	ER3
Quadrant 1:	11	20	1.53
Quadrant 2:	"	8	1.01
Quadrant 3:	"	29	1.24
Quadrant 4:	"	13	0.94

(there was only one point, in quadrant 1, which had negative depth).

Method 3: Apply an N&H variant in which we use our knowledge that the axes are projections of lines of intersection of the planar facets- This effectively reduces the number of degrees of freedom in the planar normals from 12 to 5. The 5 are computed simultaneously. Effectively convergent after 5 iterations, continued for 10.

	ERl (degrees)	ER2 (degrees)	ER3
Quadrant 1:	7	15	1.17
Quadrant 2*:	"	12	1.13
'Quadrant 3•	"	12	1.0
Quadrant 4:	"	5	0.99

Hie potential advantages of integrating regional and global processes seem very obvious from these figures. The level of noise is not excessive, nor is the viewing angle very ntrrow* So far as the key directions are concerned the LH2 results are no better than random (recalling that we had four putative solutions to choose from for each facet) and the ranging is very poor. Both Iterative procedures produce far better results. Method 3 could cleaxly withstand much more noise before generating results as poor as method 1.

Method 2 invokes only the global rigidity assumption in addition to the implied assumptions of method 1, whereas method 3 explicitly draws on the strong assumption of

depth continuity across known boundaries.

To *some* extent it is possible express such knowledge as constraints upon the flow field. For example the fact that certain points belong in two different flow fields implies that the full motion vector imputed to such a point by both sets of flow-field parameters should be the same. Also, we can use knowledge of known intersection boundaries to place limitations upon the manner in which flow field parameters may change across a boundary. For example since we know that all four planes intersect at the origin we know that N does not change across across any boundary. Since we know that the x-axis is the projection of a line of intersection of two planes we know that only the planar component L may change as we cross the x-axis. This means that only those flow-field parameters involving L (P_3, P_4 and P_7) may change. However this still allows three degrees of freedom where there is truly only one. Moreover it is difficult to enforce such a constraint except by an iterative procedure.

Note that there is a "dual" interpretation of the image of the form: a single plane undergoing deformation (the single translation vector and the four planar normals are interchanged). This corresponds to a global assumption of planarity combined with the assumption of rigidity within regions.

3.13 Polyhedra, smooth surfaces, hinged motion and rubbery motion

If motion is non-rigid, but we know it is hinged at certain points, then we can take advantage of our knowledge only if we have "exposed workings". If we have two groups of points G_1 and G_2 each of which constitutes a rigid structure and they have common members - hinge points - H then the cross-constraint between the inferred motions of G_1 and G_2 is that they should both predict the same depth and the same 3-D motion for each point in H.

Hinged motion may be regarded as a discrete approximation to rubbery motion much as a polyhedron may be regarded as the discrete approximation to a smoothly curved surface.

	Surface	Motion
Discrete	Polyhedron	Hinged motion
Continuous	Smooth surface	Rubbery motion

We are naturally prompted to a study both of smoothing and of segmentation in which we position ourselves to obtain a wide-angle view of both topics at once.

CHAPTER 4

LOCAL PROCESSING OF VELOCITY FIELDS

4.1 Introduction

In this chapter I look at locally-based computation, starting with well-known algorithms which "smooth" the optic flow. The idea that an image should be be made to yield an interpretation which is optimally smooth, in some sense, may be regarded as a specialisation of the intuitive notion that: the world is no more complex or "energetic" than it appears to be. This principle continues to lack a robust, generally-accepted expression although we could loosely subsume it under more general principles - such as the principle of Pragnanz (Koffka 1935) of Economy (Attneave 1972) of General Viewpoint (Huffman 1971). An interesting attempt at formalising it and avoiding reference to principles such as these which have a slightly mystical flavour is Grimson's (1982) "no news is good news" dictum. This makes the roundabout statement that: if the world were less smooth than it appears to be, it would not appear to be so smooth. His statement has the merit of being very specifically argued in terms of a model of the relationship between surface shape and the formation of edges in the image.

Smoothness is a notion susceptible of an extremely wide number of definitions. In its weakest form it implies little more than a correlation, at some scale, between different qualities. Thus a function (say a depth map) need not be continuous - even at zeroth order - for it to be smooth to some degree, by some definition. The disparity gradient limit (Pollard et al 1985) is an example of a smoothness measure applied to discrete points. The smoothness of the disparity field is quantified by the number of conformities to the d.g. limit. (Though the 3-space interpretation of the d.g. limit tends to be in terms of a continuous Lipschitz surface). At the opposite extreme are measures based upon differential quantities - such as the familiar integral of gradient-squared over some range of a function.

In the case of our being given only edge motion we cannot derive both the global motion parameters and a depth map without some form of additional constraint (Chapter 2.8) and the idea that we seek to extremise smoothness, by some definition, has an obvious appeal.

In this chapter I explore two applications of the idea of smoothing by local computation (out of a vast number of possibilities) to optic flow interpretation. These are, briefly:

1) Recovery of a full-vector field by "relaxation" after the style of Horn and Schunck or Hildreth. In this approach global interpretation, by an algorithm such as LH1, is strictly posterior to the computation of a smooth flow field.

2) Finding the motion parameters *such that* the inferred inverse depth field is

maximally smooth by some definition. Here global interpretation and locally-based smoothing proceed in parallel.

I end this chapter with a discussion of the relation between local and global processes.

4.2 The Horn/Schunck and Hildreth smoothing algorithms

Smoothing procedures, being locally based in image space, appear to offer opportunities both for locating flow boundaries and for respecting structural boundaries which have been identified by other processes. The need to locate discontinuities in flow was claimed as a prime motivating factor in Horn and Schunck's (1981) paper describing their algorithm. Although they did not in fact incorporate a segmentation process, others have since done so (Schunck 1984; Cornelius and Kanade 1983). Hildreth's algorithm (Hildreth 1982, 1984a, 1984b) not only takes note of structural boundaries but tends to propagate estimates of velocity along them. Her procedure is designed to work on closed curves, such as zero-crossing contours.

In their constrained optimisation form both the Horn/Schunck and Hildreth algorithms (the "MIT algorithms") aim to generate an estimate of the flow-field which displays minimal departure from smoothness while conforming to the data. The measure of smoothness in both cases is the integral of gradient-squared of the velocity field. In the Horn/Schunck case integration is over the area of the image or region being smoothed, whereas in Hildreth's case it is along the length of a contour. The use of gradient, and of a quadratic norm, to quantify smoothness has nice mathematical implications. Uniqueness of the solution can be demonstrated and the solution can be achieved through strictly local, iterative procedures which are computationally very simple. See for example the discussion of the Gauss-Seidel method in (Horn and Schunck 1981) and Hildreth's (1984) discussion of alternative smoothness measures.

The authors of both algorithms make allowance for error in the data by expanding the cost function to be the weighted sum of the smoothness norm and a quadratic error norm (the familiar sum of squares of departure from the data). The relative weighting of these terms notionally reflects our confidence in the data relative to the strength of our faith in the idea that the world is a place which generates boring optic flow fields. The note of irony is intended to remind readers that "smoothness" is a principle which continues to lack a sound theoretical foundation, at least in the domain of optic flow, and that principled means of fixing the relative weighting of error and smoothness norms have not been developed.

For brevity I will outline only Horn and Schunck's algorithm. Hildreth's is logically similar.

The state of the vector I (= {u v}) at each pixel is updated at the Nth iteration as follows:

1) The motion constraint equation associated with the pixel, if there is one, gives the "opinion of the data".

$$m = au + bv \tag{4.2.1}$$

2) "The opinion of the neighbours" is that:

$$u = \overline{u} \quad \text{and} \quad v = \overline{v} \tag{4.2.2}$$

where $\overline{u}$ and $\overline{v}$ are the (suitably weighted) average values of u and v in the 8-connected neighbours after the previous iteration, the (N-1)th.

3) We have three linear equations and two unknowns so we compute the least-squares values of u and v, after deciding how much weight to award the data constraint (4.2.1) relative to the smoothness constraints (4.2.2). The computed values of u and v are those which minimise:

$$(1 - \sigma)^2 (m - au - bv)^2 + \sigma^2 ((u - \overline{u})^2 + (v - \overline{v})^2) \tag{4.2.3}$$

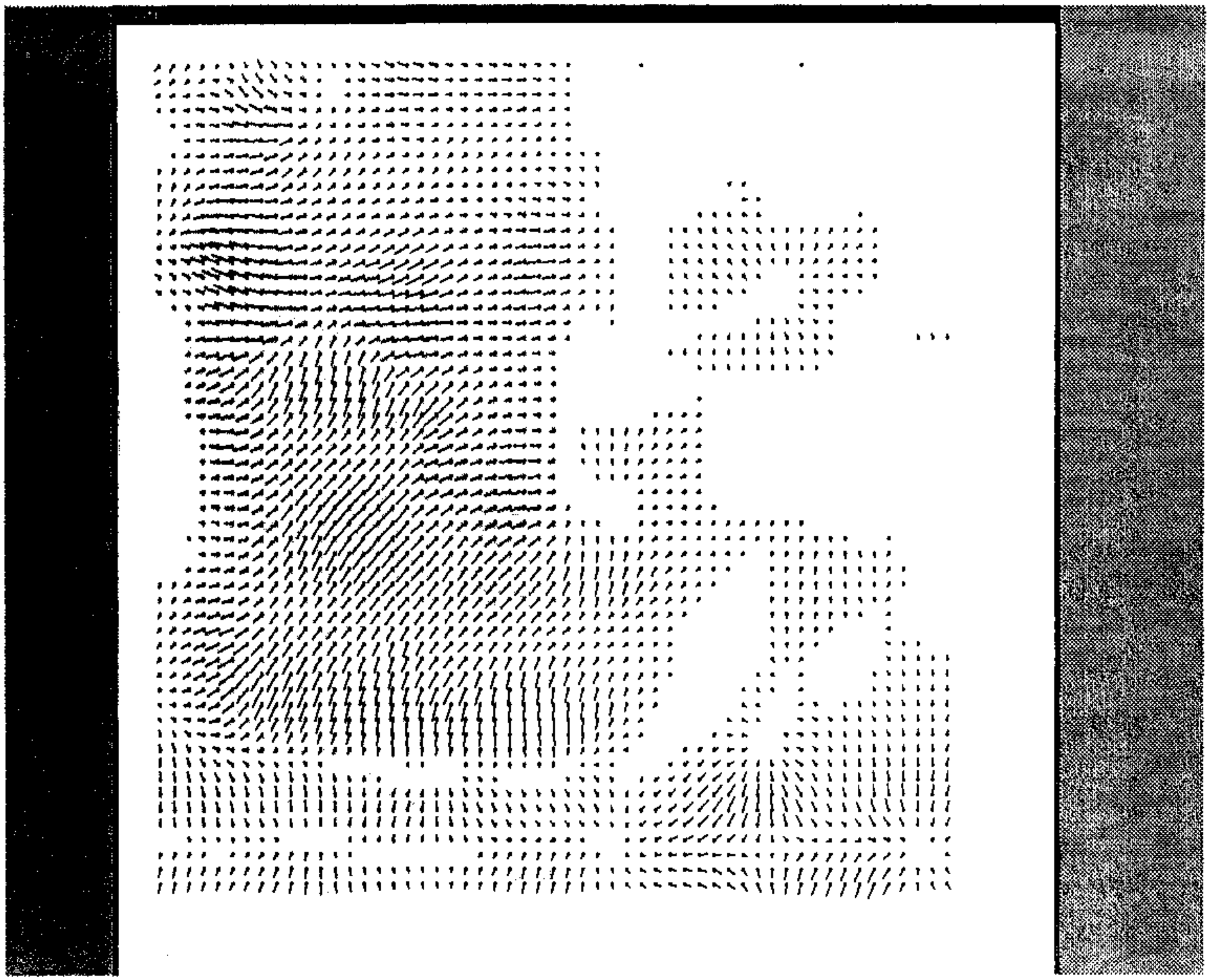
The parameter σ $(0 < \sigma < 1)$ is the "smoothing pressure". If there is no data at the pixel (4.2.3) will be zero since the two conditions (4.2.2) are satisfied exactly. The first term in (4.2.2) refers to inferred error in the data, the second to the (finite element estimate of) the sum of squares of the Laplacians of the flow-field at the pixel in question. If σ is very small then the recovered field will adhere slavishly to the data - the problem tending toward the condition of a constrained optimisation. If σ is made large then the iterative process will tend to flatten out anomalies in the data - including such anomalies as *bona fide* flow boundaries!

Theoretically the initial state of estimated motion in each pixel is irrelevant. For practical purposes it is simplest to set this to {0 0} everywhere.

At the boundaries of the image or of the region being smoothed a pixel lacks outside neighbours and its state is computed by reference only to the average opinion of such neighbours as it does have. This causes a "flattening" effect which I will discuss in the last section of this chapter.

The magnitude of the smoothing pressure σ is only meaningful taken in conjunction with the "intrinsic reliability" of the data. In the Horn and Schunck case the constraint equation implicitly carries a weight with it, as discussed in Chapter 1.

To modify each of our motion constraints, obtained from the principal axes procedure, to the Horn/Schunck "format" we multiply them by the explicit confidence measure which we associated with them at the time of their derivation. We have *two* constraint equations for each pixel so both of these are included, together with the neighbourhood constraints (4.2.2), in the least-squares regression at each pixel.



[illegible] field after 5 [illegible] of Horn/Schun[illegible] [illegible]thing

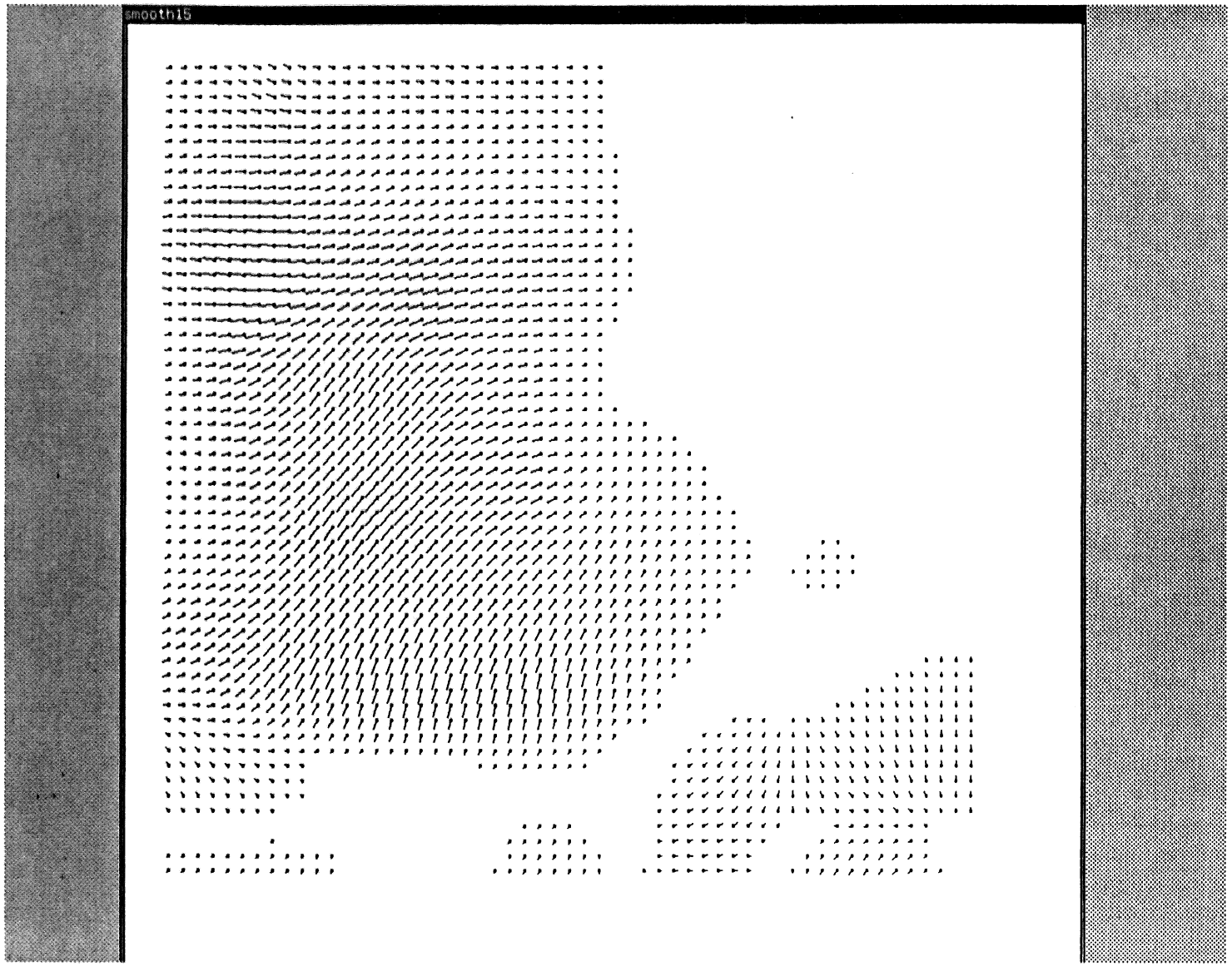


Figure 4:2 64x64 field after 15 applications of Horn/Schunck smoothing

Figures 4.1 and 4.2 show the effects of processing the London Bus 64×64 velocity field with Horn and Schunck's algorithm, slightly modified as above. The vectors displayed are the current estimates, after 5 and 15 iterations respectively, of the full velocity at each pixel. As with earlier figures vectors with a magnitude of less than 0.3 of a pixel are not shown. The smoothing pressure parameter, σ, was chosen so that, on average over all pixels, "the opinion of the data" carried comparable weight to "the opinion of the neighbours" . We can immediately observe two characteristic qualities of the process which will come to concern us later:

1) While the field is very quickly brought to a "fluent" state at a local scale there is little sign of the more "global" order which we know obtains. The vectors in the *veridical* field are pointing towards the top right hand corner - but they are a long way off doing this at the 15th iteration.

2) Long before the field has been brought to order it is leaking into the road viz. the communication between pixels on either side of the flow boundary between the region subtended by the bus and the background has led to background pixels becoming "excited" and bus pixels becoming "depressed".

(The flow field after the 15th iteration is close to the state to which it converges for this particular value of σ. Further iterations make little difference to its appearance. If we wish for more accurate alignment of the vectors overall we must increase σ - with concomittant "leakage" as a consequence).

However we can see that mild smoothing (say after 5 iterations) seems to make more sense, from the point of view of motion, than the original unsmoothed image (figure 4.1). Motion is locally more coherent and isolated patches which appeared to be in motion in the raw image have faded into the background. The band of video noise at the bottom has not been completely suppressed - but then it is a very marked feature, in apparent coherent motion, to the human eye. The reader should be careful not to be misled by the loss of *structural* information from the raw image to the smoothed images. Our "principal axes procedure", like the Horn/Schunck motion constraint computation, is directionally selective and acts in some respects as an edge-detector. It is this property, spurious from the purely motion point of view, which makes it possible to "see" much of the structure of the bus in the raw velocity image.

But a smoothing process cannot be an end in itself - it is an intermediate step on the way to 3-D interpretation and beyond. However pretty the enhanced estimates of flow may be we would like to know under what circumstances they will be a good approximation to the truth. Variational recovery of image-motion would be hard to justify if it did not tend to yield results consistent with those obtained through analytical methods, at least in certain simple circumstances.

Rather trivially, both the MIT algorithms will yield the veridical field where it is uniform (zeroth-order) and the data is free of error. But this situation only obtains, under

orthographic projection, in the absence of any rotational motion component. Under perspective projection it only obtains when the camera is, in addition, translating parallel to a fronto-parallel plane.

Subject to reservations concerning boundary conditions, Horn and Schunck's algorithm will yield the correct flow-field where this is first-order in image coordinates and the data are free of error. When the Laplacians (of the two velocity components) are everywhere zero the state of the field is stationary - both in the Calculus of Variations sense and computationally. Regardless of the magnitude of the gradient a local computation, at a pixel not on the boundary, causes no change of state since there is no "curvature" in either component of the velocity field. At the boundary, however, a pixel lacks outside neighbours to offset the "pull" of its inside neighbours - the field thus tends to deteriorate to a flat, zeroth-order condition at the margins unless there is much strong data to effectively fix boundary conditions.

In Hildreth's case the situation is more complex since anisotropic smoothing across the image plane is replaced, in her algorithm, with smoothing along a curve in the image. Even if velocity is first-order in image coordinates it will not vary linearly with distance along an arbitrary curve. In the particular case of a polygon it can be argued that a first-order field will be correctly recovered in noise-free, constrained optimisation, conditions because:

a) the full motion is determined at the corners since there are effectively two differently-oriented edge motions at a corner and

b) the interpolation between corners will be correct since the field is first-order in distance along a straight line.

Yuille (1984) has formally demonstrated this and shown that for a general curved contour the *recovery* will not be exactly correct, even with perfect data applied as an absolute constraint. It should also be noted that, where the data is given only finite weight, Hildreth's flow-field will *suffer* from zeroth-order erosion, even in the case of a polygon. The comers are the "boundaries* in her case.

But Hildreth (1984) argues plausibly that the field recovered by her method will be "qualitatively similar* to the veridical field and also adduces some evidence that the human eye misinterprets - if that is right word - the motion of certain closed curves in a manner similar to her algorithm.

We might conclude that, on balance, gradient-squared smoothing seems to be justified where we anticipate a lixst-oxder flow field. However, to the extent that the veridical field has second or higher components, it is certain that both the *MST* algorithms will recover a distorted version of it. Undo: orthographic projection t first- or zeroth-order field results from any rigid motion of a planar surface. But under perspective projection such scene and motion conditions generally give rise to a second-order field. However, I will show that there is * wty of conjuring with the geometry of the perspective, planar* case - by defining

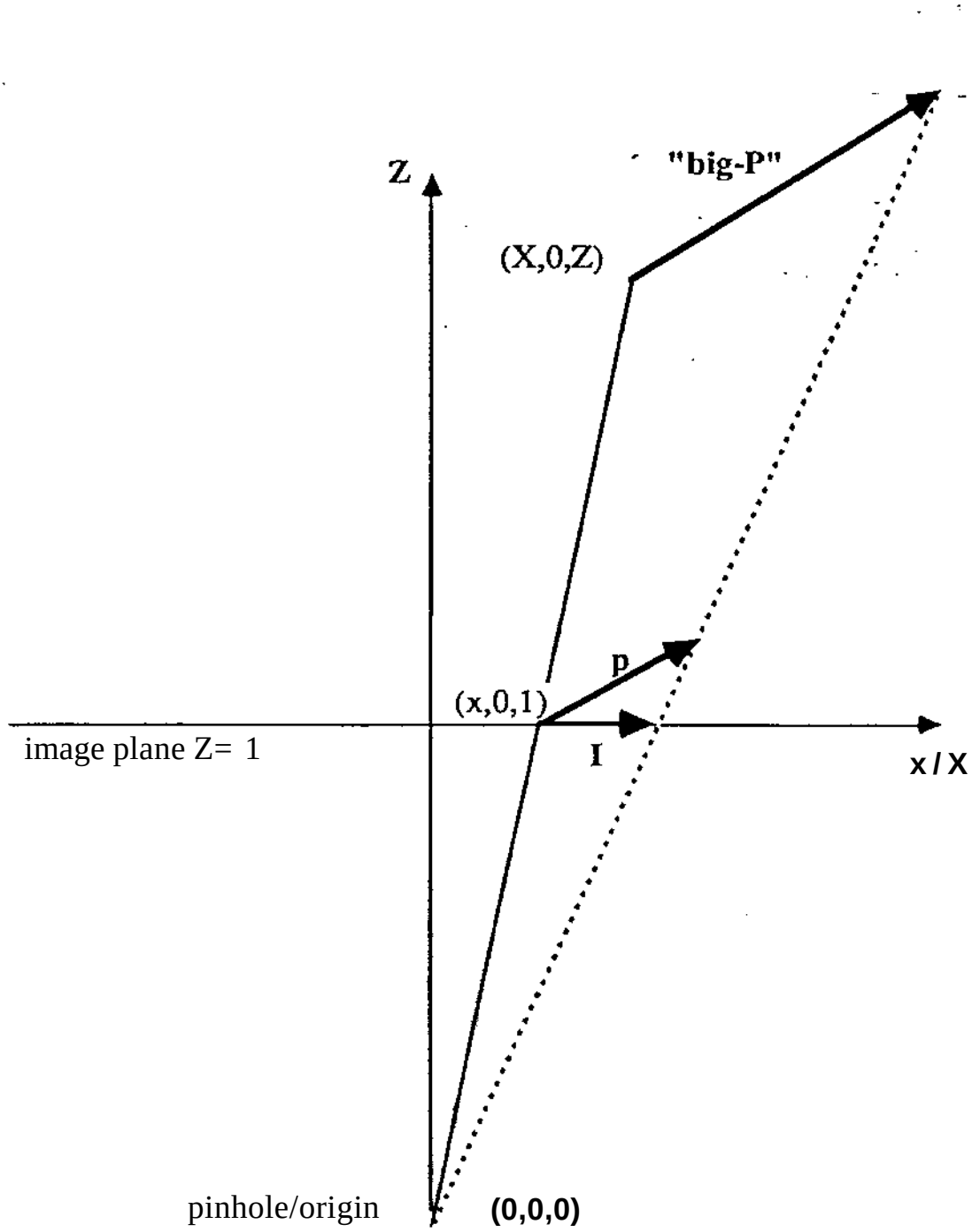


Figure *A3* Relationship **between. 3-D** motion, the vector pand the vector I (section through the plane Y«0)

the true image-motion vector as a 3-vector - which removes the second-order components. It is straightforward to generalise Horn and Schunck's or Hildreth's method to recover this vector.

4.3 The geometry of the vector **p**

We continue to use the same coordinate system as in Chapter 2. The 3-space relative motion of a point ("**big-P**") is:

$$\begin{aligned} dX/dt &= -U - BZ + CY \\ dY/dt &= -V - CX + AZ \\ dZ/dt &= -W - AY + BX \end{aligned} \qquad (4.3.1)$$

Longuet-Higgins' original derivation of equations (2.1) is different from the derivation which follows (which is also due to him):

If we divide (4.1) through by Z we obtain a motion vector "**p**", with components p_1, p_2, p_3:

$$\begin{aligned} p_1 &= -U/Z - B + Cy \\ p_2 &= -V/Z - Cx + A \\ p_3 &= -W/Z - Ay + Bx \end{aligned} \qquad (4.3.2)$$

We may visualise this vector as located in the image plane, at {x y}, but as having a component, p_3, out of the image plane. It is parallel to the true motion of the point but scaled by inverse depth.

The vector **p** is related to the vector **I** which we actually see in the image thus: **I** is the "shadow" cast by **p** on the image plane. It is the projection of **p** onto the image plane along the sight-ray. (figure 4:3). Some elementary trigonometry reveals that **p** and **I** are related thus:

$$u = p_1 - xp_3 \qquad (4.3.3a)$$
$$v = p_2 - yp_3 \qquad (4.3.3b)$$

If we substitute (4.3.2) into (4.3.3) we immediately obtain equations (2.2.1).

To investigate the properties of **p** when the viewed surface is planar we make the substitution (Lx + My + N) for 1/Z in equations (4.3.2). Gathering terms:

$$\begin{aligned} p_1 &= -(UN+B) - ULx - (UM-C)y \\ p_2 &= -(VN-A) - (VL+C)x - VMy \\ p_3 &= -WN - (WL-B)x - (WM+A)y \end{aligned} \qquad (4.3.4)$$

Note that all three components are first-order in image coordinates. If we substitute (4.3.4) into (4.3.3) we obtain, after a little reorganising, equations (2.5.3).

Observation of **I** does not determine **p** but constrains it according to (4.3.2) to lie on a line in 3-D velocity space. In the case where we have only one component of motion we can concatenate (4.3.2) and the motion constraint equation to obtain a "constraint plane" for **p**:

$$m = ap_1 + bp_2 - (ax+by)p_3 \qquad (4.3.5)$$

or

$$m = \cos(\theta)p_1 + \sin(\theta)p_2 - (x\cos(\theta) + y\sin(y))p_3 \qquad (4.3.5b)$$

In both the MIT algorithms the image vector **I** is recovered by smoothing subject to the influence of data-constraints of form (4.3.3). But we can recover estimates of the vector **p**, subject to constraints of form (4.3.5), by an analogous process which involves no extension of the logic of these algorithms.

4.4 An intuitive interpretation of the vector **p**

In an edge-motion velocity image we are given only one component of motion at a point. Smoothing approaches provide a means of "hallucinating" the missing component by cross-constraining the states of neighbouring pixels. We thus obtain a 2-vector, **I**, which is the projection of that component of real 3-D motion which is orthogonal to the sight-ray. But while we are in the hallucination business why not exert ourselves to estimate a component along the sight-ray as well?

Edge- or point-motion information in the velocity image does not tell us directly what **p** is but places a constraint on it analogous to the constraint placed upon point-motion by edge-motion data. Recovering **p** from image point-motion is an "aperture problem" homologous to the problem of recovering image point-motion from edge-motion.

To solve the traditional aperture problem it is necessary that the constraint lines (in velocity space) are not all parallel - for in that case there is no intersection and thus no cross- constraint. Similarly, to recover **p** from estimates of image motion, it is necessary that the sight-rays (which determine the constraint lines or planes appropriate to **p**) are not all parallel. So we could not hope to recover the vector **p** in the case of orthographic projection. But in perspective projection the sight rays are variously orientated, making intersection in 3-D velocity space possible.

From the smoothing point of view: you cannot make a 2-vector field any smoother (at least in terms of of gradient-squared) by adding a third, orthogonal component to any or all of its members. But in the perspective case the sight-rays are not orthogonal to the image plane except at the principal point. There will, in general, be a gain in smoothness to

be obtained by adding components along the sight-rays. To take the simple example of an approach to a fronto-parallel plane: This gives rise to a divergent field of vectors in the image plane - with the focus of expansion being the principal point. But the field of **p**-vectors is perfectly uniform, since true velocity and depth are the same for all points. The integral of gradient-squared is finite for the **I**-field and zero for the **p**-field.

As the angle of view narrows so the cross-constraint on **p** and the gain to be had from extending smoothing to a third dimension disappears.

4.5. A subtlety

If there exists a first order **p**-field consistent with the image data then there exists a one-parameter family of such fields. A **p**-field:

$$\begin{aligned} p_1 &= -a - (b+D)x - cy \\ p_2 &= -d - ex - (f+D)y \\ p_3 &= -(g+D) - hx - iy \end{aligned} \qquad (4.5.1)$$

gives rise to an I-field which is independent of D. The natural interpretation of D is that it is a 3-D "dilation" parameter which defines the extent to which the viewed scene is imploding towards or exploding away from the pinhole along the sight-rays. Such an effect is first-order in 3-space but is, obviously, undetectable in the image.

We have the variational curiosity of an infinity of stationary states, only one of which corresponds to the global minimum state. Furthermore there is no guarantee, in the planar rigid-motion case, that the global minimum state of the **p**-field corresponds to its veridical condition. However, from the point of view of reconstructing the **I**-field this is irrelevant - since all members of the stationary family project to the same **I**-field.

4.6. Computation

To adapt the Horn/Schunck algorithm to recover the **p**-field we merely substitute equation (4.3.5a) as the data-constraint and associate a third vector component with each pixel. Making the substitution c for -(ax+by) equation (4.3.5) now reads:

$$m = ap_1 + bp_2 + cp_3 \qquad (4.6.1)$$

and the opinion of the neighbours can be summarised:

$$p_1 = \overline{p_1} \quad \text{and} \quad p_2 = \overline{p_2} \quad \text{and} \quad p_3 = \overline{p_3} \qquad (4.6.2)$$

We select a value of σ and compute a least-squares fit to these four conditions to yield the values of u, v and w which minimise:

$$(1 - \sigma)^2 (m - ap_1 - bp_2 - cp_3)^2$$

$$+ \sigma^2((p_1 - \overline{p_1})^2 + (p_2 - \overline{p_2})^2 + (p_3 - \overline{p_3})^2) \qquad (4.6.3)$$

4.7. Applicability of p-field smoothing

So far as *translational* motion is concerned the **p**-field is of lower order than the **I**-field to the extent that there is a component of motion along the optic axis. The interaction of axial motion W with the slant components of the plane (L and M) generates a second-order effect on **I** but a first-order effect on **p**. In the case that motion is purely axial (U = 0, V = 0) and the plane is parallel to the optic axis (N = 0) the resulting I-field is purely second-order. Such a field arises on the retina of a forward-facing observer in an aircraft about to touch down, for example. Even if the angle of view is restricted this is one situation in which it might not pay to make the "narrow angle approximation" and drop the second-order terms!

The interaction of axial motion and the fronto-parallel component (N) of the plane generates first-order effects on the **I**-field but zeroth-order effects on the **p**-field. In a pure "WN" case (U = V = L = M = A = B = C = 0) smoothing of the **p**-field will thus yield a recovery undistorted by boundary effects.

Rotation about an axis other than the optic axis causes second-order effects on **I** and first-order on **p**, as well as generating in both fields a zeroth-order component which can be difficult, if the viewing angle is small, to distinguish from a component due to translational motion. If the camera is "fixating" a point - tracking it - these zeroth-order components tend to cancel out and we are left with a predominantly second-order field. Where motion parameters are more general rotational second-order effects tend to be small (relative to the overall magnitude of image motion) except towards the periphery of a wide-angle image.

It is not the angle of view itself which is important in the "wide angle" case but distance from the principal point. It is arguable, however, that a sound way of dealing with this case is to transform pixel coordinates and motion constraints to simulate a rotation of the camera to centrally position the region of interest. The second-order components would then become of minor importance and **I**-field smoothing would be adequate.

4.8. Empirical results of p-field smoothing

I have compared both I-field and p-field versions of Horn and Schunck's algorithm on a subset of the London Bus raw velocity data. The image was divided into an 11×11 grid of "big pixels" each containing 5×5 of the original pixels. The best *single* component of the 50 contained within each big pixel was chosen to be the single "edge motion" vector associated with the big pixel. This constitutes the "Big Pixel Edge Data". Such a coarse image is computational fast to work with and easy to display meaningfully.

When smoothing was carried out across the 11×11 field both **I**-field and **p**-field smoothing leaked badly across the boundary of the bus at interesting levels of smoothing pressure so I selected the 9 vertical × 6 horizonal region which the bus roughly subtends as a sub-image and applied smoothing to that.

The accuracy of the recovered *I-field* was used as the standard in each case (bearing in mind the one parameter indeterminacy in the **p**-field). The error measure was the E-norm (the sum of squares of the components of each vector orthogonal to the line connecting the pixel to the focus of expansion).

We have a large relative magnitude of p_3 in this image and, with such a narrow angle of view, convergence of the **p**-field from a {0 0 0} starting position proved to be painfully slow. To accelerate matters I rescaled image-coordinates by a factor of ten (thus simulating a wide angle image) for the first 20 iterations - and then rescaled both it and the estimates of p_3 back to the correct focal length before proceeding.

With moderate smoothing pressure (the opinion of the neighbours carrying twice as much weight as the opinion of the data on average) the recovery errors after 50 iterations were:

```
I-smoothing:  1.17
p-smoothing:  0.64
```

After a further 50 iterations at very high smoothing pressure (which effectively forces a zeroth-order condition via the flattening effect at the boundary) the errors were:

```
I-smoothing: 2.13
p-smoothing: 0.06
```

The **p**-vectors, after 100 iterations, were all aligned within 5 degrees of the true 3-D direction of travel of the bus.

The state of the I-field recovered directly (top two pictures) and via the the p-field (bottom two pictures) are shown at the two stages in figure 4:4. The parameter s which has values 1 and 5 in figure 4:4 is not σ, though it is related to it. Computationally it is convenient to give the "opinion of the data" its full weight (which averages about 0.5 for each pixel in this data) and simply add on the "opinion of the neighbours" weighted by a smoothing

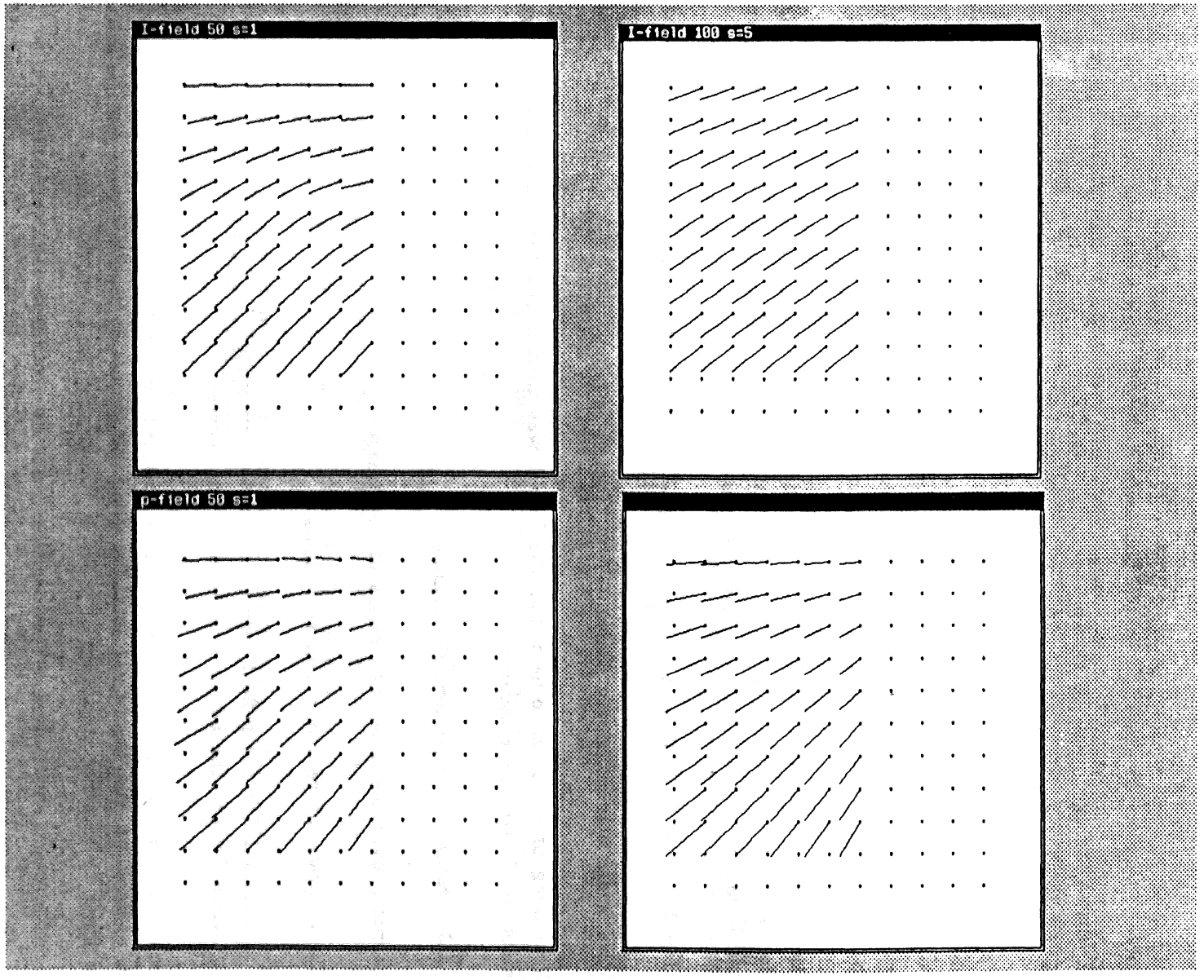


Figure 4:4 11x11 velocity field with I-smoothing (top pair) and p-smoothing (bottom pair). (see text)

pressure parameter s. To provide a crude simulation of Hildreth's algorithm I ran the Horn/Schunck procedure on the 28 boundary pixels only - inner pixels being ignored both as sites of computation and as neighbours. With the same parameters as previously the results were:

	50 cycles	100 cycles
I-sxaoothlng	**0.45**	**0.62**
p-smootliing	**0.26**	**0.16**

For various reasons the error measures for the Horn/Schunck case and the "Hildreth" case should not be directly compared. In both cases the deterioration of the result with the conventional algorithms is due to the zeroth-order erosion effect. This favours the p-field version in both cases since it is, truly, nearly zeroth-order. (There is some depth variation across the bus so the true p-field is not perfectly uniform).

As a test on a case with significant second-order effects I generated a very wide angle (70 degree viewing angle) perspective image with random edge orientations in each pixel, and the following motion and planar normal parameters:

U = 1 V = -2 W = 4 L = 1 M = 1 N - 0.5 A - 0 B = 2 C = 0

(this image was extremely coarse - 7 x 7 pixels only)

Smoothing of both the p-field and the I-field was carried out for ten iterations. I use as a measure of error in recovery:

$\overline{Z}$rll-veridical - I-estimatedlVlI-veridical!^{2}

(summed over *the* inner *5n5* region of the image)

The accuracy of recovery was as follows:

circa 0.1 for the p-field case (and slowly improving)
circa 0.6 for the I-field case (and getting worse)

However, I am inclined to regard results obtained on synthetic imagery as meaningless, save as a check on the theoretical analysis, since differences in recovery-error can be arbitrarily enlarged by relaxing standards of what is *reasonable* in a velocity image.

The vector p, though probably of marginal practical importance, is an important element in our argument - it teeters on the brink between "local[11] and "global* and cannot be emphatically classified as either. While it may be viewed as an image quantity, being a simple generalisation of the the conventional image motion vector, its 3-D direction is aligned with the true motion which it represents and it thus has a close affinity with the

outside world.

4.9 Combining global rigidity with local smoothness

It is apparent that, save in the case of a single planar surface, the velocity field recovered by any of the smoothing processes discussed above will not be consistent with either the global assumption of rigidity *or* the global assumption of planarity. Rather it will tend to be consistent with a smooth but non-planar surface (a "rubbery plane") undergoing rubbery deformation. To get a field which is consistent with rigid motion we have to "go global" - but, as we noted in 2.7, the rigidity assumption alone does not enable us to interpret edge-motion data unambiguously. We might marry global rigidity with local smoothness thus: find the motion parameters which yield the inferred surface which is smoothest by some definition. I have experimented with an algorithm (GLS2) which strives to do this and briefly describe it in this section. The iterative cycle is as follows:

1) Some initial estimate of motion parameters is made.

2) Given motion parameters we have one or more equations for each data point which contain the unknown z. These are "constraint equations" on z which reflect

a) "the opinion of the data" combined with

b) currently estimated global motion parameters

We proceed to estimate inverse depth at a point by joint reference to its constraint equation(s) and

c) the state of its neighbours.

We "relax" inverse depth, given data and motion parameters, in a manner closely analogous to that in which we smooth the velocity field.

3) New global motion parameters are estimated on the basis of these local inverse depth estimates, using equations of form (2.8.5)

$$v_1 = cu + sv = (W(xc+sy) - Uc - Vs)z$$

$$+ A(cxy + s(1 + y^2)) - B(c(1 + x^2) + sxy) + C(yc - xs) \qquad (2.8.5)$$

(These are the equations we could not use earlier because of the unknown z)

4) Return to step 2 unless no significant change

This differs from the point-motion algorithm GLS1 only in respect of step 2. In the

point-motion case each point can be regarded as consisting of two "infinitely close" neighbours (each with an edge motion associated). In such a case finitely close neighbours need not be consulted in adjusting estimates of depth. Smoothness of the depth field in the point-motion case amounts to *single-valuedness* of the field (there being in general only one single-valued field consistent with a rigid interpretation). But where we do not have point motion we must cross-constrain between neighbours which are finitely separated.

The natural smoothness measure is the (sum over the image) of some estimate of grad-squared of the inverse depth (z) field- We take *inverse* rather than absolute depth because the former has scale-invariant properties which seem highly desirable. Under perspective projection two points may be:

a) very distant from the viewer
b) very distant from each other, and yet
c) very close together in the image.

Two. stars, to take an extreme case, may be close in the image yet light-years different in depth. If we were to attempt to smooth the absolute depth (Z) field then the separation in depth between two distant objects subtending a small angle at the viewer could completely dominate the process. But the inverse depth disparity between two distant objects (however distant from each other) will tend to be small. The statistical model of the world appropriate to inverse depth smoothing would be closely related to a "fractal" model.

I have experimented with both discrete and differential measures of the inverse depth gradient. In the latter case we estimate z for every pixel (even if it has no data associated with it) by reference to the state of its immediate neighbours. This is a conventional "relaxation" approach in which the influence of data is propagated across the image by strictly local computations. But in the discrete variant we estimate z only at pixels which contain data and we do so with reference (at least in principle) to all other data-containing pixels in the image. The influence which the data of pixel P carries in the least-squares estimation of z at pixel R is inversely proportional to the square of the distance between P and R. This approach has the advantage that it is readily modified to deal with surfaces which are rough at small scale but "smooth" on a more global scale.

The inverse depth field will be zerath order in the case of **a** fronto-parallel plane and first-order for **any** other plane. Gradient-squared smoothing of the inverse depth field thus incorporates a bias towards planes and *wUMn* that a further bias, due to boundary effects, towards fronto-parallel ones.

4.10 Performance of GLS2

I have applied GLS2, with rotation parameters A, B and C fixed at zero so that we are estimating only for the focus of expansion, to the Big Pixel Bus Data. With a "smoothing weight" of 0.5 (the opinion of the neighbours carrying about as much weight as the data on average) and an initial motion estimate of {U V W} = {-1 0 0} the program rapidly (10 iterations) converged to a state in which the focus of expansion was: x = 0.12, y = -0.23. (The true focus of expansion, estimated from the appropriate vanishing point is approximately at x = 0.26 y = -0.24).

The inverse depth field (scaled for convenience of display) was as follows:

z-MAP

5	5	5	5	5	4	3	2	1	0	0
3	6	6	6	5	5	3	2	1	0	0
6	7	7	6	6	6	4	2	1	1	0
6	7	8	8	7	6	3	2	1	1	0
5	7	9	9	7	5	4	2	1	0	0
5	6	7	7	7	6	4	2	1	1	0
5	6	6	6	6	5	3	2	1	0	0
5	6	5	5	5	4	2	1	0	0	0
4	4	4	5	4	4	1	0	0	0	0
0	0	0	1	1	1	0	0	-1	0	-1
-2	2	2	-2	-1	1	0	-3	1	1	-3

The visibility constraint was not enforced and the video noise band shows up as points of mostly negative depth (recall that "depth" is a pseudo quantity in the case of the bus imagery). The z-field has "leaked" between the bus and the background as we would expect.

I have tested the algorithm (without fixing rotational parameters) on synthetic data produced by:

a) generating linear, cubic or fractal inverse depth fields

b) selecting motion parameters

c) generating from the inverse depth field and the motion parameters an edge-motion image

Where the angle of view is considerable, and sampling of the field is sufficiently dense, recovery of motion parameters and depths tends to be excellent - subject to the proviso concerning "phony duals" in situations where the inverse depth is close to first order. When the angle of view is narrow the smoothing process "takes advantage" of the relief-scale ambiguity as we would expect to produce a very flat inverse depth field. The

quantities which we expect, from the small-object analysis, to be stable are well recovered in this situation.

Alternation of local and global procedures is a feature of some other algorithms - for example that of Brooks and Horn (1985) which recovers shape-from-shading by alternately:

1) estimating local surface orientation by joint reference to

a) "the opinion of the data" (the constraint relating intensity to orientation, reflectance and illumination),

b) the currently estimated (global) direction and intensity of the light source,

c) the "opinion of the neighbours"

2) estimating the global lighting parameters with reference to currently estimated surface orientations at each pixel

4.11 Towards a unified process

Any set of linear equations can be solved in a distributed computation by the Gauss-Seidel process. We may start by treating the equations as "constraint equations" and distribute the data across the image. If the data is error free then *eventually* the field will collapse to zeroth order and the state of every pixel will correspond to the global solution. The "eventually" is advised because the Gauss-Seidel procedure is a highly inefficient way of obtaining uniform fields. Its great strength is that gives us "an order for free". It will rapidly give us a *smoothly varying* condition but very much more slowly will it bring us to an *unvarying* condition. It will take a relatively short time for the elements to render the Himalayas smooth compared to the the aeons it will take to obliterate all traces of them. So, for example, if we wished to obtain a single, global estimate of the eight parameters P1....P8 in equation (2.8.4):

$$v_1 = cu + sv$$

$$= P_1c + P_2s + P_3cx + P_4sx + P_5cy + P_6sy + P_7(cx+ sxy^2) + P_8(cxy + sy^2) \qquad (2.8.4)$$

it would be very inefficient to treat it as a "constraint equation" at each pixel and then iteratively compute at each pixel the state of the vector $\{P_1\ P_2\ P_3\ P_4\ P_5\ P_6\ P_7\ P_8\}$. Since we get "an order for free" it is much more sensible - if we are going to use local computation - to "go down an order" and compute a vector of quantities which we expect to display first-order variation across the image. This is exactly the relation of the vector p (introduced in section 4.3) to equation (2.8.4). Using the 2-component image vector I is "going down too far" for the perspective, planar case since it shows second-order variation -

and the Gauss-Seidel procedure will do its best to smooth that out, irrespective of boundary conditions. So, if we go down in order too far we get erroneous solutions and if we don't go down far enough we get terribly slow solutions. Even watching the progress of smoothing of the image vector **I** can be an excruciatingly boring experience. For example, if all the edge motions in the left of the image are oriented East-West and those in the right of the image North-South in a uniform **I**-field the Horn/Schunck algorithm will generate a remarkably *smooth* field in a number of iterations equal to about half the width, in pixels, of the image. But it then takes many times as long to produce a situation in which the pixels at the left margin and those at the right margin are in agreement to within a few percent. So the gift of an order is not always welcome. This is a situation in which it is not possible to go down far enough!

If we expected first order variation in $P_1 ... P_8$ (because we were dealing with curved surfaces for example) then we might be justified in treating (2.8.4) as a constraint equation - defining the constraint hyperplane in "P-space" - and then iteratively compute at each pixel the 8-vector $\{P_1 .... P_8\}$. This implies some fairly hefty processing - an 8×8 matrix inversion at each pixel at each iteration - but we should not allow this to alarm us unduly. Such a computation might be quite rapidly effected with parallel hardware and could, anyway, be broken down into an iterative process in which the different components of the 8-vector were adjusted in turn. (Normally it does not pay to compute the least-squares solution to a set of linear equations iteratively, but in this case we are already committed to an iterative process - so it may).

But let us return to the planar case in which (2.8.4) holds globally. We have two methods of solution. The first involves gathering together all the data in one "cluster" and solving for the eight parameters - this is the global method. The second involves leaving the data "spread out" on the image plane and locally computing the vector **p**. If the image is infinite in extent and data is evenly distributed, so that there is no boundary of any description, this second method is "perfect". Recalling that equation (2.8.4) is, in its heart-of-hearts, a first-order equation we might contrast the two situations thus:

> 1) Least-squares regression over a set of first-order constraint equations arrayed on a field of nil extent and, therefore, an infinite boundary:area ratio.
>
> 2) Least-squares regression over zeroth-order constraint equations arrayed on a field of infinite extent and, therefore, zero boundary:area ratio.

Globally fitting the field equations is like carrying out a locally distributed process in a field which is all boundary. A local process in which there is a finite, non-zero, boundary:area ratio aspires, to some extent, to the condition of a global process. That is why such striking differences could be demonstrated between the **p**-field and **I**-field methods in the case of the bus edge data. Raising the smoothing pressure effectively brings the pixels closer together and raises the boundary:area ratio, moving the process in the direction of a "clustered" global procedure. The **p**-field thrived since a single **p**-vector provides a good fit to all the data in the case of our London bus while the **I**-field, which was doing fairly well

so long as it was not squeezed, was forced into an unhappy compromise condition. Purists might say, with much justification, that I didn't play the game.

The problem of controlling the scale at which computation is effectively taking place is not one which I have just discovered or invented, of course. It is not possible for me to do justice, here, to all the work relating to this problem - particularly Terzopoulos' (1982) excellent treatment in connection with surface reconstruction. But my "slant" is slightly different from that of others, in that I am less concerned with integrating processes operating at different scales of resolution. I am interested in what *defines* the effective scale of a process and, beyond that, in how a process might find its "natural scale".

CHAPTER 5

A "NATURAL PARAMETRISATION" APPROACH TO IMAGE ORGANISATION

5.1 Introduction

It is paradoxical that structure and motion is most accurately recovered in situations in which the scene being viewed is rough - precisely the conditions under which smoothness criteria are likely to serve least well. Recalling the analysis in Chapter 2, variation in inverse depth *independent* of image position is a requirement of accurate motion parameter recovery in the general case. And very sharp depth discontinuities can be utilised, theoretically, to immediately separate local motion into rotational and translational components (Longuet-Higgins and Prazdny 1980; Reiger and Lawton 1983). The situation is broadly similar to that in static vision: smooth grey-level functions (or depth maps) contain little information. Points at which there are discontinuities or high-order variation are "where the action is".

The segmentation problem can take many shapes. In the context of smoothing of the Horn/Schunck type the form in "which it most readily presents is that of confining the "spread" of communication between pixels to within appropriate image regions. This is usually construed as the task of locating boundaries (by some means or another) to produce a piecewise continuous image. Unless an image is vary densely sampled and noise-free it is not possible to decide firmly whether a boundary lies between two pixels merely by comparing them. Evidence must be sought for a "coherent" discontinuity. But what is a coherent discontinuity?

Marr (1982) enunciated the principle of the "continuity of discontinuities" in the context of the grey-level image. One edgelet does not make an edge. Evidence for an edge is to be found in the organised grouping of edge-detector responses. His primal sketch sought to identify such "organisation* - thereby filtering out isolated responses and interpolating boundaries through points which did not yield strong responses. There have since been many sophisticated developments aimed at eliciting coherent segmentation into regions which are internally uniform or smooth. The basic problem is that two fundamental, opposing pressures must be reconciled - the pressure to group two points together within a region on the basis of their closeness or similarity and the pressure to separate them by a boundary on the grounds of their difference. At sufficiently fine resolution the "difference* principle would allocate every pixel to a private region of its own. But the "similarity* principle, left to its own devices* would fain allocate every pixel to the same class or region.

5.2 Approaches to segmentation

Blake (1983) effects the reconciliation of these forces by means of a cost of segmentation relative to a cost of grouping pixels of differing intensity. If the segmentation cost is set high then minimisation of overall cost leads to a coarse segmentation of the image - if low to a fine segmentation. Because each unit of boundary "costs" there is a tendency for regions to be compact (have a low ratio of periphery to contained area). The logic can be extended to the segmentation of boundaries themselves. In the image-restoration algorithm of Geman and Geman (1984) a similar partitioning of the image into regions of uniform intensity is sought through stochastic relaxation. An isolated black pixel surrounded by white pixels is more likely to change its state to white than a black pixel which has neighbourhood support from other black pixels. The Geman and Geman approach has been modified for application to surface reconstruction from depth maps by Marroquin (1984). He applies it in combination with the surface interpolation approach of Grimson (1981) to produce a hybrid process which he calls "mixed annealing".

We must distinguish here between approaches which aim at discovering boundaries within the optic flow image *per se* and those which "borrow" boundaries identified in the static image. Many workers primarily seek discontinuities in the flow itself (Fennema and Thompson 1979; Clocksin 1980; Jain et al 1979; Potter 1975,1977; Reiger and Lawton 1983; Schunck 1983,1984; Rubin and Richards 1985; Adiv 1984; Waxman 1984b; Wohn et al 1985).

Others rely on boundaries derived from processing of the static image to establish - or at least to suggest - the location of optic flow boundaries. Hildreth's smoothing algorithm, discussed in the last chapter, implicitly assumes that the closed contour derived from static processing either lies entirely on - or is the boundary of - a single region. Cornelius and Kanade (1983) locate static boundaries by conventional means and apply a smoothing process of the Horn/Schunck type with a modification: at a putative boundary the smoothing process is applied (at each iteration) both with and without allowing communication across the boundary. Which ever gives the best result is allowed to "stick".

The difficulty with depending upon static processes for flow segmentation is that sufficiently marked intensity (or colour or texture) boundaries are not necessarily coincident with flow boundaries. Schunck (1985) reminds us of the intrinsic image paradigm of Barrow and Tenenbaum (1978): a discontinuity in one intrinsic image quantity does not necessarily coincide with a discontinuity in another. Indeed, even so far as optic flow alone is concerned, a discontinuity in one "intrinsic" quality of the flow does not imply a discontinuity in another. In rigid relative motion a sudden discontinuity in depth (across image space) implies a discontuity in the magnitude of the component of image velocity due to translatory motion. But the the *direction* of {U V W} does not change, nor does {A B C} change in either magnitude or direction.

Partitioning of data points into sets without necessarily establishing boundaries in image space can be effected for suitable data by means similar to the Hough transform or by

testing for the analyticity of cliques (Ullman 1979). An algorithm which used a more sophisticated technique for achieving such boundary-free segmentation is described by Scott and Buxton (1985). This has similarities to a "minimum entropy transform" (Tou and Gonzales 1974). The theoretical analysis along the lines of the minimum entropy transform has been taken a little further by Buxton and Murray (1985). The human eye has the ability to deal with multiple-valued fields and it is reasonable to ask what is computationally implied in such a task. However I will not be further concerned here with the problem.

5.3 High-order variation

We saw in the last chapter that a relaxation process which tended to minimise grad-squared is theoretically justified in the case of first-order fields. What is the correct policy in the case of higher order fields? How is it related to a good policy for segmentation. Is a flow-field discontinuity merely a "very high order" variation in the field or are the two problems distinct?

Our difficulties with high-order variation are compounded by the fact that, as explained at the end of the last chapter, the introduction of boundaries tends to rob us of some of the room for variation *within* regions which smoothing methods have to offer. If we adopt a policy which leans heavily on segmentation we are likely to have to return the gift of "an order for free" - and find ourselves computing fields of image quantities which are piecewise *uniform* in the relevant scalar or vector of interpretation parameters. In that case applying a principle of "high but finite order", within regions, often becomes a matter of fitting - by local computation or otherwise - field equations of exactly the required complexity.

In the conclusion of his doctoral thesis Schunck (1983) states that he has "....developed a boundary detection and surface smoothing algorithm based on the observation that scenes, hence images, consist of surfaces separated by a finite number of lines of discontinuity". This immediately evoked in me a powerful childhood memory of smoke seeping through the thatched roof of an African hut on a rainy day. But even if I agree to see what he means his statement is only true, of this world, if by "line of discontinuity" we understand a sudden, undifferentiable, change. The familiar problem remains of how to deal with surfaces like that of the human face, which may be seen as containing a zero, finite or infinite length of lines of discontinuity depending upon definition. Is the nose perceptually "segmented" from the face or is it a region of high, but finite, order variation on the surface of the face? How does the visual system deal with it? The question is not about *scale of resolution* but about *form of representation* at any given level of resolution. If the eye was a statistician would it be a "cluster analyst" who places objects in buckets or a "multi-dimensional scaler" who arranges them in a continuous space to represent their similarity and difference? Most competent analysts use both clustering and scaling methods with equal readiness. They play both "discrete" and "continuous" games *on the same data*. Kruskal (1977) gives an illuminating commentary on the complementarity between the two approaches.

In the remainder of this chapter I examine a simple method of parametrisation of the image which responds to high orders of variation in image space by attentuating and strengthening the relationship between pixels - without entirely severing it. There are simple procedures for *then* attempting to cluster the pixels into discrete sets if required.

5.4 "JK" space

The basic requirement of processes which attentuate (or sever) communication between pixels is to prevent a higgledy-piggledy system of relationships from developing (such as: A loves B, B loves C, A detests C, C adores A etc.). Here we do this by constraining all pixels to lie in a 2-D parameter space or "association[11] space - their associative distance from each other being their Euclidean separation in this space. This severely restricts the number of possible associative configurations to a subset of states which might plausibly be called coherent.

We start by defining the "strength of association" or "affinity* AA between two pixels as the square of their Euclidean separation in "JK" space. J and K are the two dimensions of a parameter space which is initially coincident with image space (viz. we initialise J «= x, K = y at all pixels). We also define a concept of the similarity SS between two pixels based upon their closeness both in image space and in image qualities (other than J and K). Call the image separation between two pixels AR and their separation with respect to image qualities Qj» AQj, then the similarity SS between two pixels is:

$$SS - JAR!^2 + Zk\hat{}AQ_{\dot{1}}^2 \qquad (5.4.1)$$

where the a-$_{\%}$ are weights determining the importance of various image qualities (e.g. intensity, motion) relative to each other and to image separation. Both AA and SS are inverse measures - the similarity and the strength of association of a point with itself are always zero.

We first address the adjustment of J and K values, *given* image qualities at each pixel, so that the strength of association AA between pairs of pixels best reflects their similarity SS. In general it will not be possible to assign values of J and K in such a way that SS » AA everywhere (or *m* that SS/AA is constant) since the quantity SS is based upon separation in three or more dimensions whereas AA is based upon two. We are farced to "squash* a Mgber-dimensiQned configuration into a lower dimension in such a way as to best preserve the relations between its components. In the language of Multi-Dimensional Scaling (Shepard 1980; Kraskal 1978; Coxon 1982) we wish to minimise the "stress* involved in mapping an M-dimensional configuration into an M-dimensional space (where M < N).

Consider the immediate neighbourhood of a pixel (0,0) in an image with a single varying quality (intensity E let us say):

	-1	3	2	4
y	0	**6**	8	7
	1	5	8	**6**

- 1 0 1
x

With a - 0.2 the similarity between pixel (-1.-1) and (0.0). for example, is:

$$SS = IARI^2 + laAE^2$$

$$- 2 + 10.2 \text{ i } 59- = 3.0$$

whereas between pixel (0.0) and pixel (1.0) it is

$$SS - 1 + IQ.2 \text{ x } It^2 - 1.04$$

A possible formula for adjusting the J and K values of (0,0), which is initially J«0 K*0, is:

$$\textbf{newJ - L(Jy/SSy) / L(l/SS}_{t,}\textbf{p} \qquad \textbf{(5.4.2a)}$$

$$\textbf{newK - ZCKy/SSy) / Z(l/SS}_{i4}\textbf{)} \qquad \textbf{(5.4.2b)}$$

whore the sum is over the eight-connected neighbourhood

i - **-1,0,1** j - **-1.0,1** i & 0 and/or j 5* 0

(We take an average, weighted by the inverse of similarity, over the J and K values of the neighbours)-

In the present example this leads to J - 0.038 K « 0.158 for the centra! pixel.

We base a conventional relaxation procedure upon this type of local computation. Some simple boundary conditions which we will use are:

J is fixed at its original value (•* x) *ml* the vertical edges ©f **the** image, K is fixed (- y) at the horizontal edges

(If J and K are not **fixed** at the **boundary** in some way the parameter space "Wplodes").

Restricting the dimensionality of the parameter space enforces a type of neighbourhood support. Point A will move away (in JK space) from a dissimilar neighbour B only if it has "friends* which it can move *tow-ards** Abo. if it docs move away it will tend to take its friends with it (viz. they will also tend to move away from B - even if they are not

in contact with it).

If we visualise the image mapped on to JK space then the process is one in which the image becomes "stretched" in the vicinity of discontinuities while uniform regions are "compressed". If we visualise the reverse mapping it is one in which "parameter density" rises in the vicinity of high-order variation in image qualities - particularly where these are coherent - and falls in uniform regions.

The "mesh" figures in this chapter exploit this duality. The left-hand image or set of images show the image space as it distorts into parameter space viz. the Euclidean space of the paper is taken to represent JK space and each node in the mesh represents a pixel in the image. I call this "cluster" mesh. The procedure for plotting this is simple:

```
for  x from x_minimum to x_maximum do
  set the pen at the coordinates (J,K) associated with (x,y_minimum)
  for  y from y_minimum to y_maximum do
    draw to the coordinates (J,K) associated with (x,y)
```

This draws the "horizontal" mesh lines. A symmetrical procedure draws the "vertical" ones. To draw the dual mesh, which I call "parameter" mesh, we first estimate (by linear interpolation between adjacent pixels) values of x and y for each pair of integer values (J, K) between the maximum and minimum values of these parameters. We then apply the same procedure as for cluster mesh with J and x, K and y interchanged. (Note that due to the boundary conditions J_maximum = x_maximum, J_minimum = x_minimum etc).

5.5. The JK process at work - a toy example

Figure 5.1 shows in "cluster" representation and figure 5.2 shows in "parametric" representation the outcome of iteratively applying the JK process with varing values of α to the inverse depth map estimated for the bus imagery by GLS2 (Chapter 4).

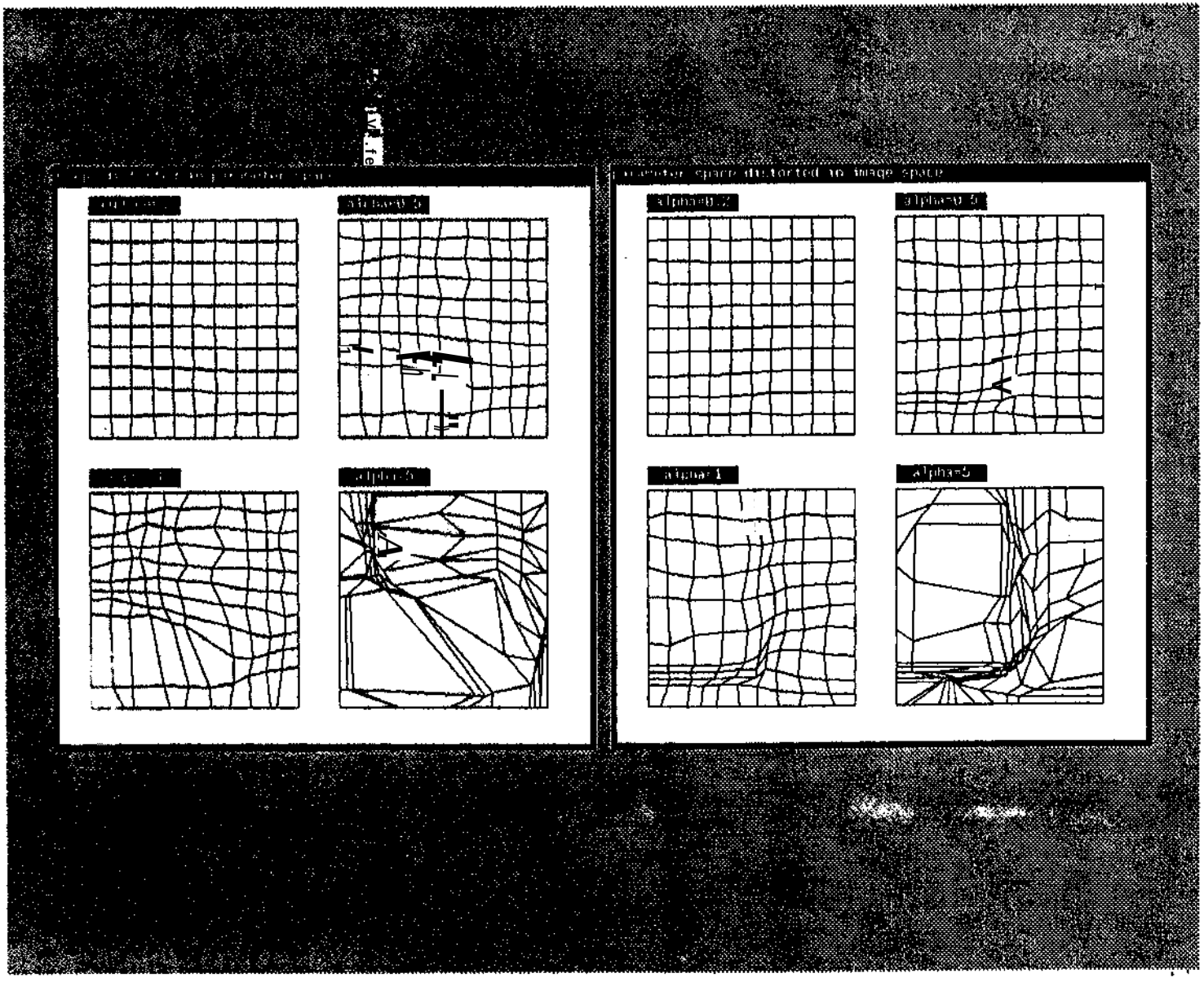


Figure 5:1 "Cluster" mode over inverse depth field

Figure 5:2 "Parameter" [illegible] field

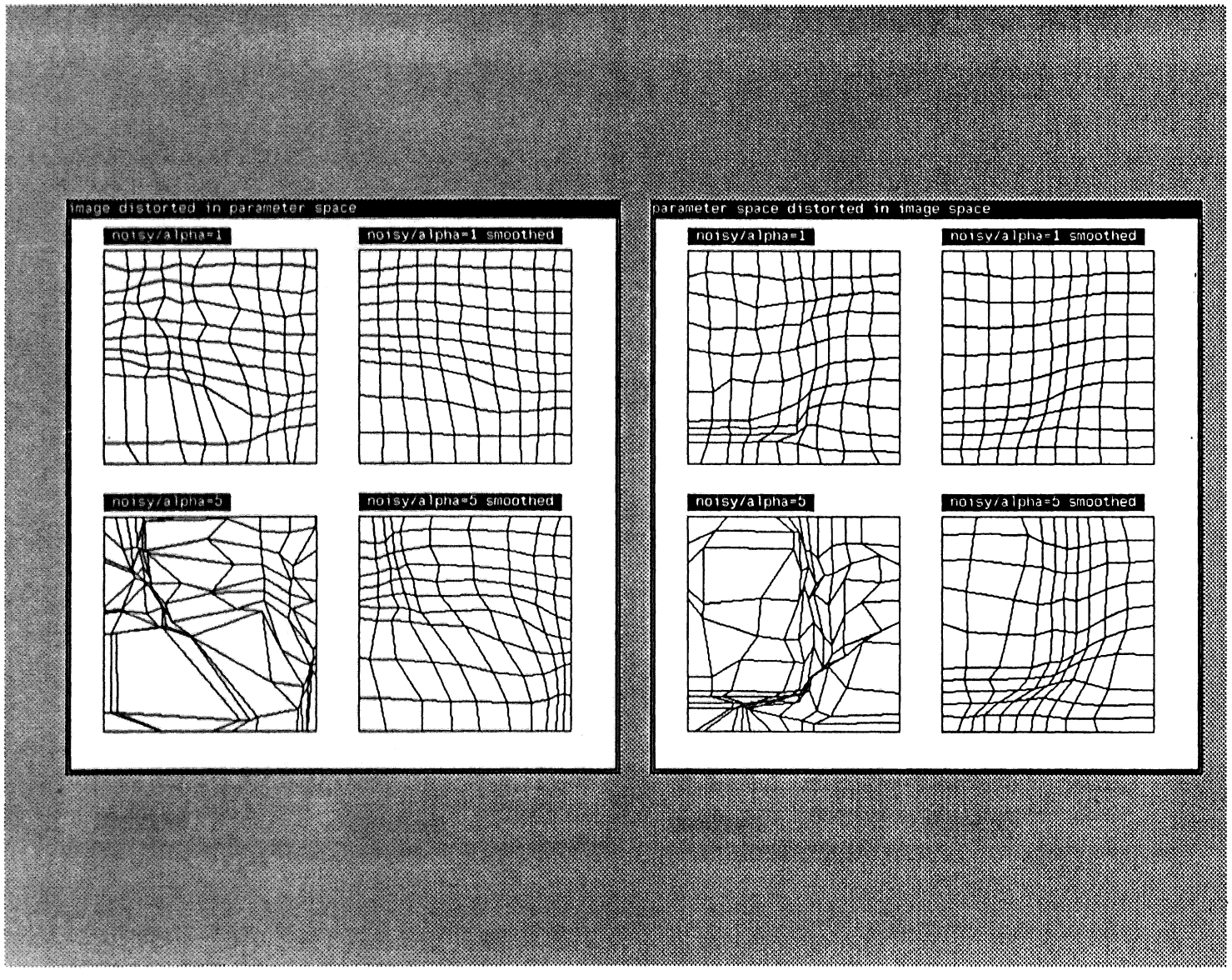


Figure 5:3 "Cluster" mesh with noise

Figure 5:4 "Parameter" mesh with noise

z-MAP

5	5	5	5	5	4	3	2	1	0	0
3	6	6	6	5	5	3	2	1	0	0
6	7	7	6	6	6	4	2	1	1	0
6	7	8	8	7	6	3	2	1	1	0
5	7	9	9	7	5	4	2	1	0	0
5	6	7	7	7	6	4	2	1	1	0
5	6	6	6	6	5	3	2	1	0	0
5	6	5	5	5	4	2	1	0	0	0
4	4	4	5	4	4	1	0	0	0	0
0	0	0	1	1	1	0	0	-1	0	-1
-2	2	2	-2	-1	1	0	-3	1	1	-3

At low values of α, where proximity in image space dominates as a measure of similarity, JK space remains closely tied to image space. When the value of α is very high, so that proximity in depth dominates, groupings of pixels are sharply "clustered" against the edges of the image space in one representation (figure 5.1.d). In the inverse representation figure 5.2.d (parameter space plotted in the undistorted image space) the right and bottom boundary of the bus are sharply demarcated. At intermediate levels of α we have a smooth but high-order mapping between the two spaces suggestive of "contour mesh" representations of surfaces. This surface has actually been smoothed (by GLS2) - so the intermediate states more accurately reflect, to our eyes, the true shape. The high-α states might be regarded as a crude "enhancement" of the image to recover the sharp boundaries which were obscured by smoothing.

Observe the fate of the unsupported non-conforming pixel (1,2) (second from the top in the left hand column) which has been "crowded out" by its neighbours while along the right hand edge of the bus a clear band of attentuation forms due to neighbourhood support. To test this noise suppression effect further we insert three non-conforming pixel values in this depth field (marked below):

NOISY z-MAP

5	5	5	5	5	4	3	2	1	0	0
3	6	6	6	5	5	3	2	1	0	0
6	7	7	6	6	6	4	2	8	1	0
6	7	8	2	7	6	3	2	1	1	0
5	7	9	9	7	5	4	2	1	0	0
5	6	7	7	7	6	4	2	1	1	0
5	6	2	6	6	5	3	2	1	0	0
5	6	5	5	5	4	2	1	0	0	0
4	4	4	5	4	4	1	0	0	0	0
0	0	0	1	1	1	0	0	-1	0	-1
-2	2	2	-2	-1	1	0	-3	1	1	-3

The pairwise differences between these pixels and their neighbours are, by and large, much greater than the differences across the coherent boundaries (or "attentuation bands") which formed with the original data. But they lack neighbours to which they *are* similar, so they do not cause "structure" to appear in the mapping between JK and xy space. Figures 5.3 and 5.4 show, in both representations, the result after 30 iterations of the JK process with α=1 and α=5. For good measure I have also shown the result of a form of smoothing of the the JK-xy mapping. This is effected by running the JK process for a single cycle at α = 0. This tends to return the mapping to its original state in which JK space and xy space are coincident - but "twisted" regions tend to be restored faster than regions of coherent attentuation. The presence of the three "spiky" pixels is not obvious in any of the diagrams, smoothed or unsmoothed.

5.6 Integrated parametrisation and smoothing

Given image qualities and weights α_i we may compute values of J and K for each pixel. *Given* J and K values for each pixel we can carry out a smoothing of image qualities which is based on strength of *association* rather than proximity in image space. The most obvious way to do this is to weight the opinion of the neighbours by the inverse of AA (distance-squared in JK space) instead of by the inverse of distance-squared in image space (as is done in conventional smoothing). If we simply "blur" the noisy big pixel depth field to remove anomalies we obtain a result such as this:

GAUSSED IN IMAGE SPACE z-MAP

```
5  5     6  5  5  4  3  2  2  1  1
6  6  6  6  5  4  4  3  2     1  1
6  6  6  6  5  5  4  3  121  1  1
6  6  6  161 6  5  4  3  2     1  1
6  6  6  6  6  5  4  3  2     1  1
6  6  6  6  6  5  3  2  1  1  1
5  5  (51 5  5  4  3  2  1  1  0
4  4  4  4  4  3  2  1  1  0  0
3  3  3  3  3  2  1  1  0  0  0
2  2  2  2  1  1  1  0  0  0  0
1  1  1  1  1  1     0  0  0  0 - 1
```

(the depths at each pixel have been rounded for convenience of display). This was generated by blurring the depth-image three times by taking an average of the depth at each pixel and the depths of its neighbours weighted by inverse squared distance in image space. There are no remaining differences between pixels (in this integer representation) of more than 1 unit. By contrast three applications of the blurring process with "the opinion of the neighbours" weighted by inverse squared distance in the JK space generated with a - 5 produced:

GAUSSED IN JK SPACE z-MAP

```
5  5  5  5  5  4  3  2  1  0     0
5  6  6  6  5  5  3  2  2  0  0
6  7  7  6  6  6  3  2  121  1  0
6  7  7  171 7  6  3  2  1  1  0
5  7  7  7  7  6  4  2  1  0  0
5  6  7  7  7  6  4  2  1  1  0
5  6  161 6  6  5  3  2  1  0  0
5  5  5  5  5  4  2  1  0  0  0
4  4  3  5  4  4  1  0  0  0  0
0  1  1  1  1  1  0  0  0  0  0
0  2  2  0  1  1  0  0  0  0  0
```

The spQuss have been completely obliterated but boundaries have been preserved. There has been virtually no communication between the Inside and the outside of the "bus* region.

(The affinity metric 1/AA should be stabilised *m* that it has tn upper bound. If two pixels arc "infinitely close* in affinity we obtain a state, pathological in most contexts, in which they "talk* only to each other and pay no attention to third parties. I did not in fact stabilise it in generating the above toy example.)

To entertain the reader's eye and perhaps distract him or her from the considerable

shortcomings of this dissertation (particularly the cryptic omissions in this chapter), I show some results obtained from the high-resolution bus data. The field after 5 iterations of Horn-Schunck smoothing (over the I-field) was shown in figure 1:6. The result of applying the JK process at increasing levels of *a* to this field is shown - in the usual dual representation - in the sequence of figures 5:5a to 5:5f. The mesh has a "soft" character because the underlying velocity image is "soft" or blurred. Figure 5:5f shows the consequence of taking a up to a level where image space proximity no longer counts in assessing similarity. There is a sense that the process is "out of control" here - not to mention an assault on the senses evocative of a bad hangover.

Figures 5:6a to 5:6c show the state of the JK mesh at various stages during a process in which smoothing of the velocity field and the JK process were *alternated,* the smoothing cross-constraint between pixels being regulated on the basis of proximity in JK space. The mesh has a different appearance because the interaction of the two processes "hardens up" the velocity image by inhibiting the spread across bands of attentuation (which in turn leads to an increase in the degree "of attentuation). Figures 5:7 show the velocity field at two stages - corresponding to the meshes shown in figure 5:6a and figure 5:6c. Figures 5:8 show the results of smoothing for an equivalent time at an equivalent pressure *without* the JK process operating. The hardening of the flow boundaries due to the JK process is evident.

5*7 The JK process in *local/global* interaction

The JK process is a tool which might be brought to bear to appropriately inhibit or facilitate mutual interaction between distributed computational processes in a number of ways:

1) We can alternate smoothing and the JK processes.

2) We may assign different JK spaces to different "intrinsic images" - so that attentuation in respect of one quality does not prevent strong affinity in respect of another.

3) Communication across the image at fine and coarse scales will automatically tend to be Integrated by the JK process. The affinity between two sites is a function of their separation in JK space and there would be no need to take special measures to enforce consistency between .affinity at one scale and affinity at another.

4) The a parameter provides an obvious means of controlling the degree to which a particular process tends to be "local* or "global*.

Ideally we would like to see self-adjusting "opportunistic[11] JK processes at work. I have experimented with processes which lead to a collapse of JK space towards a singularity in image space when the relevant Intrinsic image quality is uniform - but which do not so

collapse when it is varying. A simple way of achieving this is to add "anchor" pixels around the outside of the image and assign a "standard similarity" value to the similarity between these and the boundary pixels. If all the image pixels have the same value in the relevant quality (e.g. depth) then the mesh will tend to draw together into a small region in the centre of the image. To the extent that the image pixels vary in value this process will not progress so far and the effective boundary:area ratio will accordingly be less.

5.8 Some propects and problems

The JK process is just one extremely simple (even simplistic) approach to what I have called "natural parametrisation" (the phrase appears to be due to Brady et al 1985). The topic, in a variety of forms and under various names, appears to be becoming central in computer vision. IJCAI 1985 - by way of illustration - contains three vision papers in which natural parametrisation, and ways of effecting or interpreting it, are crucial. One of these is by Weiss whose work on contours I have already mentioned in section 1.11. The other two are by Kass and Witkin and by Strat and Fischler. The first is concerned with the parametrisation of naturally textured surfaces (specifically a wooden plank). The second ("One-eyed stereo") addresses shape-from-contour, though the fact that the authors are dealing with natural parameters is obscured by their curious insistence that the natural coordinate system should appear Cartesian or polar from some viewpoint.

Whereas in conventional smoothing algorithms we construct, by variational means, a mapping of the form:

$$x, y \rightarrow Q_i$$

with the integrated smoothing and JK process we might see ourselves as constructing a mapping:

$$J, K \rightarrow Q_i, x, y$$

The corresponding analytic procedure would be to fit a parametric form in which all observables (including x and y) were *dependent* variables - the independent variables being the "hidden" coordinates J and K. Many computer vision tasks may be seen as implicitly aimed at achieving this. For example the model which underlies LH1 is one in which a 3-dimensional configuration of points is mapped into a 4-dimensional (2 of space and 2 of velocity) image:

$$X, Y, Z \rightarrow x, y, u, v$$

What we recover in a complete interpretation are:

a) The "hidden coordinates" (X, Y, Z) of each point

b) A number of parameters (whose physical meaning relates to the relative motion) which describe the mapping.

In the LH1 algorithm the unknown coordinates X, Y and Z are eliminated algebraicly and the mapping parameters solved for directly. But in my iterative algorithm GLS1 the inverse depth of each point is explicitly estimated - equivalent to explicitly assigning values of X, Y, and Z. In this particular case the "natural coordinates" happen to be Cartesian coordinates in Euclidean 3-space and we instantiate an analytic model of the mapping from them into observables. It would be interesting to explore a variational approach of the JK type to the LH1 problem.

In the "cluster mesh" representation there is a visible tendency for the mesh to become most dense along lines of (bilateral) symmetry and around centres of (radial) symmetry. This suggests a possible connection with various schemes for shape description, notably Brady's "smoothed local symmetries" (1985) and Fleck's (1986) "local rotational symmetries". In the dual "parameter mesh" representation, on the other hand, the mesh becomes most dense along edges *particularly* edges of some reasonable length (for then neighbourhood support comes into play). Here the suggestion is of a connection both with edge detection schemes and with methods of establishing continuity of edges - such as Marr's primal sketch.

My view of natural parametrisation is that it provides the key to generalising *ad hoc* principles of invariance-acroas-projection such as those based on the observation that discontinuities in the image-qualities tend to arise from discontinuities in scene qualities. A possible generalisation is perhaps along these lines:

> Natural parametrisation (or "natural stressing") tends to be projectively invariant (a discontinuity is merely an extreme case of a point or region of "high stress"). The correspondence of the natural organisation of the phenomenon to the- natural organisation of the image is only disrupted if stress intrinsic to the phenomenon is masked or "prematurely alleviated* in the process of projection itself (this is a possible formulation of the principle of General Viewpoint).

The cMef difficulty which I currently have with the JK process itself is that it is not very firmly grounded on principle. I have considerable difficulty with the theoretical analysis of it and could not even say, with certainty, what global quantity is minimised in the examples in this chapter. Obviously it is "stress* in some sense - but in what *precise* sense is unclear. On the other hand the JK process is simple in the extreme and appears to have an affinity with a diverse collection of computer vision processes connected with segmentation, shape description, fitting simple models and early image organisation. It is on account of its promise, and despite in problems, that I have in this chapter given it its first airing.

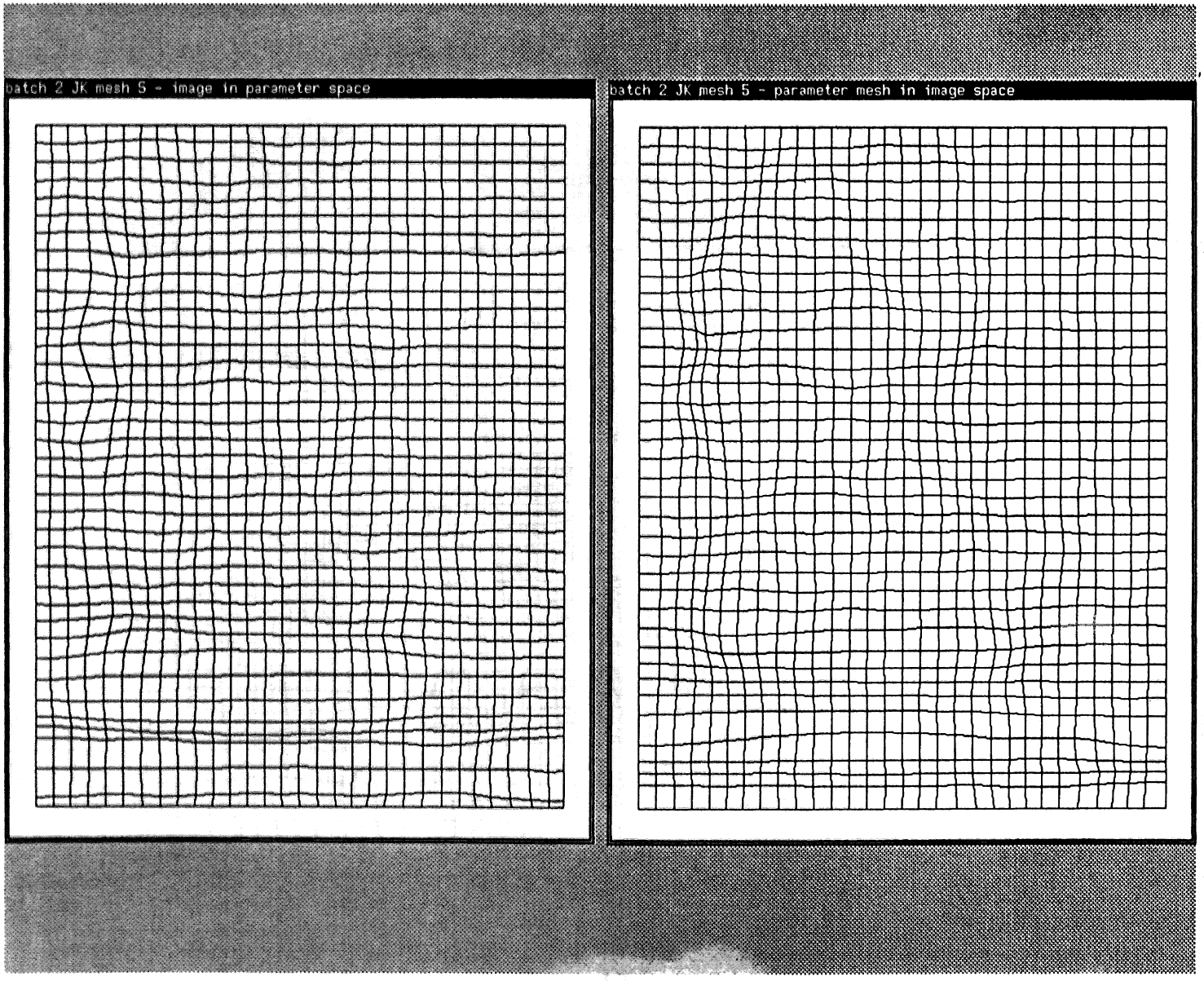

batch 2 JK mesh 5 - image in parameter space
batch 2 JK mesh 5 - parameter mesh in image space

Figure 5:5a

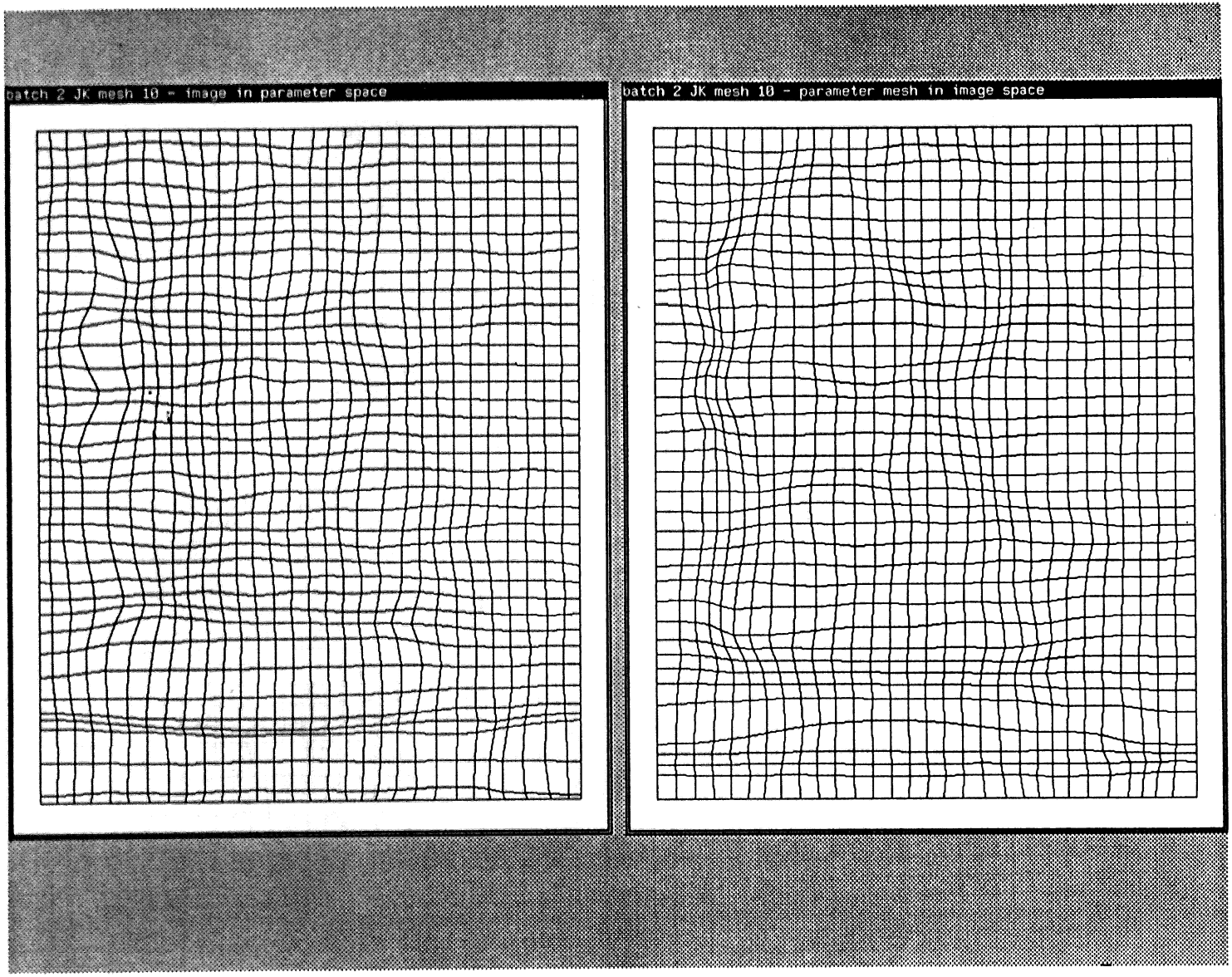

batch 2 JK mesh 10 - image in parameter space
batch 2 JK mesh 10 - parameter mesh in image space

Figure 5:5b

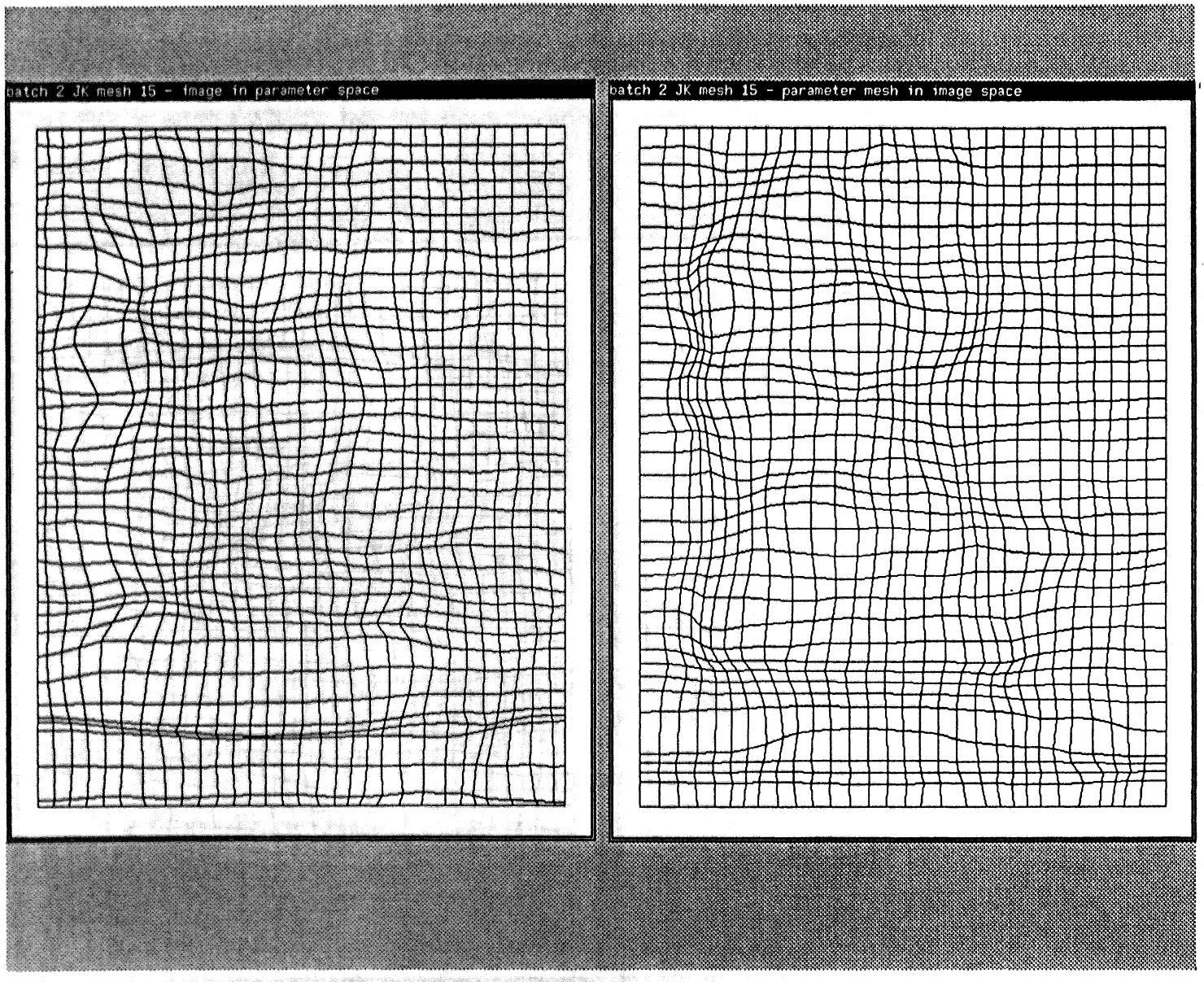

batch 2 JK mesh 15 - image in parameter space
batch 2 JK mesh 15 - parameter mesh in image space

Figure 5:5c

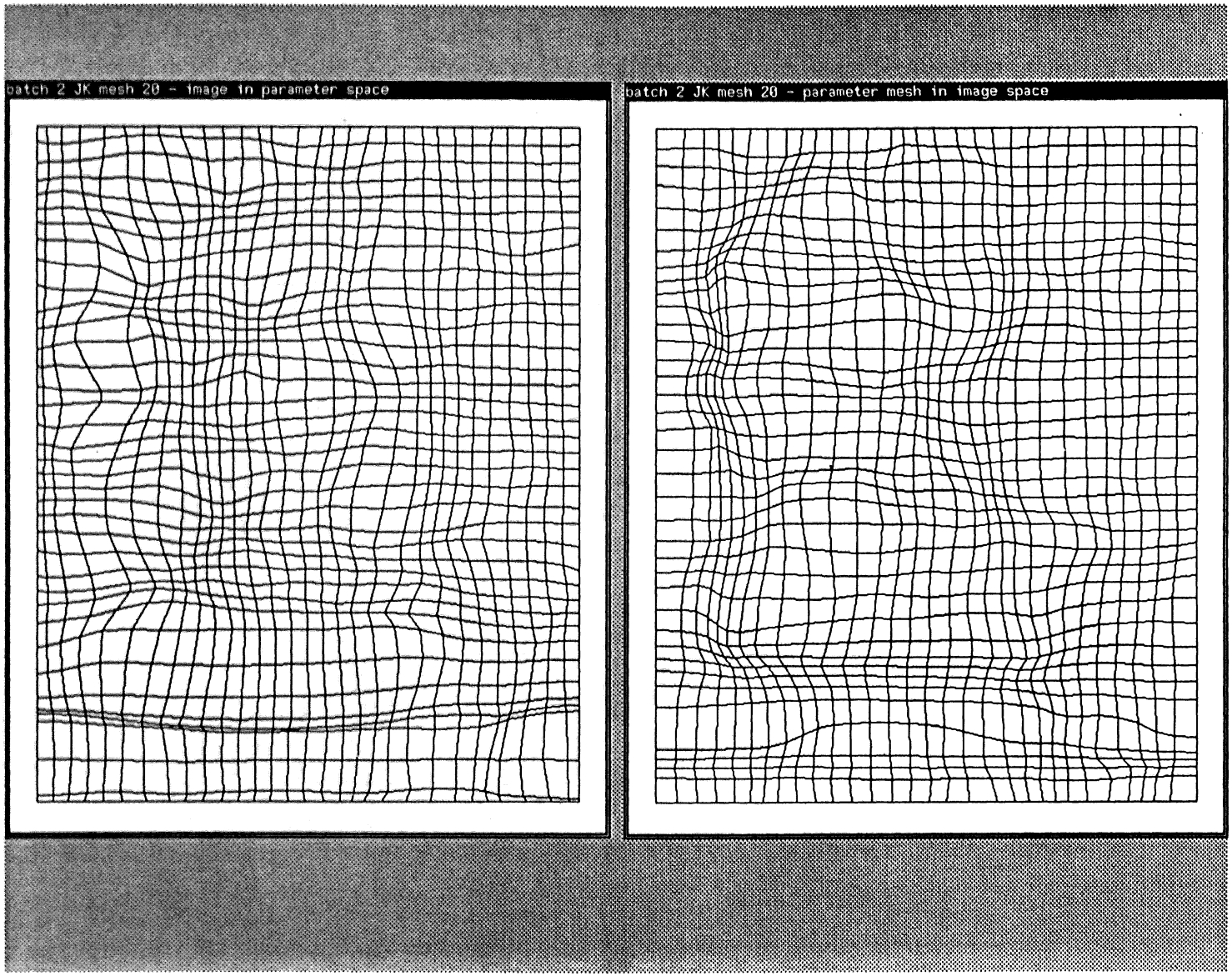

batch 2 JK mesh 20 - image in parameter space
batch 2 JK mesh 20 - parameter mesh in image space

Figure 5:5d

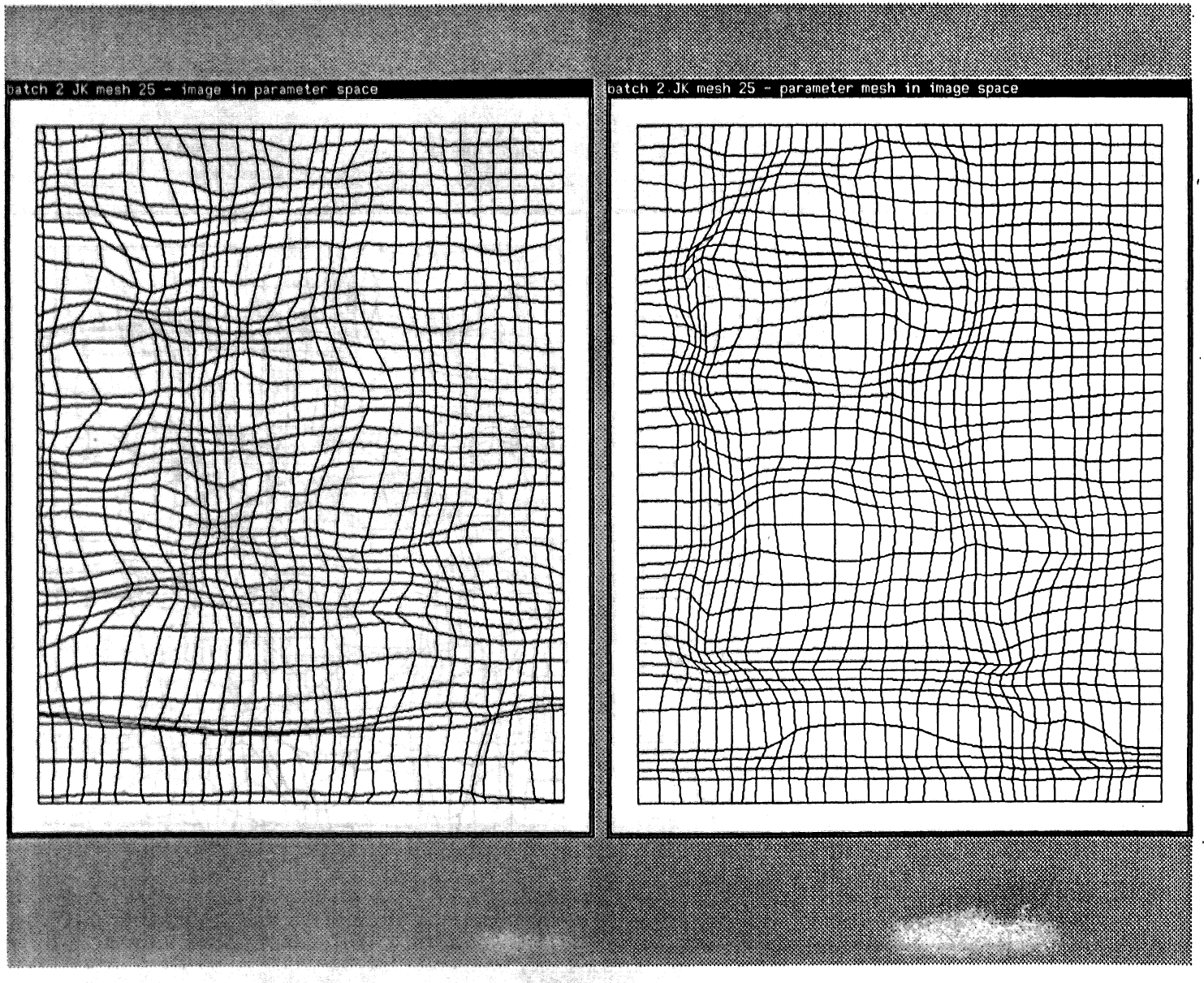

batch 2 JK mesh 25 - image in parameter space
batch 2 JK mesh 25 - parameter mesh in image space

Figure 5:5e

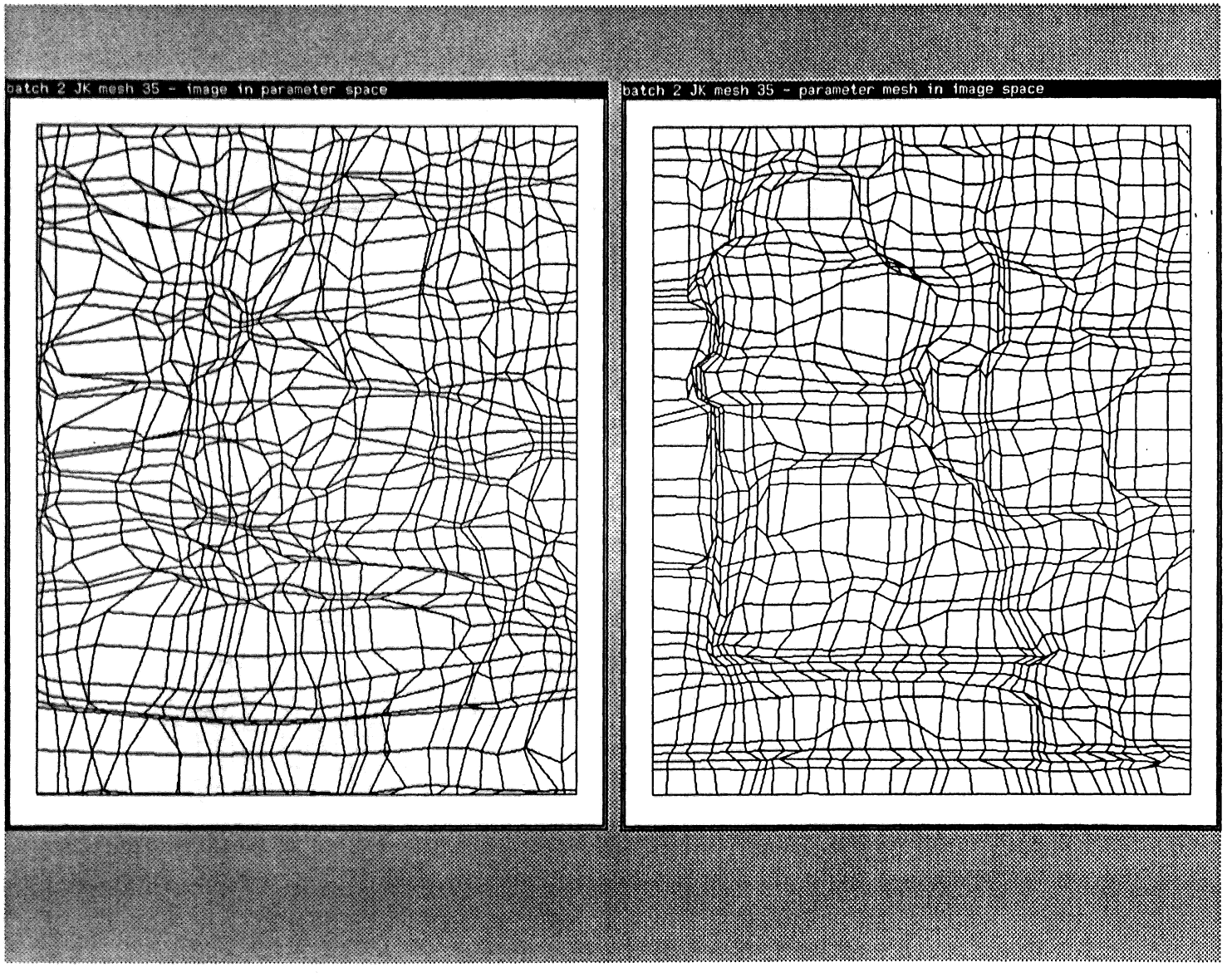

batch 2 JK mesh 35 - image in parameter space
batch 2 JK mesh 35 - parameter mesh in image space

Figure 5:5f

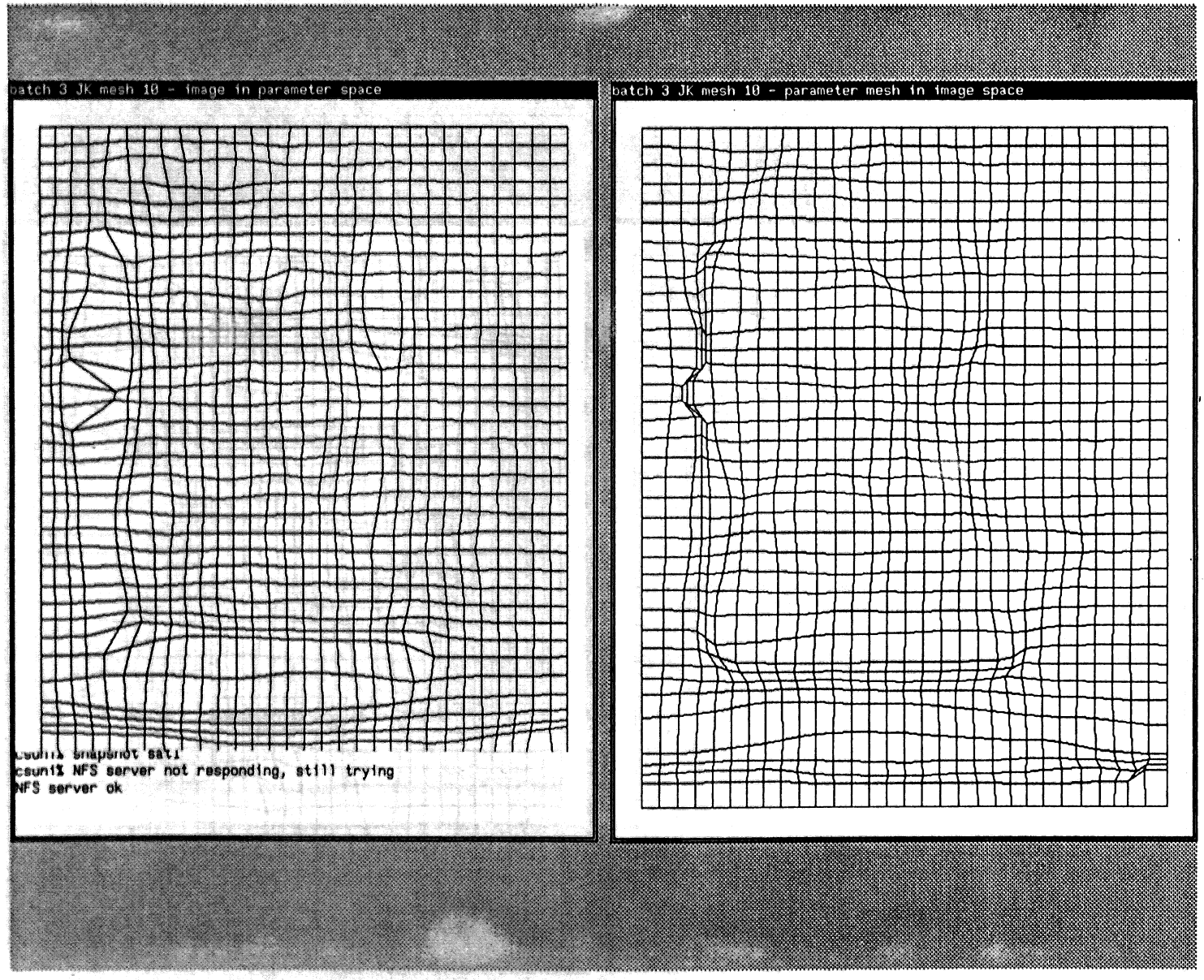

batch 3 JK mesh 10 - image in parameter space
batch 3 JK mesh 10 - parameter mesh in image space
csun1% NFS server not responding, still trying
NFS server ok

Figure 5:6a

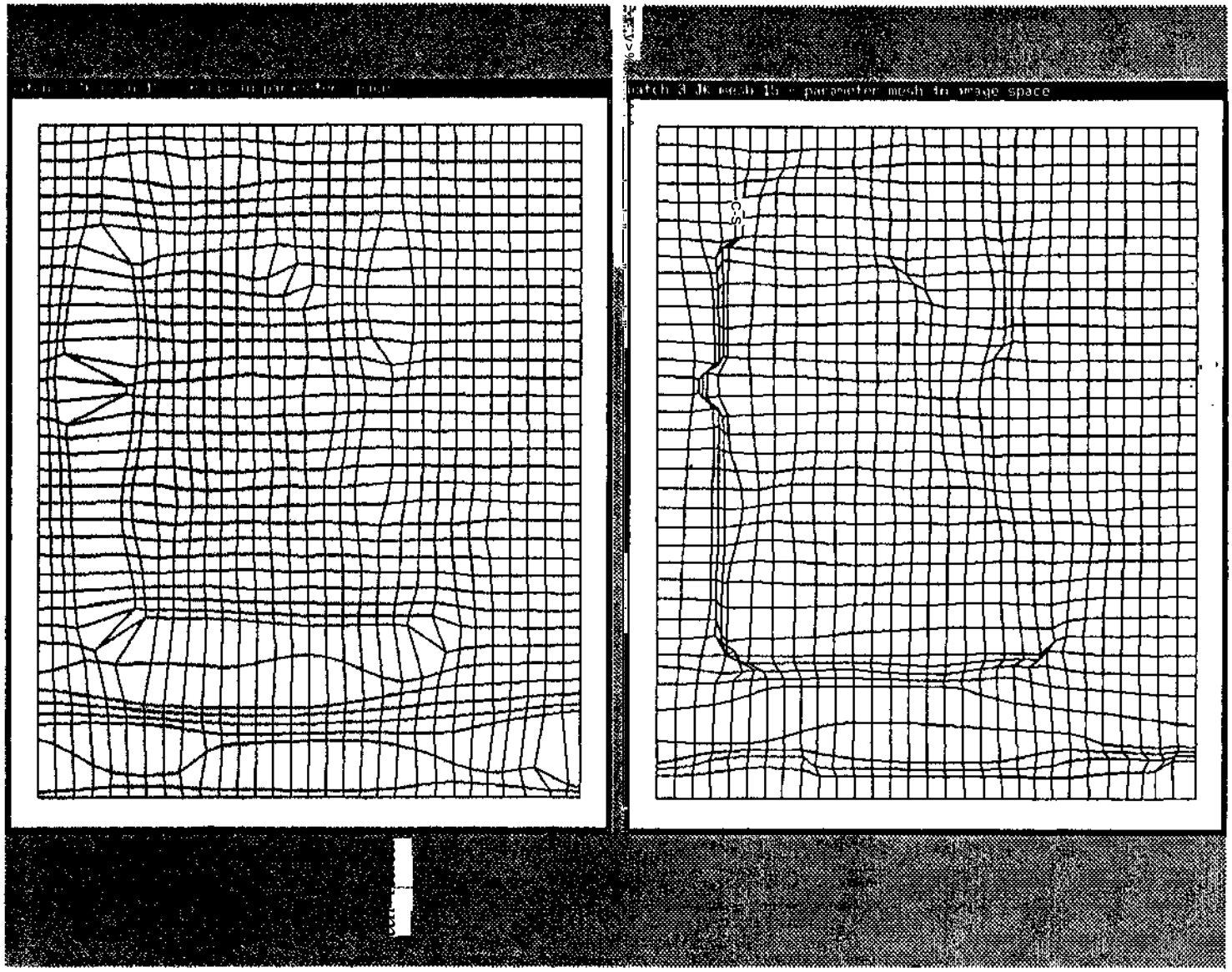


Figure 5:6b

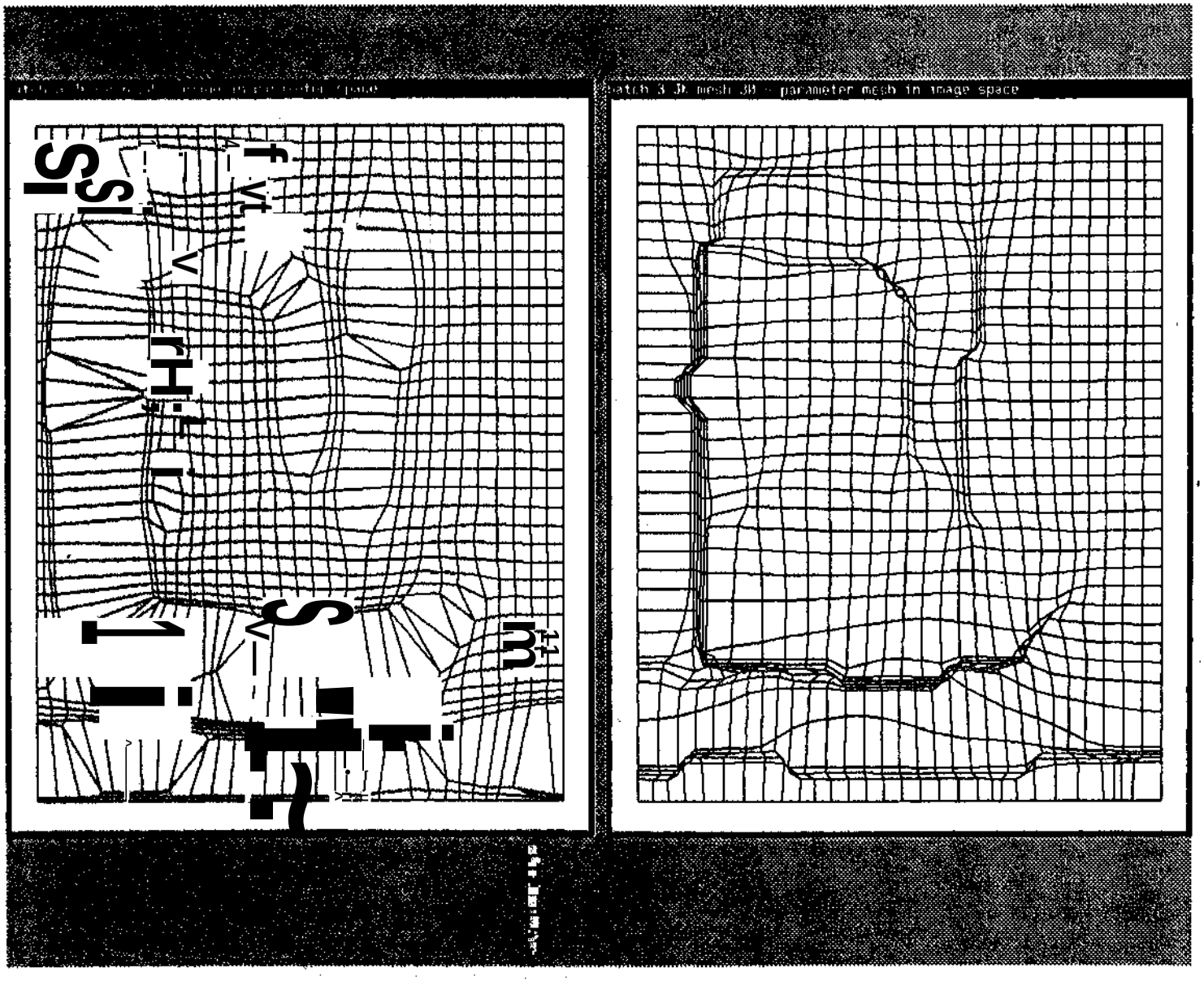


Figure 5:6c

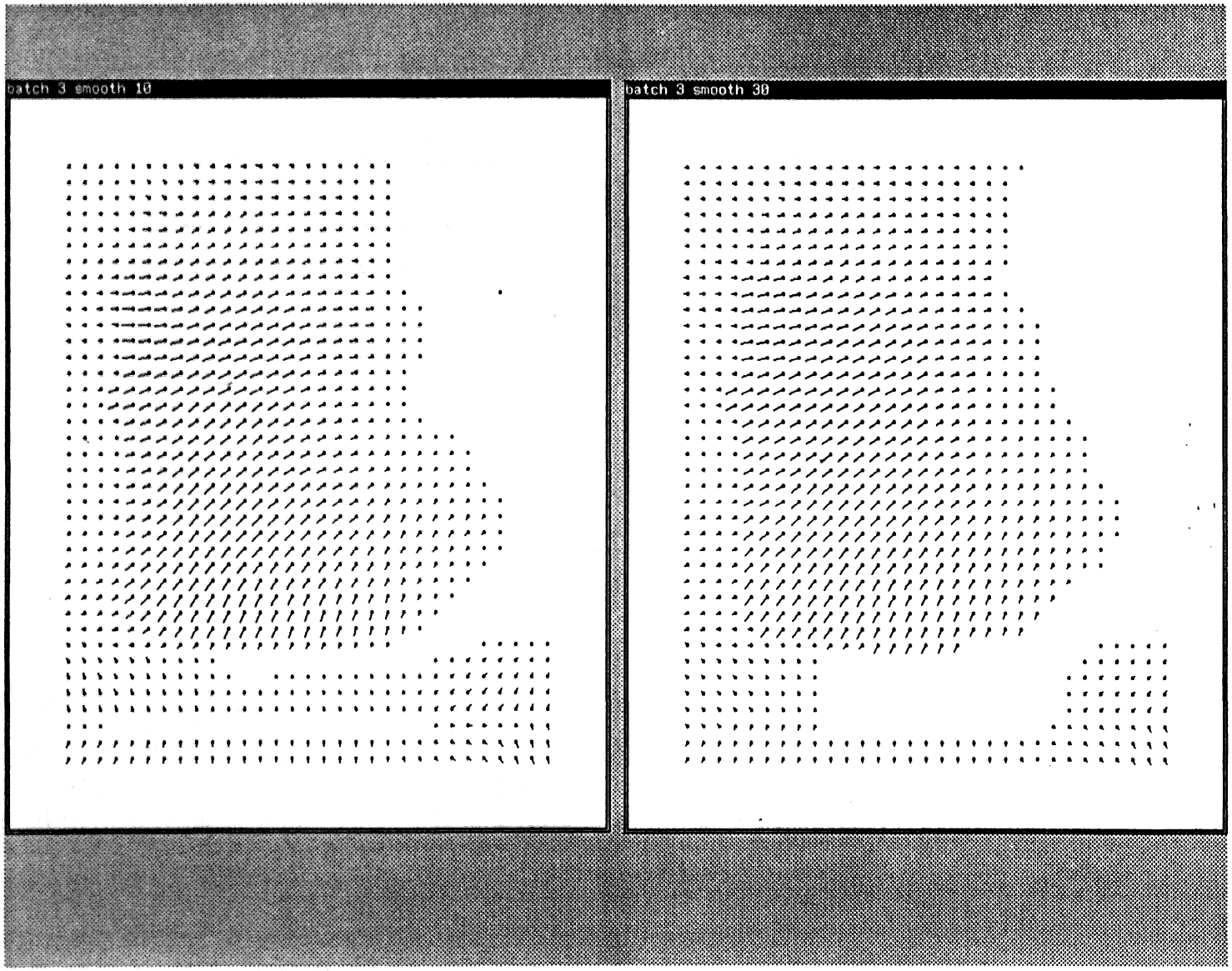


Figure 5:7

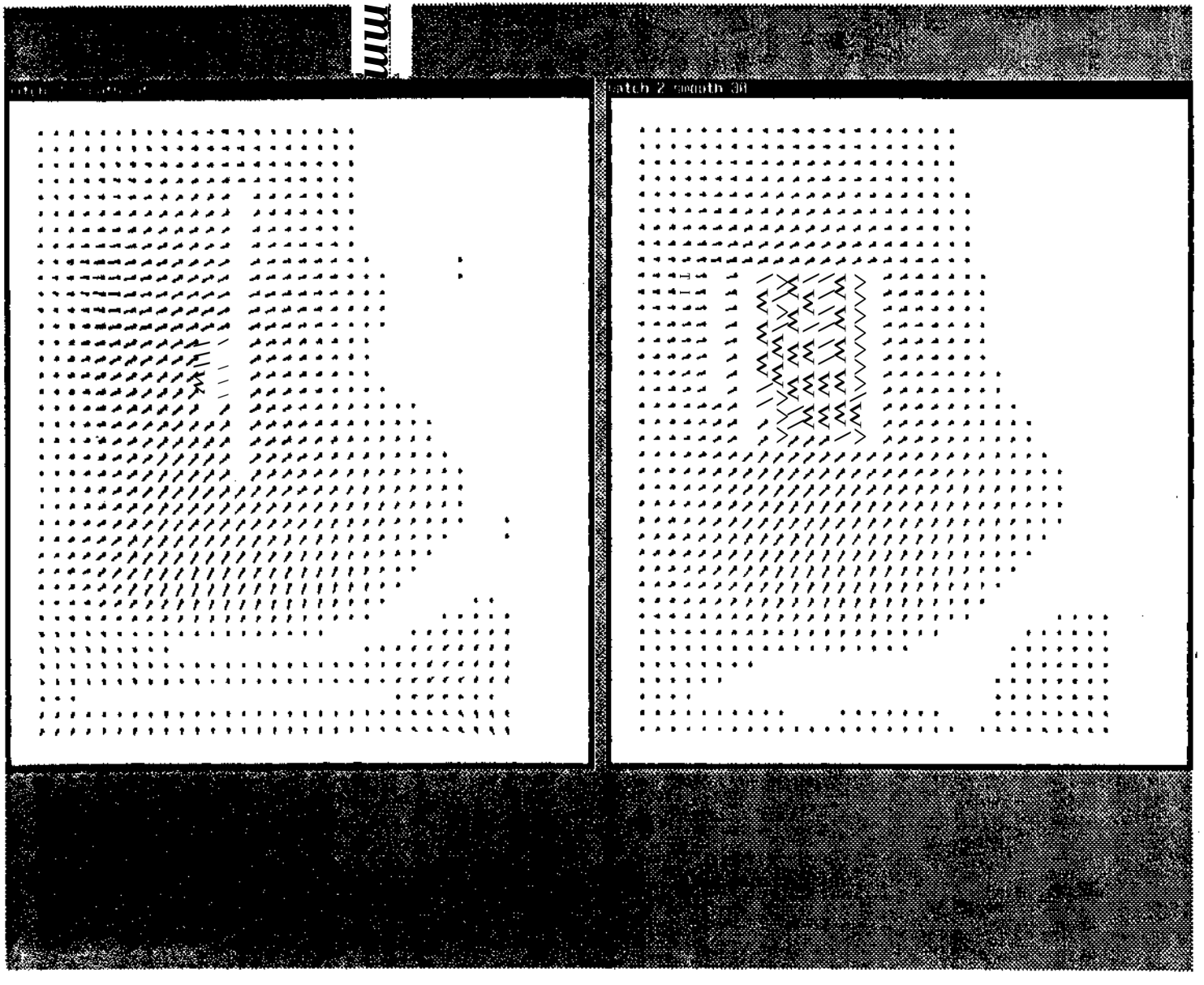


Figure 5:8

AFTERWORD

In this thesis I have sought to effect a "tour of the horizon" of optic flow interpretation, in the course of which I have tried to let myself be guided about equally by sound theory, by the lessons of hard practical computation and by good intuition. The breadth of the ground to be covered has necessitated some abrupt transitions here and there which I hope the reader will forgive. To summarise briefly some of the successes achieved, and some of the unresolved difficulties identified:

In *Chapter 1* a novel means of obtaining local motion constraints was developed on the basis of a theoretical analysis and was computationally applied to the bus imagery. This generated some data (notionally a mixture of point- and edge-motions) to which various algorithms were applied in later chapters. While the principal axes procedure has the theoretically pleasing quality that it unifies edge-tracking, grey-level gradient and feature-matching approaches it remains to be analysed in greater detail and empirically evaluated against previous techniques. Also, as I hope was made clear in this chapter, it does not provide a solution to the "correspondence problem proper".

Chapter 2 studied the 3D interpretation of perspective imagery and clarified certain issues relating to degeneracy, ambiguity, the aperture problem, narrow viewing angles and non-rigid motion. It produced two algorithms, SO1 and SO2, for the case of "the small object approximation" - effectively a form of projection intermediate between perspective and orthographic. Perhaps the main general conclusion to emerge from this chapter (particularly taken in conjuntion with the next) is that we cannot, in realistic situations, sidestep "gooey" issues of Good Form, General Viewpoint, Expectation etc. simply by appeal to the rigidity principle and the laws of Optics. The process of motion perception is as deeply mysterious as that of static image interpretation.

Chapter 3 was devoted to both the theory and practice of interpreting noisy images. A number of algorithms were empirically tested on real data. A new algorithm, GLS1, was described and compared with these. SO1 and SO2 were tested on real data with some measure of success. An algorithm for the planar case - N&H - was modified to work on a part-global, part-regional basis and its superiority in appropriate circumstances over closed-form solution methods demonstrated. The discussion and demonstrations in this chapter were angled in such a way as to illuminate the fact that *even* where the assumptions of the scene-and-motion model are sufficiently restrictive to permit algebraic derivation of closed-form algorithms, the instabilities of these tend to be such that open-form algorithms with "exposed workings" may be preferable.

In *Chapter 4* I developed a principled generalisation of the Horn/Schunck and Hildreth smoothing algorithms to the perspective case (p-smoothing) and demonstrated its validity empirically. An algorithm - GLS2 - which combines a global rigidity condition with extremisation of surface smoothness was also developed and

demonstrated. In the final part of the chapter the vexed question of the relation between local and global computational processes was discussed in terms of the concept of the "boundary:area ratio".

Chapter 5, though the shortest, contains the most original (and least rigorous!) contributions to the theory of visual organisation. The idea that parametrisation can be achieved by effecting a "minimum stress" mapping to a space of lower dimension appears to be entirely new in the field of computer vision (though it has some affinity to other "action-minimisation" principles). The conceptually and computationally very simple notion of the JK process provides a demonstrably effective means of controlling the "natural relatedness" between pixels and, thence, of controlling the degree of mutual intraction between them. The duality of "cluster" and "parameter mesh" representations is a useful aid to visualising the JK process and to grasping its affinity with a range of other approaches to early visual organisation. This chapter affords a brief glimpse into a veritable Can Of Worms - my current area of research.

Computer Vision is an enterprise which has scarcely begun and I feel it appropriate to terminate this thesis on a "open" note - rather than to try and close the narrative with a listing of firm conclusions. (Most of these would anyway only be firm until, as Vladimir Nabokov puts it, "the next bloke grabs the chalk"). It seems to me that our present understanding of vision is woefully inadequate - to the extent that it lacks dimensions, not merely chunks - and in such a situation there is no greater danger than that of a pedantic narrowing and infolding of the discourse. Only to the extent that this thesis has made the slightest contribution to preventing such closure, has it lasting merit.

Brass, A. and Horn B.K.P. (1983), "Passive navigation", *Comp. Vis. Graph. Image Proc. 21* pp 3-20

Burt, P. and Julesz, B. (1980). "A disparity gradient limit for binocular fusion", *Science vdL 208* pp 615-617

Buxton, B.F. and Buxton, H. (1983), "Monocular depth perception from optical flow by space-time signal processing", *Proc. Roy. Soc. Land. B218,* pp 22-47

Buxton, B.F., Buxton H, Murray, D.W. and Williams, N.S. (1984), "3-D Solutions to the aperture problem", *Proc 6th ECAI* pp 631-640

Buxton, B.F. and MurrayJX (1985), "Optic flow segmentation as an ill-posed problem using maximum likelihood", *Alvey Vision Conference*

Cafforio, C. and Rocca, F. (1976),- "Methods for measuring small displacements of television images", *IEEE Trans. Inform. Theory IT-22* pp 573-579

Canny, J. (1983); "Finding Edges and Lines in Images", *MIT AI Technical Report* No. AI-TR-720

Chow, W.K. and Aggarwal, JJL (1977), "Computer analysis of planar curvilinear moving images", *IEEE Trans. Comp. c-26* pp 179-185

Qocksin, W.F. (1980) "Perception of surface slant and edge labels from optical flow: a computational approach". *Perception 9* pp 253-269

Cornelius, N.H. and Kanade, T. (1983), "Adapting optical-flow to measure object motion in reflectance and X-ray image sequences", *Proc. ACM SIGGRAPH/SIGART Workshop on Motion: Representation and Perception:* Toronto, pp 50-58

Coxon. A.P.M. (1982), *The User's Guide to Multidimensional Scaling,* Heinemann Educational Books. London.

Eastman, R.D- *Mid* Waxman, A.M. (1985), "Using disparity functionals for stereo correspondence and surface reconstruction". Computer Vision Laboratory, Universìty of Maryland

Famema, CX and Thompson, WJL (1979) "Velocity determination in scenes containing several moving objects*, *Comp* Graph. Image Proc. 9* pp 301-315

Fleck, M.M. (1986), "Local Rotational Symmetries", *IEEE CVPR Miami* June 1986

Forstner, W. (1984), "Quality assessment of object location and point transfer using digital image correction techniques," *Proc. 15th ISPRS Congress*, Rio de Janiero pp 169-191

Foster, D.H. (1978), "Visual apparent motion and the calculus of variations": in Leeuwenberg and Buffart 1978

Fukinuki, T., Yoshigi, H. and Fukushima, K. (1976), "Improvement of inter-frame predictive coding of TV-signals by utilising visual properties for moving objects", *Trans. IECE Japan 59-A* pp 764-771

Gelfand, I. and Fomin, S. (1963), *Calculus Of Variations*, Prentice-Hall, New Jersey

Geman, S. and Geman, D. (1984), "Stochastic relaxation, Gibbs distributions, and the Bayesian restoration of images", *IEEE Trans. Pattern Anal. Mach. Intell., 6* pp 721-741

Gibson, J.J. (1950), *The Perception of the Visual World*, Howard Miffin, Boston

Gibson, J.J. and Gibson, E.J. (1957), "Continuous perspective transformations and the perception of rigid motion", *J. Exp. Psychol. 54* pp 129-138

Gibson, J.J. (1979), *The Ecological Approach to Visual Perception*, Howard Miffin, Boston

Grimson, W.E.L. (1981), "A computational theory of visual surface interpolation", *MIT AI Memo No. 613*

Grimson, W.E.L. (1981), "A computer implementation of a theory of human stereo vision", *Phil. Trans. Roy. Soc. Lond B292* pp 217-253

Grimson, W.E.L. (1982), *From images to surfaces*, MIT Press, Cambridge

Hadani, I., Ishai, G. and Gur, M. (1970), "Visual stability and space perception in monocular vision: mathematical model", *J. Opt. Soc. Am 70* pp 60-65

Hademard, J. (1923), *Lectures on the Cauchy Problem in Linear Partial Differential Equations* Yale University Press, New Haven, Conn

Hamming, R.W. (1962), *Numerical Methods for Scientists and Engineers* McGraw-Hill, New York

Haralick, R.M. (1981), "The digital edge", *Proceedings of IEEE Conference on Pattern Recognition and Image Processing*, August, pp 285-291

Helmholtz, H. L. F. (1925) *Physiological Optics* (trans) vol 3 Opt Soc Am, Rochester, N. Y. pp 312 ff

Hildebrand, F.B. (1956,1974), *Introduction to Numerical Analysis* McGraw-Hill, New York

Hildreth, E.C. and Ullman, S. (1982), "The measurement of visual motion", *MIT AI Memo No. 699*

Hildreth, E.C. (1984), *The Measurement of Visual Motion*, MIT Press, Cambridge, Mass

Hildreth, E.C. (1984), "Computation of the velocity field", *Proc. Roy. Soc. Lond. B 221* pp 189-220

Hoffman, D.D. (1980) "Inferring shape from motion fields", *MIT AI Memo 592*

Hogg, D.C., Sullivan, G.D., Baker, K.D. and Mott, D.H. (1984), "Recognition of vehicles in traffic scenes using geometric models", *Road Traffic Data Collection, IEE Conference Publication 242* pp 115-119

Horn, B.K.P. and Schunck B.G. (1981), "Determining optical flow", *Artificial Intelligence 17* pp 185-203

Horn, B.K.P. (1986), *Robot Vision* (MIT Press, Cambridge)

Hough, P.V.C. (1962), Method and means for recognising complex patterns," *U.S. Patent 3,069,654*

Hueckel, M.H. (1969), "An operator which locates edges in digital pictures," *Stanford University Science Dept. Memo AIM-105*

Huffman, D.A. (1971), "Impossible objects as nonsense sentences", *Machine Intelligence 6*

Ibison, M.C. and Zapalowski, L. (1985a), "Structure from motion: an alternative approach". *IEEE CVPR 85 p 203*

Ibison, M.C., Zapalowski, L. and Harris, C.G. (1985b), "Surface reconstruction from a moving sensor", *Image and Vision Computing (in press)*

Ibison, M.C., Zapalowski, L. and Harris C.G. (1986), Personal communication

Jain, R., Martin, W.N. and Aggarwal, J.K. (1979) "Segmentation through the detection of changes due to motion", *Computer Graphics and Image Processing 11 (1)* pp 13-34

Jain, R. (1983), "Direct computation of the focus of expansion", *IEEE PAMI-5 (1)* pp 58-64

Jerian, C. and Jain, R. (1984), "Determining motion parameters for scenes with translation and rotation", *IEEE PAMI-6(4)* pp 523-530

Johannson, G. (1973), "Visual perception of biological motion and a model for its analysis", *Perception and Psychophysics 14* pp 201-211

Kanatani, K. (1985), "Structure from motion without correspondence: General principle", *Proc. 9th IJCAI*, pp 886-888

Kitahashi, T. and Hiroyuki, E. (1985), "A new method of 3-D motion analysis using a concept of projective geometry", *Proc 9th IJCAI* pp 902-904

Koenderink, J.J. and van Doorn, A.J. (1975), "Invariant properties of the motion parallax field due to the movement of rigid bodies relative to an observer", *Opt Acta 22* pp 773-791

Koenderink, J.J. and van Doorn, A.J. (1976a) "Local structure of movement parallax of the plane, *J. Opt. Soc. Am. 66* pp 717-723

Koenderink, J.J. and van Doorn, A.J., (1976b), "Geometry of binocular vision and a model for stereopsis", *Biological Cybernetics 21* pp 29-35

Koffka, K. (1935) *Principles of Gestalt Psychology* Harcourt, Brace and World, New York

Kolers, P.A. and Pomerantz, J.R. (1971), "Figural Changes in Apparent Motion", *Journal of Experimental Psychology 87* pp 99-108

Kolers, P.A. (1982), *Aspects of Motion Perception* Oxford: Pergammon Press

Kruskal, J. (1977), "The relationship between Multidimensional Scaling and Clustering", in Van Ryzin, J. (Ed), *Classification and Clustering*, Academic Press (17-44)

Kruskal, J. (1978). *Multidimensional scaling,* (Sage University Paper series on Quantitative Application in the Social Sciences, series 07-011), London: Sage Publications

Lawson, C.L. and Hanson, RJ. (1974), *Solving least-squares problems* Prentice-Hall, New Jersey

Lawton, D.T. (1983), "Processing translational motion sequences", *Comp. Vis. Graph. Image Proc. 22* pp 116-168

Lee, D.N. (1980), "The optic flow field: the foundation of vision", *Proc. Roy. Soc. Land. B290* pp 169-179

Lee, D.N. (1974) "Visual information during locomotion" : in *Perception,* (eds) McLeod and Pick, Cornell University Press, Ithaca pp 250-267

Leese, J.A., Novak, C.S. and Taylor, V.R. (1970),[11]"The determination of cloud pattern motion from geosynchronous satellite image data," *Pattern Recognition Letters 2* pp 279-292

Leeuwenberg E.LJ. and Buffart, H.FJ.M. (1978) (eds), *Formal Theories of Visual Perception,* John Wiley and Sons

Limb, J.O. and Murphy, J.A. (1975), "Estimating the velocity of moving objects from television signals", *Comp. Graph. Image Proc. 4* pp 311-327

Longuet-Higgins, H.C. (1981), "A computer algorithm for reconstructing a scene from two projections". *Nature 293* pp 133-135

Longuet-Higgins, H.C. and Prazdny, K. (1981), "The interpretation of moving retinal images", *Proc. Roy. Soc. Lond. B.* 205 pp 385-397

Longuet-Higgins, H.C. (1982), "The role of the vertical dimension in stereoscopic vision", *Perception 11* pp 377-386

Longuet-Higgins, H.C. (1982), Appendix to Mayhew (1982)

Longuet-Higgins, HLG (1984), "The Visual ambiguity of a moving plane[11], *Proc. Roy. Soc. B223* pp 165-175

Loaguet-HiggiiiS, H.C. (1984), "Hie reconstruction of a seme from two projections - configurations that defeat the 8-pomt algorithm[11], *IEEE Proc. First Conference* on *AI Applications* pp 395-397

Longuet-Higgins, H.C. (1986), "The reconstruction of a plane surface from two perspective projections," *Proc. Roy. Soc. A (in press)*

Longuet-Higgins (1986/87), H.C., *Mental Processes* MIT Press (in press)

Lucas, B.D. and Kanade, T. (1985), "Optical navigation by the method of differences", *Proc. 9th IJCAI* pp 981-984

Marr, D. (1982), *Vision*, W.H.Freeman Co, San Francisco

Marr, D. and Hildreth, E.C. (1980), "Theory of edge detection", *Proc. Roy. Soc. Lond. B. 207* pp 187-217

Marr, D. and Poggio, T. (1979), "A computational theory of human stereo vision", *Proc. Roy. Soc. Lond. B. 204* pp 301-328

Marr, D. and Ullman, S. (1981), "Directional selectivity and its use in early visual processing", *Proc. Roy. Soc. Lond. B211* pp 151-180

Marroquin, J. L. (1984), "Surface reconstruction preserving discontinuities", *MIT AI Memo No 792*

Maybank, S.J. (1984), "The angular velocity associated with the optical flow field due to a single moving rigid plane", *Proc. 6th ECAI* pp 641-644

Maybank, S.J. (1985), "Optical flow and the Taylor expansion", *BPRA, St Andrews*

Maybank, S.J. (1985b), "The angular velocity associated with the optical flowfield arising from motion through a rigid environment", *Proc. Roy. Soc. A401* pp 317-326

Maybank, S.J. (1986), "The apparent area of a rigid moving body," submitted to Alvey Conference (Bristol).

Mayhew, J.E.W. and Frisby, J.P. (1981), "Psychophysical and computational studies towards a theory of human stereopsis", *Artificial Intelligence 17* pp 349-385

Mayhew, J.E.W. (1982), "The interpretation of stereo-disparity information: the computation of surface orientation and depth", *Perception 11*

Mayhew, J.E.W and Longuet-Higgins, H.C. (1982) "A computational model of binocular depth perception", *Nature 297* pp 276-373

Medioni, G. and Nevatia, R. (1984), "Matching images using linear features", *IEEE Trans. PAMI-6 6* pp 675-685

Nagel, H.-H. (1978), "Analysis techniques for image sequences", *Proc. of 4th Intl. Conf. on Pattern Recognition*, Kyoto, Japan

Nagel, H.-H. (1982), "On change detection and displacement vector estimation in image sequences", *Patt. Recog. Letters I* pp 55-59

Nagel, H.-H. (1983a), "On the estimation of dense displacement vector fields from image sequences", *Proc. ACM SIGGRAPH / SIGART Workshop on Motion* Toronto, Ontario pp 59-65

Nagel, H.-H. (1983b), "Displacement vectors derived from second-order intensity variations in image sequences", *CVGIP 21* pp 85-117

Nakayama, K. and Loomis, J.M. (1974), "Optical velocity patterns, velocity sensitive neurones and space perception", *Perception 3* pp 63-80

Navon, D. (1976), "Irrelevance of figural identity for resolving ambiguities in apparent motion", *Journal of Experimental Psychology: Human Perception and Performance 2* pp 130-138

Negahjdaripour, S. and Horn, B.K.P. (1985a) "Direct passive navigation", *MIT AI Memo no. 821 February*

Negahjdaripour, S. and Horn, B.K.P. (1985b), "Determining 3-D motion of planar objects from image brightness patterns", *Proc 9th IJCAI* pp 898-902

Netravali, A.N. and Robbins, J.D. (1979), "Motion-compensated television coding: Part I.", *Bell System Tech. J. 58:* pp 631-670

O'Gorman, F. (1976), "Edge detection using Walsh functions," *Proc AISB, July* pp 195

Poggio, T. and Torre, V. (1984) , "Ill-posed problems and regularisation analysis in early vision", *MIT AI Lab memo no. 773*

Poggio, T. (1985), "Early vision: from computational structure to algorithms and parallel hardware", *CVGIP 31* pp 139-155

Pollard S.B., Mayhew, J.E.W. and Frisby, J.P. (1985), "Disparity gradients and stereo correspondences", Technical Report, Sheffield University

Potter. J.L. (1975), "Velocity as a cue to segmentation", *IEEE Trans. Sys. Man. Cyb. SMC-5* pp 390-394

Potter, J.L. (1977), "Scene segmentation using motion information", *Comp. Graph. Image Proc. 6* pp 558-581

Prazdny, K. (1980), "Egomotion and relative depth map from optical flow", *Bid. Cyb. 36* pp 87-102

Prazdny, K. (1981), "A simple method for recovering a relative depth map in the case of a translating sensor", *Proc. 7th IJCAI* pp 698-699

Prazdny, K. (1985), "Detection of binocular disparities", *Bid. Cybernetics, Vd 52* pp 93-99

Prewitt, J.M.S. (1970), "Object enhancement and extraction," in Lipkin, B.S. and Rosenfeld, A. (eds) *Picutre Processing and Psychopictorics,* Academic Press, New York.

Reichardt. W. (1961), "Autocorrelation, a principle for the evaluation of sensory information by the central nervous system," in Rosenblith. W (ed) *Sensory Communication* (MTT Press, Cambridge)

Reiger, J.H. and Lawton, D.T. (1983), "Relative depth and sensor motion from difference fields of optic flows". *Proceedings of &th IJCAI* pp 1027-1031

Restle, F. (1979), "Coding theory and the perception of motion configurations", *Psychological Review 86* pp 1-24

Rubin, J.M. and Richards. W.A. (1985). "Boundaries of visual motion". *MIT AI Memo Mo. $35*

Schunck, B.G. (1983). *Motion Segmentation mid Estimation,* PhD Dissertation, MIT

Schunck, B.G. (1984), "Motion se^aentatloii and estimation by constraint line clustering", *Proc. Workshop om» Ckmpider Vision,* Annapolis, Maryland pp 58-62

Schunck* B.G. (1985a), "Image flow: Fundamentals and Future Research[18], *Pmc. Conf. Computer Vision and Pattern SmmgnUkm,* San Francisco, June

Schunck, B.G. (1985b) , Image flow continuity equations for motion and density". Genera! Motors Research Laboratories* Warren, Michigan Submitted to Workshop on Motion* 1986

Scott, G.L. (1984), "Obtaining the structure(s) of a non-rigid body from two views by maximising invariance", *Proc. 6th ECAI* p. 658

Scott, G.L. and Buxton, B.F. (1985), "Interpreting heterogenous and under-determined optic flow fields", *Alvey Vision Conference*

Scott, G.L. (1986), "Smoothing the optic flow field under perspective projection", *CVPR, Miami, June*

Scott, G.L. (1983), MA thesis, University of Sussex

Shah, M.A. and Jain, R. (1984), "Detecting time-varying corners", *CVGIP 28* pp 345-355

Shepard, R.N. (1966), "Metric structures in ordinal data", *Journal of Mathematical Psychology 3* pp 287-315

Shepard, R.N. (1980), "Multidimensional scaling, tree-fitting, and clustering", *Science 210* pp 390-98

Spacek, L.A. (1985) *The Detection of Contours and their Visual Motion*, DPhil Dissertation, University of Essex

Stevens, K.A. (1980), "Surface Perception by local analysis of texture and contour", *MIT AI Lab Technical Report 512*

Strat, T. and Fischler, M., "One-eyed stereo," *Proc. 9th IJCAI* pp 937-942

Terzopoulos, D. (1982), "Multilevel reconstruction of visual surfaces: Variational principles and finite element representations", *MIT AI Lab Memo No 671*

Thompson, E.H. (1959), *Photogrammetric Record 3(14)* pp 152-159

Todd, J.T. (1985), "Perception of structure from motion: is projective correspondence a necessary condition?" *Jour. Exp. Psych. 6* pp 689-710

Tsai, R.Y. and Huang, T.S. (1981), "Estimating three dimensional motion parameters of a rigid planar patch", *IEEE ASSP-29 (6)*, pp 1147-1152

Tsai,R.Y., Huang, T.S. and Zhu, W-L (1982), "Estimating three dimensional motion parameters of a rigid planar patch, II: Singular value decomposition", *IEEE ASSP-30 (4)*, pp 525-534

Tsai, R.Y. and Huang, T.S. (1984), "Estimating three-dimensional motion parameters of a rigid planar patch, HI: Finite point correspondences and the three-view problem" *IEEE ASSP-32(2),* pp 213-220

Tsai, R.Y., and Huang, T.S. (1984), Uniqueness and estimation of three-dimensional motion parameters of rigid objects with curved surfaces, *IEEE Vol PAMI-6 I* pp 13-27

Tou, J.T. and Gonzales, R.C. (1974), *Pattern Recognition Principles* Addison-Wesley Publishing Co

Ternus, J. (1967) "Experimentelle Untersuchung uber phanomenale Identitat", Psychologische Forschung, 7, 81-136. Translated in Ellis, WJX, *A Source Book of Gestalt Psychology,* Humanities Press, N.Y

Tikhonov, A.N. (1963), "Solution of incorrectly formulated problems and the regularisation method", *Soviet Math. DokL 4* pp 1035-1038

Tikhonov, A.N. and Arsenin, V.Y. (1977), *Solutions of Mi-posed Problems,* Winston and Sons, Washington D.C.

Ullman, S. (1979), *The Interpretation of Visual Motion,* MIT Press, Cambridge

Ullman, S. (1979), "Relaxation and constrained optimisation by local processes", *Comp. Graph. Image Proc. 9* pp 115-125

Oilman, S. (1983a), "Computational studies in the interpretation of structure and motion: summary and extension", *MIT AI. Lab. Memo 706*

Ullman, S. (1983b), "Maximising rigidity: the incremental recovery of 3-D structure from rigid and rubbery motion", *MIT AJ. Lab. Memi 721*

Wallach, H. (1976), "On perceived identity: 1. The direction of motion of straight lines", in: *On Perception,* H. Wallach (Ed.) Quadrangle, New York

Waxman, A. and Ullman, S. (1984), ^Surface structure and 3-D motion from image flow: A kinematic analysis". Center for Automation Research, University of Maryland

Waxman, AJM and Sinha, S.S. (1984a), "Dynamic stereo: passive ranging to moving objects from relative image flows," Computer Vision Laboratory, University of Maryland

Waxman, A.M. (1984b), "An image flow paradigm" Computer Vision Laboratory, University of Maryland

Waxman, A.M. and Wohn, K. (1984), "Contour evolution, neighbourhood deformation and global image flow: Planar surfaces in motion", Computer Vision Laboratory, University of Maryland

Weiss, I. (1985), "3-D shape representation by contours", *Proc. 9th IJCAI* pp 969-972

Wertheimer, M. (1912), "Experimentelle Studien uber das Sehen von Bewegung", *Zeitshrift fur Psychologie 61* pp 161-265

Wohn, K., Davis, L.S. and Thrift, P. (1983), "Motion estimation based on multiple local constraints and nonlinear smoothing", *Pattern Recognition 16 6* pp 563-570

Xu, G., Tsuji, S. and Asada, M. (1985), "Coarse-to-fine control strategy for matching motion stereo pairs", *Proc. 9th IJCAI* pp 892-894

Yuille, A. (1984), "The smoothest velocity field and token matching schemes", *Proc 6th ECAI* p 621-630