Chapter 4

Comparametric Transforms for Transmitting Eye Tap video with Picture Transfer Protocol (PTP)

Comparametric Transforms for Transmitting Eye Tap video with Picture Transfer Protocol (PTP)

W. Steve G. Mann
University of Toronto
Department of Electrical and Computer Engineering
10 King's College Road, Toronto, Ontario, Canada, M5S 3G4
mann@eecg.toronto.edu
http://wearcam.org

Abstract:

Eye Tap video is a new genre of video imaging facilitated by and for the apparatus of the author's eyeglass–based "wearable computer" invention [1]. This invention gives rise to a new genre of video which is best processed and compressed by way of Comparametric Equations, and Comparametric Image Processing. These new methods are based on an "Edgertonian" philosophy, in sharp departure from the traditional "Nyquist" philosophy of signal processing. A new technique is given for estimating the comparameters (relative parameters between successive frames of an image sequence) taken with a camera (or Eye Tap device) that is free to pan, tilt, rotate about its optical axis, and zoom. This technique solves the problem for two cases of static scenes: images taken from the same location of an arbitrary 3-D scene, or images taken from arbitrary locations of a flat scene, where it is assumed that the gaze pattern of the eye sweeps around on a much faster time scale than the movement of the body (e.g. an assumption that image flow across the retina induced by change in eye location is small compared to that induced by gaze pattern).

4.1 Introduction: Wearable Cybernetics

Wearable cybernetics is based on the WearComp invention of the 1970s, originally intended as a wearable electronic photographer's assistant [2].

4.1.1 Historical overview of WearComp

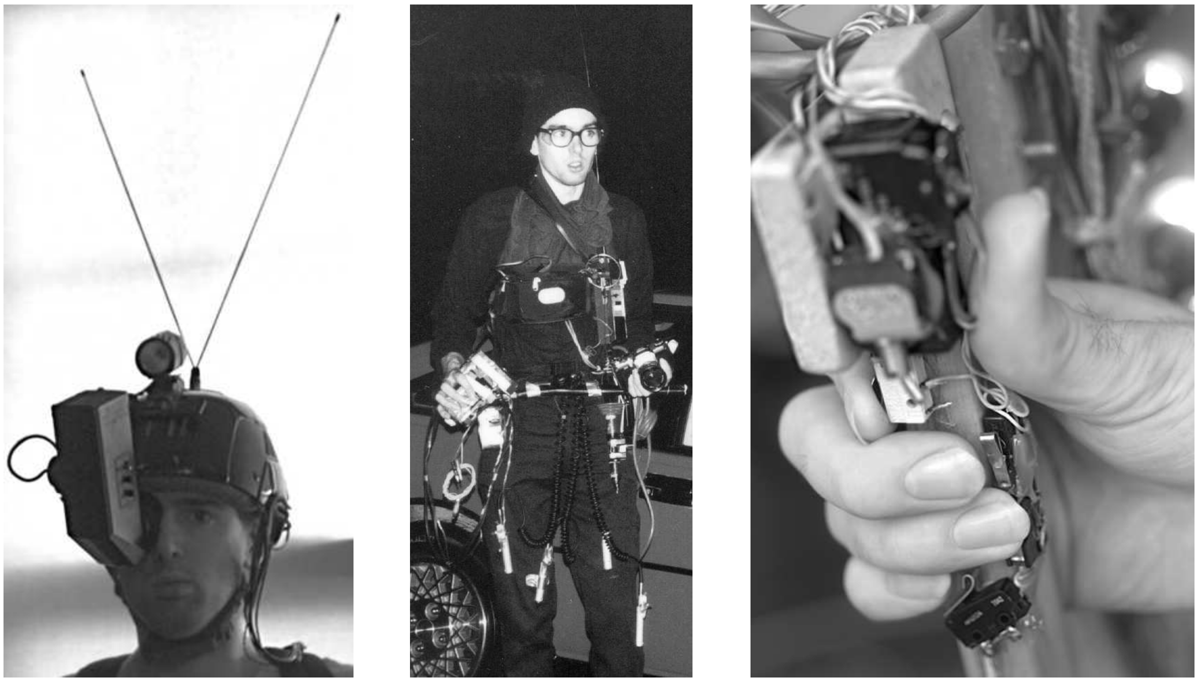
A goal of the author's WearComp/WearCam (wearable computer and Personal Imaging) inventions of 1970s and early 1980s (Fig. 4.1) was to make the metaphors of technology as extensions of the mind and body into a reality. In some sense, these inventions transformed the body into not just a camera, but also a networked cybernetic entity. The body thus became part of a system always seeking the best picture, in all facets of ordinary day–to–day living. These systems served to illustrate the concept of the camera as a true extension of the mind and body of the wearer.

4.1.2 Eye Tap video

Eye Tap video [3] is video captured from the pencil of rays that would otherwise pass through the center of the lens of the eye. The Eye Tap device is typically worn, like eyeglasses.

4.2 The Edgertonian image sequence

Traditional image sequence compression, such as MPEG [4] [5] (see, for example, the Moving Picture Expert Group FAQ), is based on processing frames of video as a continuum. The



(a) (b) (c)

Figure 4.1: *Personal Imaging in the 1970s and 1980s:* Early embodiments of the author's WearComp invention that functioned as a "photographer's assistant" for use in the field of Personal Imaging. (a) Author's early headgear (b) Author's early "smart clothing" including cybernetic jacket and cybernetic pants. (c) Author's 1970s chording keyboard comprising switches mounted to a light source, similar to the mid 1980s version depicted in author's right hand in (b).

integrity of motion is often regarded as being more important than, or at least as important as, the integrity of each individual frame of the image sequence. However, it can be argued that temporal integrity is not always of the utmost importance, and can, in fact, often be sacrificed with good reason.

4.2.1 Edgertonian versus Nyquist thinking

Consider the very typical situation when the frame rate of a picture acquisition process vastly exceeds the frame rate at which it is possible to send pictures of satisfactory quality over a given bandwidth–limited communications channel. This situation arises, for example, with Web based cameras, including the "Wearable Wireless Webcam"[6].

Suppose, for example, that the camera provides 30 pictures per second, but the channel only allows us to send one picture per second (ignore for the moment the fact that we can trade spatial resolution, temporal resolution, and compression quality to adjust the frame rate). In order to downsample our 30 pictures per second down to one picture per second, the "Nyquist school of thought" would suggest that we should temporally low pass filter the image sequence, in order to remove any temporal frequencies that would exceed the Nyquist frequency.

To apply this standard "lowpass filter then downsample" approach, we might average together each 30 successive pictures, to obtain one output picture. Thus fast moving objects would be blurred to prevent temporal aliasing.

We might be tempted to think that this blurring is desirable, given temporal aliasing that would otherwise result. However, cinematographers and others who produce motion pictures, often disregard concepts like temporal aliasing. Most notably, Harold E. Edgerton [7], the inventor of the electronic flash, who is known for his movies of high speed events in which objects are "frozen" in time, has produced movies and other artifacts that defy any avoidance of temporal aliasing. Edgerton's movies provide us with a temporal sampling that is more like a Dirac comb (downsampling of reality) than a lowpass filtered and then downsampled version of reality.

Back to the example of downsamping from 30 frames per second to one frame per second, an Edgertonian thinker would likely advocate simply taking every 30th frame from the original sequence, and throwing all the others away.

The Edgertonian downsampling philosophy gives rise to image sequences in which propeller blades or wagon wheel spokes appear to spin backwards or stand still. The Nyquist philosophy gives rise to image sequences in which the propeller blades or wagon wheel spokes would visually disappear.

The author believes that it is preferable that the propeller blades and wagon wheel spokes appear to spin backwards, or stand still, rather than visually disappear. More generally, an important assumption upon which the thesis of this chapter rests, is that it is preferable to have a series of crisp well–defined "snapshots" of reality, rather than the blur of images that one would get by following the antialiasing approach of traditional signal processing.

The author's personal experience with his wearable Eye Tap video camera invention, wearing the camera often 8 to 16 hours a day, led to a certain understanding of how the world looks through Web–based video. On this system, it was possible to choose from among various combinations of Edgertonian and Nyquist sampling strategies. It was found that experiencing the world through "Edgertonian eyes" was generally preferable to the Nyquist approach.

4.2.2 Frames versus rows, columns, and pixels

There is a trend now toward processing sequences of images as spatio-temporal volumes, e.g. as a function $f(x, y, t)$. Within this conceptual framework, motion pictures are treated as static $3d$ volumes of data. So–called *spatio-temporal filters* $h(x, y, t)$ are applied to these spatio-temporal volumes $f(x, y, t)$.

However, this unified treatment of the three dimensions (discretized to row, column, and frame number) ignores the fact that the time dimension has a much different intuitive meaning than the other two. Apart from the progressive (forward–only) direction of time, there is the more important fact (even for stored image sequences) that a snapshot in time (e.g. a still picture selected from the sequence) often has immediate meaning to the human observer. A single row of pixels across a picture, or a single column of pixels down a picture do not generally have similar significance to the human observer. Likewise a single pixel means little to the human observer in the absence of surrounding pixels.

Notwithstanding their utility, slices of the form $f(x, t)$ or of the form $f(y, t)$ are often confusing at best, compared to the still picture $f(x, y)$ which remains as an extraction from a picture sequence that is far more meaningful to a typical human observer.

Thus the author believes that downsampling across rows or downsampling down columns of an image should be preceded by lowpass filtering, whereas temporal downsampling should not.

There is therefore, a special significance to the notion of a "snapshot in time" and the processing, storage, transmission, etc., of a motion picture as a sequence of such "snapshots". The object of this chapter is to better understand the relationship between individual sharply defined frames of an "Edgertonian" sequence of pictures.

4.3 Picture Transfer Protocol (PTP)

When applying data compression to a stream of individual pictures that will be viewed in real-time (for example in videoconferencing, such as the first–person–perspective videoconferencing of the wearable Eye Tap device), it is helpful to take into account the manner in which the data will be sent.

Most notably, pictures are typically sent over a packet–based communications channel. For example, Wearable Wireless Webcam used the AX25 Amateur Radio [8] protocol. Accordingly, packets typically either arrive intact or are corrupt. Packets that are corrupt, traditionally would be re–sent.

An interesting approach is to provide data compression on a per–image basis, and to vary the degree of compression so that the size of each picture in the image sequence is exactly equal to the length of one packet.

Together with the prior assumption (that images are acquired at a rate that exceeds the channel capacity), it will generally be true that by the time we know that a packet (which is a complete picture) is corrupt at the receiver, there will have already been acquired a newer picture than the one that was sent. Suppose, for example, the round trip time (RTT) is 100 ms (which is equal to the time it takes to generate three pictures), there would be little sense in re–sending a picture that was taken three pictures ago.

The commonly arising situation in which pictures are captured at a rate that exceeds the RTT suggests that there will always be newer picture information at the transmit site than what would be re–sent in the event of a lost packet.

This approach forms the basis for the "Picture Transfer Protocol" (PTP) proposed by the author. In particular, PTP is based on the idea of treating each "snapshot" in time as a single entity, in isolation, and compressing it into a single packet, so that it will have either arrived in its entirity or not arrived at all (and therefore can be discarded). It should be clear that the philisophical underpinnings of PTP are closely related to those of Edgertonian downsampling.

4.4 Best case imaging and fear of functionality

A direct result of Edgertonian sampling is that a single picture from a picture sequence has a high degree of relevance and meaning even when it is taken in isolation.

Similarly, a direct result of PTP is that a single packet from a packet sequence has a high degree of relevance and meaning even when it is taken in isolation (for example when the packets before and after it have been corrupted).

It is therefore apparent that if a system were highly unreliable, to the extent that pictures could only be transmitted occasionally, and unpredictably, then the Edgertonian sampling combined with PTP would provide a system that would degrade gracefully.

Indeed, if we were to randomly select just a few frames from one of Harold Edgerton's motion pictures, we would likely have a good summary of the motion picture, since any given frame would provide us with a sharp picture in which subject matter of interest could be clearly discerned. Likewise, if we were to randomly select a few packets from a stream of thousands of packets of PTP, we would have data that would provide a much more meaningful interpretation to the human observer than if all we had was randomly selected packets from an MPEG sequence.

Personal Imaging systems are characterized by a wearable incidentalist “always ready” mode of operation in which the system need not always be functioning to be of benefit. It is the *potential* functionality, rather than the actual functionality of such a system that makes it so different from other imaging systems like hand held cameras and the like.

Accordingly, an object of the Personal Imaging project is to provide a system that transmits pictures in harsh or hostile environments. One such application of such a system is the Personal Safety Device described in [9].

The Personal Safety Device differs from other wireless data transmission systems in the sense that it was designed for “Best Case” operation. Ordinarily, wireless transmissions are designed for worst case scenarios, such as might guarantee a certain minimum level of performance throughout a large metropolitan area. The Personal Safety Device, however, is designed to make it hard for an adversary to guarantee total nonperformance.

It is not a goal of the Personal Safety Device to guarantee connectivity in the presence of hostile jamming of the radio spectrum, but, rather, the goal is to make it difficult for the adversary to guarantee the absence of connectivity. Therefore, an otherwise potential perpetrator of a crime would never be able to be certain that the wearer’s device was non–operational and would therefore need to be on his or her best behavior at all times.

Traditional surveillance networks, based on so–called “public safety” camera systems have been proposed to reduce the allegedly rising levels of crime. However, building such surveillance superhighways may do little to prevent, for example, crime by representatives of the surveillance state, or those who maintain the database of images. Human rights violations can continue, or even increase, in a police state of total state surveillance. The same can be true of owners of an establishment where surveillance systems are installed and maintained by these establishment owners. Examples such as the famous Latasha Harlins case in which a shopper, falsely accused of shoplifting by a shopkeeper, was shot dead by the shopkeeper. Therefore what is needed is a Personal Safety Device, to function as a crime deterrent, particular with regards to crimes perpetrated by those further up the organizational hierarchy.

Since there is, and would be, the possibility of just one packet, which contains just one picture, providing incriminating evidence of wrongdoing, an individual can simply wear a personal safety device to obtain protection from criminals, assailants, and attackers, notwithstanding any alleged public or corporate video surveillance system already in place.

An important aspect of this paradigm is the Fear of Functionality (FoF) model. The balance is typically tipped in favour of the state or large organization, in the sense that state owned, or corporate owned surveillance cameras are typically mounted on fixed mount points, and networked by way of high bandwidth land lines. The Personal Safety Device (PSD), on the other hand, would be connected by way of wireless communication channels of limited bandwidth and limited reliability. For example, in the basement of a department store, the individual has a lesser chance of getting a good reliable data connection than the store–owned surveillance cameras. Just as many department stores use a mixture of fake nonfunctional cameras and real ones, so that the customer never knows whether or not a given camera is operational, what is needed is a similar means of best case video transmission. Not knowing whether or not one is being held accountable for one's actions, one must be on one's best behavior at all times. Thus a new philosophy, based on FoF, can become the basis of design for image compression, transmission, and representation.

Fig 4.2(a) illustrates an example of a comparison between two systems, SYSTEM "A", and SYSTEM "B". These systems are depicted as two plots, in a hypothetical parameter space. The parameter space could be time, position, or the like. For example, SYSTEM "A" might work acceptably (e.g. meet a certain guaranteed degree of functionality F_{GUAR}) everywhere at all times, whereas SYSTEM "B" might work really well sometimes, and poorly at other times. Much of engineering is motivated by an *articulability* model, e.g. that one can make an articulable basis for choosing SYSTEM "A" because it gives the higher Worst Case degree of functionality.

A new approach, however, reverses this argument, by regarding functionality as a bad thing — bad for the perpetrator of a crime — rather than a good thing. Thus we turn the whole graph on its head, and looking at the problem in this reversed light, come to a new solution, namely that SYSTEM "B" is better because there are times when it works really well.

Imagine, for example, a user in the sub basement of a building, inside an elevator. Suppose SYSTEM "A" would have no hope of connecting to the outside world. SYSTEM "B", however, through some strange quirk of luck, might actually work, but we don't know one way or the other, in advance.

The fact of the matter is, however, that one who was hoping that the system would no function, would be more afraid of SYSTEM "B", than SYSTEM "A", because it would take more effort to ensure that SYSTEM "B" would be nonfunctional.

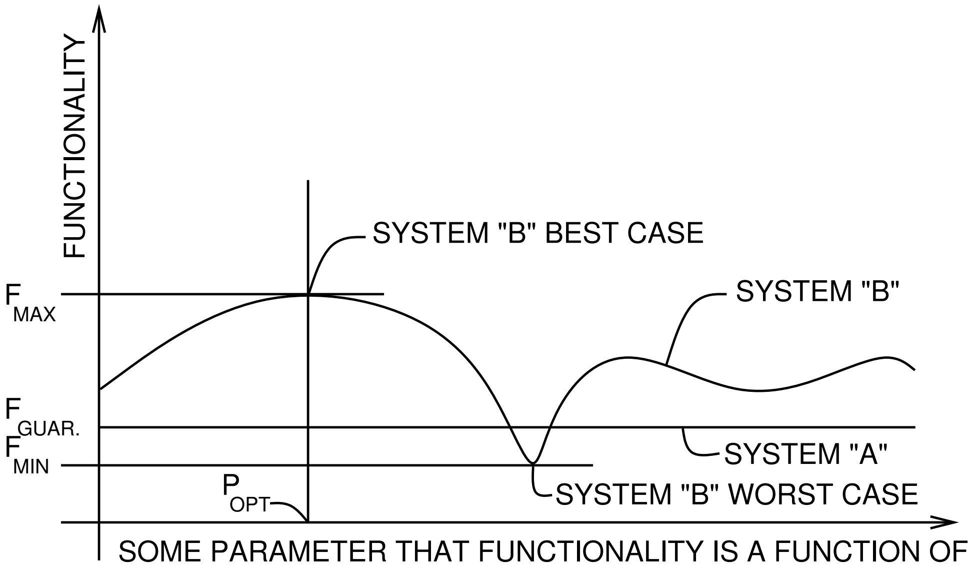
The Fear of Functionality (FoF) model means that if there exists the possibility that the system might function part of the time, so that a would-be perpetrator of a crime against the wearer of the personal safety device must be on his or her best behavior at all times.

Fig 4.2(b) depicts what we might do to further improve the "fear factor" of our SYSTEM "B", to arrive at a new SYSTEM "$\tilde{B}$". The new SYSTEM "$\tilde{B}$" is characterized by making it even more idiosyncratic, such that the occasional time that SYSTEM "$\tilde{B}$" works, it works really well, but most of the time it doesn't work at all, or works very poorly.

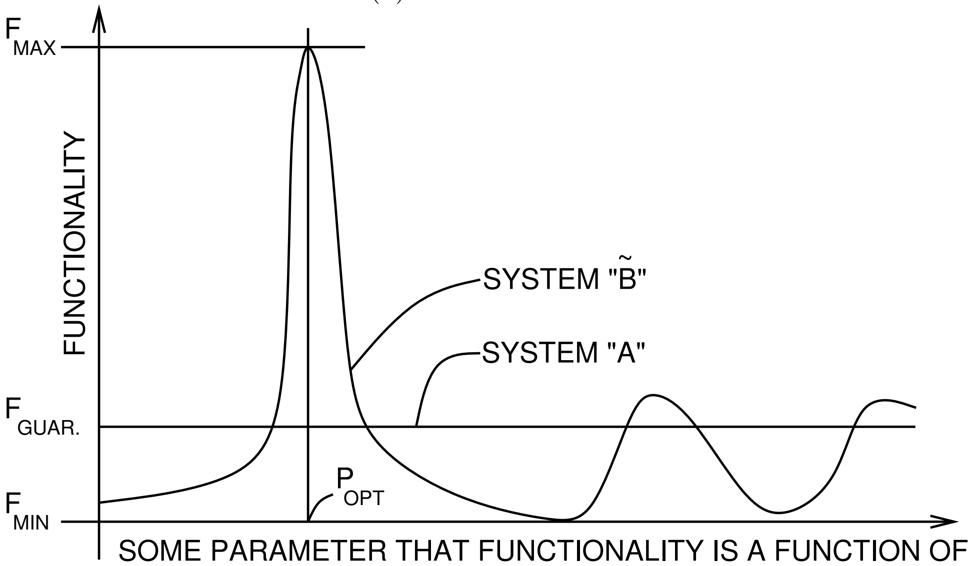
Other technologies, such as the Internet, have been constructed to be robust enough to resist the hegemony of central authority (or an attack of war). However, an important difference here is that the FoF paradigm is not suggesting the design of *robust* data compression and transmission networks.

Quite the opposite is true!

The FoF paradigm suggests the opposite of robustness, in the sense that



(a)



(b)

Figure 4.2: **Fear of Functionality (FoF)**: (a) Given two different systems, SYSTEM "A" having a guaranteed minimum level of functionality F_{GUAR} that exceeds that of SYSTEM "B", an articulable basis for selecting SYSTEM "A" can be made. Such an articulable basis might appeal to lawyers, insurance agents, and others who are in the business of guaranteeing easily defined articulable boundaries. However, a thesis of this book chapter is that SYSTEM "B" might be a better choice. Moreover, given that we are designing and building a system like SYSTEM "B", traditional Worst Case engineering would suggest focusing on the lowest point of functionality of SYSTEM "B". (b) Instead, it is proposed that one might focus one's efforts on the highest point of functionality of SYSTEM "B", to make it even higher, at the expense of further degrading the SYSTEM "B" WORST CASE, and even at the expense of decreasing the overall average performance. The new SYSTEM "$\tilde{B}$" is thus sharply serindipitous (e.g. peaked in its space of various system parameters.

SYSTEM "$\tilde{B}$" is even more sensitive to mild perturbations in the parameter space about the optimal operating point, P_{OPT}, than SYSTEM "B". In this sense, our preferred SYSTEM "$\tilde{B}$" is actually much less robust than SYSTEM "B". Clearly it is not robustness, in and of itself, that the author is proposing here.

The personal safety device need not work constantly, but, rather, must simply present criminals with the possibility that it could work sometimes, or even just occasionally. This scenario forms the basis for best-case design as an alternative to the usual worst-case design paradigm.

The Personal Imaging system therefore transmits video, but the design of the system is such that it will, at the very least, occasionally transmit a meaningful still image. Likewise the philosophy for data compression and transforms needs to be completely re–thought for this FoF model.

This rethinking extends from the transforms and compression approach right down to the physical hardware. For example, typically the wearer's jacket functions as a large low frequency antenna, providing transmission capability in a frequency band that is very hard to stop. For example, the 10 meter band is a good choice because of its unpredictable performance (owing to various "skip" phenomena, etc.). However, other frequencies are also used in parallel.

For example, a peer to peer form of infrared communication is also included, to "infect" other participants with the possibility of having received an image. In this way, it becomes nearly impossible for a police state to suppress the signal, because of the *possibility* that an image may have escaped an iron–fisted regime.

It is not necessary to have a large aggregate bandwidth to support an FoF network. In fact, quite the opposite.

Since it is not necessary that everyone transmit everything they see, at all times, very little bandwidth is needed. It is only necessary that anyone *could* transmit a picture at any time. This potential transmission (e.g. fear of transmission) need not even be to the Internet, e.g. it could simply be from one person to another to another person.

4.5 Comparametric Image Sequence Analysis

Video sequences from the Personal Safety Device are generally collected and assembled into a small number of still images, each still image being robust to the presence or absence of individual constituent frames of the video sequence from which it is composed.

Processing video sequences from the apparatus of the author's Eye Tap camera invention requires finding the coordinate transformation between two images of the same scene or object. Whether to recover gaze motion between video frames, to stabilize retinal images, to relate or recognize Eye Tap images taken from two different eyes, to compute depth within a 3-D scene, or to align images for lookpaintings (high-resolution enhancement resulting

Model	Coordinate transformation from $\mathbf{x}$ to $\tilde{\mathbf{x}}$	Parameters
Translation	$\tilde{\mathbf{x}} = \mathbf{x} + \mathbf{b}$	$\mathbf{b} \in \mathbb{R}^2$
Affine	$\tilde{\mathbf{x}} = \mathbf{A}\mathbf{x} + \mathbf{b}$	$\mathbf{A} \in \mathbb{R}^{2\times 2}, \mathbf{b} \in \mathbb{R}^2$
Bilinear	$\tilde{x} = q_{\tilde{x}xy}xy + q_{\tilde{x}x}x + q_{\tilde{x}y}y + q_{\tilde{x}}$ $\tilde{y} = q_{\tilde{y}xy}xy + q_{\tilde{y}x}x + q_{\tilde{y}y}y + q_{\tilde{y}}$	$q_* \in \mathbb{R}$
Projective	$\tilde{\mathbf{x}} = \frac{\mathbf{A}\mathbf{x}+\mathbf{b}}{\mathbf{c}^T\mathbf{x}+1}$	$\mathbf{A} \in \mathbb{R}^{2\times 2}, \mathbf{b}, \mathbf{c} \in \mathbb{R}^2$
Pseudoperspective	$\tilde{x} = q_{\tilde{x}x}x + q_{\tilde{x}y}y + q_{\tilde{x}} + q_{\alpha}x^2 + q_{\beta}xy$ $\tilde{y} = q_{\tilde{y}x}x + q_{\tilde{y}y}y + q_{\tilde{y}} + q_{\alpha}xy + q_{\beta}y^2$	$q_* \in \mathbb{R}$
Biquadratic	$\tilde{x} = q_{\tilde{x}x^2}x^2 + q_{\tilde{x}xy}xy + q_{\tilde{x}y^2}y^2 + q_{\tilde{x}x}x + q_{\tilde{x}y}y + q_{\tilde{x}}$ $\tilde{y} = q_{\tilde{y}x^2}x^2 + q_{\tilde{y}xy}xy + q_{\tilde{y}y^2}y^2 + q_{\tilde{y}x}x + q_{\tilde{y}y}y + q_{\tilde{y}}$	$q_* \in \mathbb{R}$

Table 4.1: Image coordinate transformations discussed in this chapter: The Translation, Affine, and Projective models are expressed in vector form, e.g. $\mathbf{x} = [x, y]^T$ is a vector of length 2, and $\mathbf{A} \in \mathbb{R}^{2x2}$ is a matrix of dimension 2 by 2, etc..

from looking around), it is desired to have both a precise description of the coordinate transformation between a pair of Eye Tap video frames, and some indication as to its accuracy.

Traditional *block matching* [10] (e.g. as used in *motion estimation*) is really a special case of a more general *coordinate transformation*. In this chapter a solution to the *motion estimation* problem using this more general estimation of a coordinate transformation, is proposed, together with a technique for automatically finding the comparametric projective coordinate transformation that relates two frames taken of the same static scene.

The technique takes two frames as input, and automatically outputs the comparameters of the exact model, to align the frames. It does not require the tracking or correspondence of explicit features, and yet is computationally practical.

Although the theory presented makes the typical assumptions of static scene and no parallax, the estimation technique is robust to deviations from these assumptions. In particular, the technique is applied to image resolution enhancement and lookpainting [11], illustrating its success on a variety of practical and difficult cases, including some that violate the non-parallax and static scene assumptions.

A coordinate transformation maps the image coordinates, $\mathbf{x} = [x, y]^T$ to a new set of coordinates, $\tilde{\mathbf{x}} = [\tilde{x}, \tilde{y}]^T$. Generally, the approach to "finding the coordinate transformation" relies on assuming it will take one of the models in Table 4.1, and then estimating the two to twelve scalar parameters of the chosen model. An illustration showing the effects possible with each of these models is given in Fig. 4.3.

The most common assumption (especially in motion estimation for coding, and optical flow for computer vision) is that the coordinate transformation between frames is a translation. Tekalp, Ozkan, and Sezan [12] have applied this assumption to high-resolution image reconstruction. Although translation is the least constraining and simplest to implement of the six coordinate transformations in Table 4.1, it is poor at handling large changes due to camera zoom, rotation, pan and tilt.

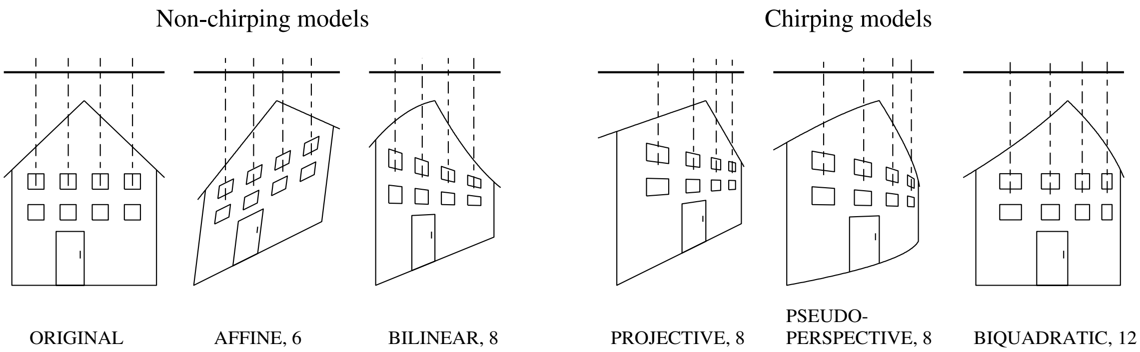


Figure 4.3: Pictorial effects of the six coordinate transformations of Table 4.1, arranged left to right by number of parameters. Note that translation leaves the ORIGINAL house unchanged, except in its location. Most importantly, only the three coordinate transformations at the right affect the periodicity of the window spacing (e.g. induce the desired "chirping" which corresponds to what we see in the real world.) Of these, only the PROJECTIVE coordinate transformation preserves straight lines. The 8-parameter PROJECTIVE coordinate transformation "exactly" describes the possible camera motions.

Zheng and Chellappa [13] considered a subset of the affine model — translation, rotation and scale — in image registration. Other researchers [14][15] have assumed affine motion (six parameters) between frames. For the assumptions of static scene and no parallax, the affine model exactly describes rotation about the optical axis of the camera, zoom of the camera, and pure shear, which the camera does not do, except in the limit as the lens focal length approaches infinity. The affine model cannot capture camera pan and tilt, and therefore cannot accurately express the "chirping" and "keystoning" we see in the real world (see Fig. 4.3). Consequently, the affine model tries to fit the wrong parameters to these effects. When the parameter estimation is not done properly to align the images, then a greater burden is placed on designing post-processing to enhance the poorly aligned images.

The 8-parameter *projective* model gives the exact eight desired parameters to account for all the possible camera motions. However, its parameters have traditionally been mathematically and computationally too hard to find. Consequently, a variety of approximations have been proposed. Before the solution to estimating the projective parameters is presented, it will be helpful to better understand these approximate models.

Going from first order (affine), to second order, gives the 12-parameter 'biquadratic' model. This model properly captures both the chirping (change in spatial frequency with position) and converging lines (keystoning) effects associated with projective coordinate transformations, although, despite its larger number of parameters, there is still considerable discrepancy between a projective coordinate transformation and the best-fit biquadratic coordinate transformation. Why stop at 2nd order? Why not use a 20-parameter 'bicubic' model? While an increase in the number of model parameters will result in a better fit, there is a tradeoff, where the model begins to fit noise. The physical camera model fits exactly in the 8-parameter projective group; therefore, we know that "eight is enough." Hence, it is appealing to find an approximate model with only eight parameters.

The 8-parameter bilinear model is perhaps the most widely-used [16] in the fields of image processing, medical imaging, remote sensing, and computer graphics. This model is easily obtained from the biquadratic model by removing the four x^2 and y^2 terms. Although the resulting bilinear model captures the effect of converging lines, it completely fails to capture the effect of chirping.

The 8-parameter *pseudo-perspective* model [17] does, in fact, capture both the converging lines and the chirping of a projective coordinate transformation. This model may be thought of as first, removal of two of the quadratic terms ($\mathbf{q}_{\tilde{x}y^2} = \mathbf{q}_{\tilde{y}x^2} = 0$), which results in a ten parameter model (the 'q-chirp' of [18]) and then constraining the four remaining quadratic parameters to have two degrees of freedom. These constraints force the "chirping effect" (captured by $\mathbf{q}_{\tilde{x}x^2}$ and $\mathbf{q}_{\tilde{y}y^2}$) and the "converging effect" (captured by $\mathbf{q}_{\tilde{x}xy}$ and $\mathbf{q}_{\tilde{y}xy}$) to work together in the "right" way to match, as closely as possible, the effect of a projective coordinate transformation. By setting $\mathbf{q}_\alpha = \mathbf{q}_{\tilde{x}x^2} = \mathbf{q}_{\tilde{y}xy}$, the chirping in the x-direction is forced to correspond with the converging of parallel lines in the x-direction (and likewise for the y-direction). Therefore, of the 8-parameter approximations to the true projective, we would expect the *pseudo-perspective* model to perform the best.

Of course, the desired "exact" eight parameters come from the projective model, but they have been notoriously difficult to estimate. The parameters for this model have been solved by Tsai and Huang [19], but their solution assumed that features had been identified in the two frames, along with their correspondences. In this chapter, a simple featureless means of registering images by estimating their comparameters is presented.

Other researchers have looked at projective estimation in the context of obtaining $3-D$ models. Faugeras and Lustman [20], Shashua and Navab [21], and Sawhney [22] have considered the problem of estimating the projective parameters while computing the motion of a rigid planar patch, as part of a larger problem of finding 3-D motion and structure using parallax relative to an arbitrary plane in the scene. Kumar *et al.* [23] have also suggested registering frames of video by computing the flow along the *epipolar* lines, for which there is also an initial step of calculating the gross camera movement assuming no parallax. However, these methods have relied on feature correspondences, and were aimed at 3-D scene modeling. Our focus is not on recovering the 3-D scene model, but on aligning 2-D images of 3-D scenes. Feature correspondences greatly simplify the problem; however, they also have many problems which are reviewed below. The focus of this chapter is a simple featureless approach to estimating the projective coordinate transformation between image frames.

Two similar efforts exist to the new work presented here. Mann [24], and Szeliski and Coughlan [25] independently proposed featureless registration and compositing of either pictures of a nearly flat object, or of pictures taken from approximately the same location. Both used a 2-D projective model, and searched over its 8-parameter space to minimize the mean-square error (or maximize the inner product) between one frame and a 2-D projective coordinate transformation of the next frame. However, in both these earlier works the algorithm relies on nonlinear optimization techniques which we are able to avoid with the new technique presented here.

4.5.1 Camera, eye, or head motion: common assumptions and terminology

Two assumptions are relevant to this work. The first assumption is that the scene is relatively constant – changes of scene content and lighting are small between frames, relative to changes that are induced by camera, eye, or head motion (e.g. a person can turn his or her head, hence turning an Eye Tap camera, and induce a much greater image flowfield than that induced by movement of objects in the scene). The second assumption is that of an ideal pinhole camera – implying unlimited depth of field with everything in focus (infinite resolution) and implying that straight lines map to straight lines[1]. This assumption is particularly valid for laser Eye Tap cameras which actually do have infinite depth of focus. Consequently, the camera, eye, or head has three degrees of freedom in 2-D space and eight degrees of freedom in 3-D space: translation (X, Y, Z), zoom (scale in each of the image coordinates x and y), and rotation (rotation about the optical axis, pan, and tilt.

In this chapter, an "uncalibrated camera" refers to one in which the principal point[2] is not necessarily at the center (origin) of the image and the scale is not necessarily isotropic. It is assumed that the film, sensor, retina, or the like, is flat (although we know in fact that the retina is curved).

It is assumed that the zoom is continually adjustable by the camera user, and that we do not know the zoom setting, or whether it changed between recording frames of the image sequence. We also assume that each element in the camera sensor array returns a quantity that is linearly proportional to the quantity of light received[3].

4.5.2 VideoOrbits

Tsai and Huang [19] pointed out that the elements of the projective *group* give the true camera motions with respect to a planar surface. They explored the group structure associated with images of a 3-D rigid planar patch, as well as the associated *Lie algebra*, although they assume that the correspondence problem has been solved. The solution presented in this chapter (which does not require prior solution of correspondence) also relies on projective group theory. We briefly review the basics of this theory, before presenting the new solution in the next section.

[1] When using low cost wide-angle lenses, there is usually some barrel distortion which we correct using the method of [26].

[2] The principal point is where the optical axis intersects the film, retina, sensor, or the like, as the case may be.

[3] This condition can be enforced over a wide range of light intensity levels, by using the Wyckoff principle [27][28].

Projective group in 1-D coordinates

For simplicity, the theory is first reviewed for the projective coordinate transformation in one dimension[4]: $\tilde{x} = (ax + b)/(cx + 1)$, where the images are functions of one variable, x. The set of all projective coordinate transformations for which $a \neq 0$ forms a group, $\mathbf{P}$ the *projective group*. When $a \neq 0$ and $c = 0$, it is the affine group. When $a = 1$ and $c = 0$, it becomes the translation group.

Of the six coordinate transformations in the previous section, only the projective, affine, and translation operations form groups. A group of operators together with the set of 1-D images (operands) form a *group operation*[5]. The new set of images that results from applying all possible operators from the group to a particular image from the original set is called the *orbit* of that image under the group operation [29].

A camera at a fixed location, and free to zoom and pan, gives rise to a resulting pair of (1-D) frames taken by the camera, which are related by the coordinate transformation from x_1 to x_2, given by [30]:

$$\begin{aligned} x_2 &= z_2 \tan(\arctan(x_1/z_1) - \theta), \;\; \forall x_1 \neq o_1 \\ &= (ax_1 + b)/(cx_1 + 1), \;\; \forall x_1 \neq o_1 \end{aligned} \tag{4.1}$$

where $a = z_2/z_1$, $b = -z_2 \tan(\theta)$, $c = \tan(\theta)/z_1$, and $o_1 = z_1 \tan(\pi/2 + \theta) = -1/c$, is the location of the singularity in the domain. We should emphasize that c, the degree of perspective, has been given the interpretation of a **c**hirp-rate [30]. The coordinate transformations of (4.1) form a group operation. This result, and the proof of this group's isomorphism to the group corresponding to nonsingular projections of a flat object are given in [31].

Projective group in 2-D coordinates

The theory for the projective, affine, and translation groups also holds for the familiar 2-D images taken of the 3-D world. The 'video orbit' of a given 2-D frame is defined to be the set of all images that can be produced by applying operators from the 2-D projective group to the given image. Hence, we restate the coordinate transformation problem: Given a set of images that lie in the same orbit of the group, we wish to find for each image pair, that operator in the group which takes one image to the other image.

If two frames, say, f_1 and f_2, are in the same orbit, then there is a group operation $\mathbf{p}$ such that the mean-squared error (MSE) between f_1 and $f_2' = \mathbf{p} \circ f_2$ is zero, where the symbol $\circ$ denotes the operation of $\mathbf{p}$ acting on frame f_2. In practice, however, we find which element of the group takes one image "nearest" the other, for there will be a certain amount of parallax, noise, interpolation error, edge effects, changes in lighting, depth of focus, etc. Fig. 4.4 illustrates the operator $\mathbf{p}$ acting on frame f_2, to move it nearest to frame f_1. (This

[4] In a 2-D world, the "camera" consists of a center of projection (pinhole "lens") and a line (1-D sensor array or 1-D "film").

[5] also known as a *group action* or *G-set* [29].

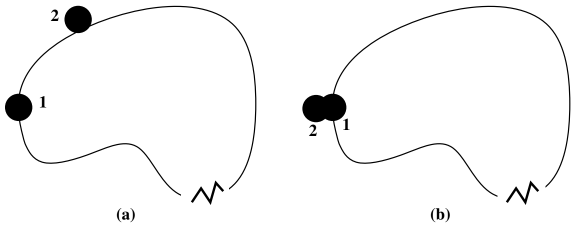


Figure 4.4: Video orbits. (a) The orbit of frame 1 is the set of all images that can be produced by acting on frame 1 with any element of the operator group. Assuming that frames 1 and 2 are from the same scene, frame 2 will be close to one of the possible projective coordinate transformations of frame 1. In other words, frame 2 "lies near the orbit of" frame 1. (b) By bringing frame 2 along its orbit, (which is nearly the same orbit as the orbit of frame 1), we can determine how closely the two orbits come together at frame 1.

figure does not, however, reveal the precise shape of the orbit, which occupies an 8-D space.)

The primary assumptions in these cases are that of no parallax, and of a static scene. Because the 8-parameter projective model is "exact," it is theoretically the right model to use for estimating the coordinate transformation. The examples which follow demonstrate that it also performs better in practice than the other proposed models. In the next section, a new technique for estimating its eight parameters is shown.

4.6 Framework: comparameter estimation and optical flow

Before the new results are presented, existing methods of comparameter estimation for coordinate transformations are reviewed. By comparameters, what is meant is the relative parameters that transform one image into the other, between a pair of images from an image sequence. Estimation of comparameters in a pairwise fashion can be dealt with globally based on the group properties, assuming the parameters in question trace out an orbit of a group.

We break existing methods into two categories: feature-based, and featureless. Of the featureless methods, consider two subcategories: 1) methods based on minimizing MSE (generalized correlation, direct nonlinear optimization) and 2) methods based on spatio-temporal derivatives and optical flow. Note that variations such as *multiscale* have been omitted from these categories; multiscale analysis can be applied to any of them. The new algorithm developed in this chapter (with final form given in Sec. 4.7) is featureless, and is based on multiscale spatio-temporal derivatives.

Some of the descriptions below will be presented for hypothetical 1-D images taken in a 2-D space. This simplification yields a clearer comparison of the estimation methods. The new theory and applications will be presented subsequently for 2-D images taken in a 3-D space.

4.6.1 Feature-based methods

Feature-based methods [32][33] assume that point correspondences in both images are available. In the projective case, given at least three correspondences between point pairs in the two 1-D images, we will find the element, $\mathbf{p} = \{a, b, c\} \in \mathbf{P}$ that maps the second image into the first. Let $x_k, k = 1, 2, 3, \ldots$ be the points in one image, and let $\tilde{x}_k$ be the corresponding points in the other image. Then: $\tilde{x}_k = (ax_k + b)/(cx_k + 1)$. Re-arranging yields $ax_k + b - x_k\tilde{x}_k c = \tilde{x}_k$, so that a, b, and c can be found by solving $k \geq 3$ linear equations in 3 unknowns:

$$\left[\begin{array}{ccc} x_k & 1 & -\tilde{x}_k x_k \end{array}\right] \left[\begin{array}{ccc} a & b & c \end{array}\right]^T = \left[\begin{array}{c} \tilde{x}_k \end{array}\right] \tag{4.2}$$

using least squares if there are more than three correspondence points. The extension from 1-D images to 2-D images is conceptually identical; for the affine and projective models, the minimum number of correspondence points needed in 2-D is three and four respectively.

A major difficulty with feature-based methods is finding the features. Good features are often hand-selected, or computed, possibly with some degree of human intervention [34]. A second problem with features is their sensitivity to noise and occlusion. Even if reliable features exist between frames, these features may be subject to signal noise and occlusion. The emphasis in the rest of this chapter will be on robust featureless methods.

4.6.2 Featureless methods based on generalized cross-correlation

Cross-correlation of two frames is a featureless method of recovering translation model comparameters. Affine and projective comparameters can also be recovered using generalized forms of cross-correlation between two images (e.g. comparing two images using cross correlation and related methods).

Generalized cross-correlation is based on an inner-product formulation which establishes a similarity metric between two functions, say, g and h, where $h \approx \mathbf{p} \circ g$ is an approximately coordinate-transformed version of g, but the comparameters of the coordinate transformation, $\mathbf{p}$ are unknown.[6] We can find, by exhaustive search (applying all possible operators, $\mathbf{p}$, to h), the "best" $\mathbf{p}$ as the one which maximizes the inner product:

$$\int_{-\infty}^{\infty} g(x) \frac{\mathbf{p}^{-1} \circ h(x)}{\int_{-\infty}^{\infty} \mathbf{p}^{-1} \circ h(x)dx} dx \tag{4.3}$$

where we have normalized the energy of each coordinate-transformed h before making the comparison. Equivalently, instead of maximizing a similarity metric, we can minimize some anti–similarity metric, such as MSE, given by $\int_{-\infty}^{\infty} \left(g(x) - \mathbf{p}^{-1} \circ h(x)\right)^2 dx$. Solving (4.3) has an advantage over finding MSE when one image is not only a coordinate-transformed version of the other, but is also an amplitude-scaled version, as generally happens when there is an automatic gain control or an automatic iris in the camera.

[6] In the presence of additive white Gaussian noise, this method, also known as "matched filtering", leads to a maximum likelihood estimate of the parameters [35].

In 1-D, the affine model permits only dilation and translation. Given h, an affine coordinate-transformed version of g, generalized correlation amounts to estimating the parameters for dilation, a and translation, b by exhaustive search. The collection of all possible coordinate transformations, when applied to one of the images (say, h) serves to produce a family of templates to which the other image, g, can be compared. If we normalize each template so that all have the same energy:

$$h_{a,b}(x) = \frac{1}{\sqrt{a}} h(ax + b)$$

then the maximum likelihood estimate corresponds to selecting the member of the family that gives the largest inner product:

$$\langle g(x), h_{a,b}(x) \rangle = \int_{-\infty}^{\infty} g(x) h_{a,b}(x) dx$$

This result is known as a *cross-wavelet transform*. A computationally efficient algorithm for the cross-wavelet transform has recently been presented [36]. (See [37] for a good review on wavelet-based estimation of affine coordinate transformations.)

Just like the cross-correlation for the translation group, and the cross-wavelet for the affine group, the 'cross-chirplet' can be used to find the comparameters of a projective coordinate transformation in 1-D, searching over a 3-parameter space. The chirplet transform [38] is a generalization of the wavelet transform. The 'projective-chirplet', has the form:

$$h_{a,b,c} = h\left(\frac{ax + b}{cx + 1}\right) \tag{4.4}$$

where h is the 'mother chirplet', analogous to the *mother wavelet* of wavelet theory. Members of this family of functions are related to one another by projective coordinate transformations.

With 2-D images, the search is over an 8-parameter space. A dense sampling of this volume is computationally prohibitive. Consequently, combinations of coarse-to-fine and iterative or repetitive gradient-based search procedures are required. Adaptive variants of the chirplet transform have been previously reported in the literature [39]. However, there are still many problems with the adaptive chirplet approach; thus, featureless methods based on spatio-temporal derivatives are now considered.

4.6.3 Featureless methods based on spatio-temporal derivatives

Optical flow - "translation flow"

When the change from one image to another is small, optical flow [40] may be used. In 1-D, the traditional optical flow formulation assumes each point x in frame t is a translated version of the corresponding point in frame $t + \Delta t$, and that Δx and Δt are chosen in the ratio $\Delta x / \Delta t = u_f$, the translational flow velocity of the point in question. The image brightness $E(x,t)$ is described by:

$$E(x,t) = E(x+\Delta x, t+\Delta t), \ \ \forall (x,t). \tag{4.5}$$

In the case of pure translation, u_f is constant across the entire image. More generally, though, a pair of 1-D images are related by a quantity, $u_f(x)$ at each point in one of the images.

Expanding the right side of (4.5) in a Taylor series, and canceling 0th order terms gives the well-known optical flow equation: $u_f E_x + E_t + h.o.t. = 0$, where E_x and E_t are the spatial and temporal derivatives respectively, and $h.o.t.$ denotes higher order terms. Typically, the higher order terms are neglected, giving the expression for the optical flow at each point in one of the two images:

$$u_f E_x + E_t \approx 0 \tag{4.6}$$

"Affine fit" and "affine flow": a new relationship

Given the optical flow between two images, g and h, we wish to find the coordinate transformation to apply to h to make it look most like g. We now describe two approaches based on the affine model: (1) finding the optical flow at every point, and then fitting this flow with an affine model ('affine fit'), and (2) rewriting the optical flow equation in terms of an affine (not translation) motion model ('affine flow').

Wang and Adelson have proposed fitting an affine model to an optical flow field [41] of 2-D images. We briefly examine their approach with 1-D images (1-D images simplify analysis and comparison to other methods). Denote coordinates in the original image, g, by x, and in the new image, h, by $\tilde{x}$. Suppose that h is a dilated and translated version of g, so $\tilde{x} = ax + b$ for every corresponding pair $(\tilde{x}, x)$. Equivalently, the affine model of velocity (normalizing $\Delta t = 1$), $u_m = \tilde{x} - x$, is given by $u_m = (a-1)x + b$. We can expect a discrepancy between the flow velocity, u_f, and the model velocity, u_m, due to either errors in the flow calculation, or errors in the affine model assumption. Accordingly, we apply linear regression to obtain the best least-squares fit by minimizing:

$$\varepsilon_{fit} = \sum_x (u_m - u_f)^2 = \sum (u_m + E_t/E_x)^2. \tag{4.7}$$

The constants a and b that minimize ε_{fit} over the entire patch are found by differentiating (4.7), and setting the derivatives to zero. This results in the so–called the "affine fit" equations [42]:

$$\begin{bmatrix} \sum_x x^2, \sum_x x \\ \sum_x x, \sum_x 1 \end{bmatrix} \begin{bmatrix} a-1 \\ b \end{bmatrix} = - \begin{bmatrix} \sum_x x E_t/E_x \\ \sum_x E_t/E_x \end{bmatrix}. \tag{4.8}$$

Alternatively, the affine coordinate transformation may be directly incorporated into the brightness change constraint equation (4.5). Bergen *et al.* [43] have proposed this method, which has been called 'affine flow', to distinguish it from the 'affine fit' model of Wang and Adelson (4.8). Let us show how 'affine flow' and 'affine fit' are related. Substituting $u_m = (ax + b) - x$ directly into (4.6) in place of u_f and summing the squared error:

$$\varepsilon_{flow} = \sum_x (u_m E_x + E_t)^2 \tag{4.9}$$

over the whole image, differentiating, and equating the result to zero, gives a linear solution for both a and b:

$$\begin{bmatrix} \sum_x x^2 E_x^2, \sum_x x E_x^2 \\ \sum_x x E_x^2, \sum_x E_x^2 \end{bmatrix} \begin{bmatrix} a-1 \\ b \end{bmatrix} = -\begin{bmatrix} \sum_x x E_x E_t \\ \sum_x E_x E_t \end{bmatrix} \tag{4.10}$$

To see how this result compares to the 'affine fit' we rewrite (4.7)

$$\varepsilon_{fit} = \sum_x \left(\frac{u_m E_x + E_t}{E_x}\right)^2 \tag{4.11}$$

and observe, comparing (4.9) and (4.11) that 'affine flow' is equivalent to a weighted least-squares fit, where the weighting is given by E_x^2. Thus the 'affine flow' method tends to put more emphasis on areas of the image that are spatially varying than does the 'affine fit' method. Of course, one is free to separately choose the weighting for each method in such a way that 'affine fit' and 'affine flow' methods both give the same result. Practical experience tends to favor the 'affine flow' weighting, but, more generally, perhaps we should ask "What is the best weighting?" (e.g. maybe there is an even better answer than the choice among these two). Lucas and Kanade [44], among others, have considered weighting issues.

Another approach to the 'affine fit' involves computation of the optical flow field using the multiscale iterative method of Lucas and Kanade, and *then* fitting to the affine model. An analogous variant of the 'affine flow' method involves multiscale iteration as well, but in this case the iteration and multiscale hierarchy are incorporated directly into the affine estimator [43]. With the addition of multiscale analysis, the 'fit' and 'flow' methods differ in additional respects beyond just the weighting. Experience indicates that the direct multiscale 'affine flow' performs better than the 'affine fit' to the multiscale flow. Multiscale optical flow makes the assumption that blocks of the image are moving with pure translational motion, and then, paradoxically, the affine fit refutes this pure-translation assumption. However, 'fit' provides some utility over 'flow' when it is desired to segment the image into regions undergoing different motions [45], or to gain robustness by rejecting portions of the image not obeying the assumed model.

'Projective fit' and 'projective flow': new techniques

Analogous to the "affine fit" and "affine flow" of the previous section, two new methods are proposed: 'projective fit' and 'projective flow'. For the 1-D affine coordinate transformation, the graph of the range coordinate as a function of the domain coordinate is a straight line; for the projective coordinate transformation, the graph of the range coordinate as a function of the domain coordinate is a rectangular hyperbola [31]. The 'affine fit' case used linear regression; however, in the projective case 'hyperbolic regression' is used. Consider the flow velocity given by (4.6) and the model velocity:

$$u_m = \tilde{x} - x = \frac{ax+b}{cx+1} - x \tag{4.12}$$

and minimize the sum of the squared difference paralleling (4.9):

$$\varepsilon = \sum_x \left(\frac{ax+b}{cx+1} - x + \frac{E_t}{E_x}\right)^2. \tag{4.13}$$

For 'projective-flow' ('p-flow') we use, as for affine flow, the Taylor series of u_m:

$$u_m + x = b + (a-bc)x + (bc-a)cx^2 + (a-bc)c^2x^3 + \cdots \tag{4.14}$$

and again use the first 3 terms, obtaining enough degrees of freedom to account for the 3 comparameters being estimated. Letting $\epsilon = \sum(-h.o.t.)^2 = \sum((b+(a-bc-1)x+(bc-a)cx^2)E_x+E_t)^2$, $q_2 = (bc-a)c$, $q_1 = a-bc-1$, and $q_0 = b$, and differentiating with respect to each of the 3 comparameters of $\mathbf{q}$, setting the derivatives equal to zero, and verifying with the second derivatives, gives the linear system of equations for "projective flow":

$$\begin{bmatrix} \sum x^4E_x^2 & \sum x^3E_x^2 & \sum x^2E_x^2 \\ \sum x^3E_x^2 & \sum x^2E_x^2 & \sum xE_x^2 \\ \sum x^2E_x^2 & \sum xE_x^2 & \sum E_x^2 \end{bmatrix} \begin{bmatrix} q_2 \\ q_1 \\ q_0 \end{bmatrix} = - \begin{bmatrix} \sum x^2E_xE_t \\ \sum xE_xE_t \\ \sum E_xE_t \end{bmatrix} \tag{4.15}$$

In Sec. 4.7 we will extend this derivation to 2-D images and show how a repetitive approach may be used to compute the parameters, $\mathbf{p}$, of the exact model, by using a feedback system where the feedforward loop involves computation of the approximate parameters, $\mathbf{q}$, in the extension of (4.15) to 2-D.

As with the affine case, 'projective fit'and 'projective flow' (4.15) differ only in the weighting assumed, although 'projective fit' provides the added advantage of enabling the motion within an arbitrary subregion of the image to be easily found. In this chapter only global image motion is considered, for which the projective flow model has been found to be best [42].

4.7 Multiscale 'projective flow' comparameter estimation

In the previous section, two new techniques, 'p-fit' and 'p-flow' were proposed. Now we describe our algorithm for estimating the projective coordinate transformation for 2-D images using 'p-flow'. We begin with the brightness constancy constraint equation for 2-D images [40] which gives the flow velocity components in the x and y directions, analogous to (4.6):

$$u_fE_x + v_fE_y + E_t \approx 0. \tag{4.16}$$

As is well-known [40] the optical flow field in 2-D is underconstrained[7]. The model of *pure translation* at every point has two comparameters, but there is only one equation (4.16) to solve, thus it is common practice to compute the optical flow over some neighborhood, which must be at least two pixels, but is generally taken over a small block, 3×3, 5×5, or sometimes larger (e.g. the entire image, as in this chapter).

Our task is not to deal with the 2-D translation flow, but with the 2-D projective flow, estimating the eight comparameters in the coordinate transformation:

$$\tilde{\mathbf{x}} = \begin{bmatrix} \tilde{x} \\ \tilde{y} \end{bmatrix} = \frac{\mathbf{A}[x, y]^T + \mathbf{b}}{\mathbf{c}^T[x, y]^T + 1} = \frac{\mathbf{A}\mathbf{x} + \mathbf{b}}{\mathbf{c}^T\mathbf{x} + 1}. \tag{4.17}$$

The desired eight scalar parameters are denoted by $\mathbf{p} = [\mathbf{A}, \mathbf{b}; \mathbf{c}, 1]$, $\mathbf{A} \in \mathbb{R}^{2\times 2}$, $\mathbf{b} \in \mathbb{R}^{2\times 1}$, and $\mathbf{c} \in \mathbb{R}^{2\times 1}$.

As with the 1-D images, we make similar assumptions in expanding (4.17) in its own Taylor series, analogous to (4.14). If we take the Taylor series up to 2nd order terms, we obtain the biquadratic model mentioned in Sec. 4.5. As mentioned in Sec. 4.5, by appropriately constraining the twelve parameters of the biquadratic model we obtain a variety of 8-parameter approximate models. In our algorithm for estimating the exact projective group parameters, we will use one of these approximate models in an intermediate step.[8] We illustrate the algorithm below using the bilinear approximate model since it has the simplest notation.[9] First we will incorporate the approximate model directly into the generalized fit or generalized flow. The Taylor series for the bilinear case gives:

$$\begin{aligned} u_m + x &= q_{\tilde{x}xy}xy + (q_{\tilde{x}x} + 1)x + q_{\tilde{x}y}y + q_{\tilde{x}} \\ v_m + y &= q_{\tilde{y}xy}xy + q_{\tilde{y}x}x + (q_{\tilde{y}y} + 1)y + q_{\tilde{y}} \end{aligned} \tag{4.18}$$

Incorporating these into the flow criteria yields a simple set of eight scalar "linear" (correctly speaking, affine) equations in eight scalar unknowns, for "bilinear flow":

$$\begin{bmatrix}
\sum x^2y^2E_x^2, & \sum x^2yE_x^2, & \sum xy^2E_x^2, & \sum xyE_x, & \sum x^2y^2E_yE_x, & \sum x^2yE_yE_x, & \sum xy^2E_yE_x, & \sum E_yxyE_x \\
\sum x^2yE_x^2, & \sum x^2E_x^2, & \sum xyE_x^2, & \sum xE_x^2, & \sum x^2yE_yE_x, & \sum x^2E_yE_x, & \sum xyE_yE_x, & \sum E_yxE_x \\
\sum xy^2E_x^2, & \sum xyE_x^2, & \sum y^2E_x^2, & \sum yE_x^2, & \sum xy^2E_yE_x, & \sum xyE_yE_x, & \sum y^2E_yE_x, & \sum E_yyE_x \\
\sum xyE_x^2, & \sum xE_x^2, & \sum yE_x^2, & \sum E_x^2, & \sum xyE_yE_x, & \sum xE_yE_x, & \sum yE_yE_x, & \sum E_yE_x \\
\sum x^2y^2E_xE_y, & \sum x^2yE_xE_y, & \sum xy^2E_xE_y, & \sum E_xxyE_y, & \sum x^2y^2E_y^2, & \sum x^2yE_y^2, & \sum xy^2E_y^2, & \sum xyE_y^2 \\
\sum x^2yE_xE_y, & \sum x^2E_xE_y, & \sum xyE_xE_y, & \sum E_xxE_y, & \sum x^2yE_y^2, & \sum x^2E_y^2, & \sum xyE_y^2, & \sum xE_y^2 \\
\sum xy^2E_xE_y, & \sum xyE_xE_y, & \sum y^2E_xE_y, & \sum E_xyE_y, & \sum xy^2E_y^2, & \sum xyE_y^2, & \sum y^2E_y^2, & \sum yE_y^2 \\
\sum xyE_xE_y, & \sum xE_xE_y, & \sum yE_xE_y, & \sum E_xE_y, & \sum xyE_y^2, & \sum xE_y^2, & \sum yE_y^2, & \sum E_y^2
\end{bmatrix}
\begin{bmatrix} q_{\tilde{x}xy} \\ q_{\tilde{x}x} \\ q_{\tilde{x}y} \\ q_{\tilde{x}} \\ q_{\tilde{y}xy} \\ q_{\tilde{y}x} \\ q_{\tilde{y}y} \\ q_{\tilde{y}} \end{bmatrix}$$

$$= -\left[\sum E_txyE_x, \sum E_txE_x, \sum E_tyE_x, \sum E_tE_x, \sum E_txyE_y, \sum E_txE_y, \sum E_tyE_y, \sum E_tE_y \right]^T \tag{4.19}$$

The summations are over the entire image (all x and y) if computing global motion (as is done in this chapter), or over a windowed patch if computing local motion. This equation looks similar to the 6×6 matrix equation presented in Bergen *et al.* [43], except that it serves to address projective geometry rather than the affine geometry of [43].

[7] Optical flow in 1-D did not suffer from this problem.

[8] Use of an approximate model that does not capture chirping or preserve straight lines can still lead to the true projective parameters as long as the model captures at least eight degrees of freedom.

[9] The pseudo-perspective gives slightly better performance; its development is the same but with more notation.

In order to see how well the model describes the coordinate transformation between 2 images, say, g and h, one might *warp*[10] h to g, using the estimated motion model, and then compute some quantity that indicates how different the resampled version of h is from g. The MSE between the reference image and the warped image might serve as a good measure of similarity. However, since we are really interested in how the *exact model* describes the coordinate transformation, we assess the goodness of fit by first relating the parameters of the approximate model to the exact model, and then find the MSE between the reference image and the comparison image after applying the coordinate transformation of the exact model. A method of finding the parameters of the exact model, given the approximate model, is presented in Sec 4.7.1.

4.7.1 'Four point method' for relating approximate model to exact model

Any of the approximations above, after being related to the exact projective model, tend to behave well in the neighborhood of the identity, $\mathbf{A} = \mathbf{I}, \mathbf{b} = \mathbf{0}, \mathbf{c} = \mathbf{0}$. In 1-D, we explicitly expanded the Taylor series model about the identity; here, although we do not explicitly do this, we shall assume that the terms of the Taylor series of the model correspond to those taken about the identity. In the 1-D case we solve the 3 linear equations in 3 unknowns to estimate the comparameters of the approximate motion model, and then relate the terms in this Taylor series to the exact comparameters, a, b, and c (which involves solving another set of 3 equations in 3 unknowns, the second set being nonlinear, although very easy to solve).

In the extension to 2-D, the estimate step is straightforward, but the relate step is more difficult, because we now have eight nonlinear equations in eight unknowns, relating the terms in the Taylor series of the approximate model to the desired exact model parameters. Instead of solving these equations directly, we now propose a simple procedure for relating the parameters of the approximate model to those of the exact model, which we call the 'four point method':

1. Select four ordered pairs (e.g. the four corners of the bounding box containing the region under analysis, or the four corners of the image if the whole image is under analysis). Here suppose, for simplicity, that these points are the corners of the unit square: $\mathbf{s} = [s_1, s_2, s_3, s_4] = [(0,0)^T, (0,1)^T, (1,0)^T, (1,1)^T]$.
2. Apply the coordinate transformation using the Taylor series for the approximate model (e.g. (4.18)) to these points: $\mathbf{r} = \mathbf{u}_m(\mathbf{s})$.
3. Finally, the correspondences between $\mathbf{r}$ and $\mathbf{s}$ are treated just like features. This results in four easy to solve linear equations:

$$\begin{bmatrix} \tilde{x}_k \\ \tilde{y}_k \end{bmatrix} = \begin{bmatrix} x_k, y_k, 1, 0, 0, 0, -x_k\tilde{x}_k, -y_k\tilde{x}_k \\ 0, 0, 0, x_k, y_k, 1, -x_k\tilde{y}_k, -y_k\tilde{y}_k \end{bmatrix} \begin{bmatrix} a_{\tilde{x}x}, a_{\tilde{x}y}, b_{\tilde{x}}, a_{\tilde{y}x}, a_{\tilde{y}y}, b_{\tilde{y}}, c_x, c_y \end{bmatrix}^T \qquad (4.20)$$

[10]The term *warp* is appropriate here, since the approximate model does not preserve straight lines.

where $1 \leq k \leq 4$. This results in the exact eight parameters, $\mathbf{p}$.

We remind the reader that the four corners are **not** feature correspondences as used in the feature-based methods of Sec. 4.6.1, but, rather, are used so that the two featureless models (approximate and exact) can be related to one another.

It is important to realize the full benefit of finding the exact parameters. While the "approximate model" is sufficient for small deviations from the identity, it is not adequate to describe large changes in perspective. However, if we use it to track small changes incrementally, and each time relate these small changes to the exact model (4.17), then we can accumulate these small changes using the *law of composition* afforded by the group structure. This is an especially favorable contribution of the group framework. For example, with a video sequence, we can accommodate very large accumulated changes in perspective in this manner. The problems with cumulative error can be eliminated, for the most part, by constantly propagating forward the true values, computing the residual using the approximate model, and each time relating this to the exact model to obtain a goodness-of-fit estimate.

4.7.2 Overview of the new 'projective flow' algorithm:

Below is an outline of the new algorithm for estimation of 'projective flow'. Details of each step are in subsequent sections.

Frames from an image sequence are compared pairwise to test whether or not they lie in the same orbit:

1. A Gaussian pyramid of three or four levels is constructed for each frame in the sequence.
2. The comparameters $\mathbf{p}$ are estimated at the top of the pyramid, between the two lowest-resolution images of a frame pair, g and h, using the repetitive method depicted in Fig. 4.5.
3. The estimated $\mathbf{p}$ is applied to the next higher-resolution (finer) image in the pyramid, $\mathbf{p} \circ g$, to make the two images at that level of the pyramid nearly congruent before estimating the $\mathbf{p}$ between them.
4. The process continues down the pyramid until the highest-resolution image in the pyramid is reached.

4.7.3 Multiscale repetitive implementation

The Taylor-series formulations we have used implicitly assume smoothness; the performance is improved if the images are blurred before estimation. To accomplish this, we do not

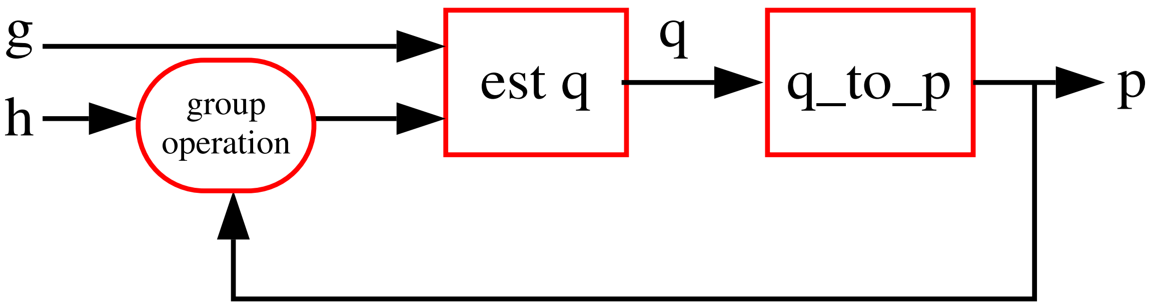


Figure 4.5: Method of computation of eight comparameters $\mathbf{p}$ between two images from the same pyramid level, g and h. The approximate model parameters $\mathbf{q}$ are related to the exact model parameters $\mathbf{p}$ in a feedback system.

downsample critically after low-pass filtering in the pyramid. However, after estimation, we use the original (unblurred) images when applying the final coordinate transformation.

The strategy we present differs from the multiscale iterative (affine) strategy of Bergen *et al.* in one important respect beyond simply an increase from six to eight parameters. The difference is the fact that we have two motion models, the 'exact motion model' (4.17) and the 'approximate motion model', namely the Taylor series approximation to the motion model itself. The approximate motion model is used to iteratively converge to the exact motion model, using the algebraic *law of composition* afforded by the exact projective group model. In this strategy, the exact parameters are determined at each level of the pyramid, and passed to the next level. The steps involved are summarized schematically in Fig. 4.5, and described below:

1. Initialize: Set $h_0 = h$ and set $\mathbf{p}_{0,0}$ to the identity operator.
2. Iterate ($k = 1 \dots K$):
   (a) **ESTIMATE:** Estimate the 8 or more terms of the approximate model between two image frames, g and h_{k-1}. This results in approximate model parameters $\mathbf{q}_k$.
   (b) **RELATE:** Relate the approximate parameters $\mathbf{q}_k$ to the exact parameters using the 'four point method'. The resulting exact parameters are $\mathbf{p}_k$.
   (c) **RESAMPLE:** Apply the *law of composition* to accumulate the effect of the $\mathbf{p}_k$'s. Denote these composite parameters by $\mathbf{p}_{0,k} = \mathbf{p}_k \circ \mathbf{p}_{0,k-1}$. Then set $h_k = \mathbf{p}_{0,k} \circ h$. (This should have nearly the same effect as applying $\mathbf{p}_k$ to h_{k-1}, except that it will avoid additional interpolation and anti-aliasing errors you would get by resampling an already resampled image[16]).

Repeat until either the error between h_k and g falls below a threshold, or until some maximum number of repetitions is achieved. After the first repetition, the parameters $\mathbf{q}_2$ tend to be near the identity since they account for the residual between the "perspective-corrected" image h_1 and the "true" image g. We find that only two or three repetitions are usually needed for frames from nearly the same orbit.

A rectangular image assumes the shape of an arbitrary quadrilateral when it undergoes a projective coordinate transformation. In coding the algorithm, we pad the undefined portions with the quantity NaN, a standard IEEE arithmetic [46] value, so that any calculations involving these values automatically inherit NaN without slowing down the computations. The algorithm, running in Matlab on an HP 735, takes about six seconds per repetition for a pair of 320x240 images. A "C" language version, optimized, compiled, and running on the wearable computer portion of various Personal Safety Devices built by the author, typicall runs in a fraction of a second, in some cases, on the order of 1/10th of a second or so. A Xilinx FPGA based version of the Personal Safety Device is currently being built by the author, together with Prof. Jonathan Rose, and others at the University of Toronto, and is expected to run the entire process in less than 1/60th of a second.

4.7.4 Exploiting commutativity for parameter estimation

There is a fundamental uncertainty [47] involved in the simultaneous estimation of parameters of a noncommutative group, akin to the Heisenberg uncertainty relation of quantum mechanics. In contrast, for a commutative[11] group (in the absence of noise), we can obtain the exact coordinate transformation.

Segman [48] considered the problem of estimating the parameters of a commutative group of coordinate transformations, in particular, the parameters of the affine group [49]. His work also deals with noncommutative groups, in particular, in the incorporation of scale in the Heisenberg group[12] [50].

Estimating the parameters of a commutative group is computationally efficient, e.g., through the use of Fourier cross-spectra [51]. We exploit this commutativity for estimating the parameters of the noncommutative 2-D projective group by first estimating the parameters that commute. For example, we improve performance if we first estimate the two parameters of translation, correct for the translation, and then proceed to estimate the eight projective parameters. We can also simultaneously estimate both the isotropic-zoom and the rotation about the optical axis by applying a log-polar coordinate transformation followed by a translation estimator. This process may also be achieved by a direct application of the Fourier-Mellin transform [52]. Similarly, if the only difference between g and h is a camera pan, then the pan may be estimated through a coordinate transformation to cylindrical coordinates, followed by a translation estimator.

In practice, we run through the following 'commutative initialization' before estimating the parameters of the projective group of coordinate transformations:

1. Assume that h is merely a translated version of g.

[11] A commutative (or *Abelian*) group is one in which elements of the group commute, for example, translation along the x-axis commutes with translation along the y-axis, so the 2-D translation group is commutative.

[12] While the Heisenberg group deals with translation and frequency-translation (modulation), some of the concepts could be carried over to other more relevant group structures.

(a) Estimate this translation using the method of Girod and Kuo [51].

(b) Shift h by the amount indicated by this estimate.

(c) Compute the *MSE* between the shifted h and g, and compare to the original MSE before shifting.

(d) If an improvement has resulted, use the shifted h from now on.

2. Assume that h is merely a rotated and isotropically zoomed version of g.

   (a) Estimate the two parameters of this coordinate transformation.

   (b) Apply these parameters to h.

   (c) If an improvement has resulted, use the coordinate-transformed (rotated and scaled) h from now on.

3. Assume that h is merely an "x-chirped" (panned) version of g, and, similarly, 'x-dechirp' h. If an improvement results, use the 'x-dechirped' h from now on. Repeat for y (tilt.)

Compensating for one step may cause a change in choice of an earlier step. Thus it might seem desirable to run through the commutative estimates repetitively. However, our experience on lots of real video indicates that a single pass usually suffices, and in particular, will catch frequent situations where there is a pure zoom, a pure pan, a pure tilt, etc, both saving the rest of the algorithm computational effort, as well as accounting for simple coordinate transformations such as when one image is an upside-down version of the other. (Any of these pure cases corresponds to a single parameter group, which is commutative.) Without the 'commutative initialization' step, these parameter estimation algorithms are prone to get caught in local optima, and thus never converge to the global optimum.

4.8 Performance/Applications

4.8.1 A paradigm reversal in resolution enhancement

Much of the previous work on resolution enhancement [14][53][54] has been directed toward military applications, where one cannot get close to the subject matter; therefore, lenses of very long focal length were generally used. In this case, there was very little change in *perspective* and the motion could be adequately approximated as affine. Budgets also permitted lenses of exceptionally high quality, so that the resolving power of the lens far exceeded the resolution of the sensor array.

Sensor arrays in earlier applications generally had a small number of pixels compared to today's sensors, leaving considerable "dead space" between pixels. Consequently, using multiple frames from the image sequence to fill in gaps between pixels was perhaps the single most important consideration in combining multiple frames of video.

We argue that in the current age of consumer video, the exact opposite is generally true: subject matter generally subtends a larger angle (e.g. is either closer, or more *panoramic* in content), and the desire for low cost has led to cheap plastic lenses that have very large distortion. Moreover, sensor arrays have improved dramatically. Accurate solution of the projective model is more important than ever in these new applications.

In addition to consumer video, there will be a large market in the future for small wearable wireless cameras. A prototype, the "wearable wireless webcam" (an eyeglass–based video production facility uplinked to the Internet [11]) has provided one of the most extreme testbeds for the algorithms explored in this research, as it captures noisy transmitted video frames, grabbed by a camera attached to a human head, free to move at the will of the individual. The projective model is especially well-suited to this new application, as people can turn their heads (camera rotation about an approximately fixed center of projection) much faster than they can undergo locomotion (camera translation). The new algorithm described in this chapter has consistently performed well on noisy data gathered from the headcam, even when the scene is not static and there is parallax.

Four ways by which "resolution" may be enhanced

1. **Sub-pixel** "filling in the gaps":
2. **Scene widening**: Increased spatial extent; stitching together images in a panorama.
3. **Saliency**: suppose we have a wide shot of a scene, and then zoom into one person's face in the scene. In order to insert the face without downsampling it, we need to upsample the wide shot, increasing the meaningful pixel count of the whole image.
4. **Perspective**: in order to seamlessly mosaic images from panning with a wide angle lens, images need to be brought into a common system of coordinates resulting in a "keystoning" effect on the previously rectangular image boundary. Thus we must hold the pixel resolution constant on the "squashed" side and upsample on the "stretched" side, resulting in increased *pixel resolution* of the entire mosaic.

The first of these four may arise from either microscopic camera movement (inducing image motion on the order of a pixel or less) or macroscopic camera movement (inducing motion on the order of many pixels). However, as movement increases, errors in registration will tend to increase, and enhancement due to (1) will be reduced, while the enhancement due to (2), (3), and (4) will increase.

Results of applying the proposed method to subpixel resolution enhancement are not presented in this chapter but may be found in [31].

4.8.2 Increasing "resolution" in the 'pixel sense'

Figure 4.6 shows some frames from a typical image sequence. Figure 4.7 shows the same

(a) (b) (c) (d) (e)

Figure 4.6: Received frames of image sequence transformed by way of comparameters with respect to frame (c). Frames from original image orbit, sent from the apparatus of the author's WearComp ("wearable computer") invention [1], connected to eyeglass based imaging apparatus. (Note the apparatus captures a sideways view so that it can "paint" out the image canvas with a wider "brush", when sweeping across for a panorama.) The entire sequence, consisting of all 20 color frames, is available (see note at end of bibliography), together with examples of applying the proposed algorithm to this data.

(a) (b) (c) (d) (e)

Figure 4.7: Received frames from image video orbit, transformed by way of comparameters with respect to frame (c). This transformed sequence involves moving them along the orbit to the reference frame (c). The coordinate-transformed images are alike except for the region over which they are defined. Note that the regions are not parallelograms; thus, methods based on the traditional affine model fail.

frames transformed into the coordinate system of frame (c), that is, the middle frame was chosen as the *reference frame*.

Given that we have established a means of estimating the projective coordinate transformation between any pair of images, there are two basic methods we use for finding the coordinate transformations between all pairs of a longer image sequence. Because of the group structure of the projective coordinate transformations, it suffices to arbitrarily select one frame and find the coordinate transformation between every other frame and this frame. The two basic methods are:

1. **Differential comparameter estimation**: The coordinate transformations between successive pairs of images, $\mathbf{p}_{0,1}$, $\mathbf{p}_{1,2}$, $\mathbf{p}_{2,3}$, ..., estimated.
2. **Cumulative comparameter estimation**: The coordinate transformation between each image and the reference image is estimated directly. Without loss of generality, select frame zero (E_0) as the reference frame and denote these coordinate transformations as $\mathbf{p}_{0,1}$, $\mathbf{p}_{0,2}$, $\mathbf{p}_{0,3}$, ...

Theoretically, the two methods are equivalent:

$$\begin{aligned} E_0 &= p_{0,1} \circ p_{1,2} \circ \ldots \circ p_{n-1,n} E_n \quad \text{differential method,} \\ E_0 &= p_{0,n} E_n \quad \text{cumulative method.} \end{aligned} \tag{4.21}$$

However, in practice, the two methods differ for two reasons:

1. **cumulative error**: In practice, the estimated coordinate transformations between pairs of images register them only approximately, due to violations of the assumptions (e.g. objects moving in the scene, center of projection not fixed, camera swings around to bright window and automatic iris closes, etc.). When a large number of estimated parameters are composed, cumulative error sets in.
2. **finite spatial extent of image plane**: Theoretically, the images extend infinitely in all directions, but, in practice, images are cropped to a rectangular bounding box. Therefore, a given pair of images (especially if they are far from adjacent in the orbit) may not overlap at all; hence it is not possible to estimate the parameters of the coordinate transformation using those two frames.

The frames of Fig 4.6 were brought into register using the differential parameter estimation, and "cemented" together seamlessly on a common canvas. "Cementing" involves piecing the frames together, for example, by median, mean, or trimmed mean, or combining on a subpixel grid [31]. (Trimmed mean was used here, but the particular method made little visible difference.) Fig 4.8 shows this result ("projective/projective"), with a comparison to two non-projective cases. The first comparison is to "affine/affine" where affine parameters were estimated (also multiscale) and used for the coordinate transformation. The second comparison, "affine/projective," uses the six affine parameters found by estimating the eight projective parameters and ignoring the two "chirp" parameters $\mathbf{c}$ (which

Figure 4.8: Frames of Fig 4.7 "cemented" together on single image "canvas", with comparison of affine and projective models. Note the good registration and nice appearance of the projective/projective image despite the noise in the serendipitous transmitter of the wearable Personal Safety Device, wind-blown trees, and the fact that the rotation of the camera was not actually about its center of projection. To see this image in color, look at http://wearcam.org/orbits where additional examples (e.g. some where the algorithm still worked despite "crowd noise" where many people were entering and leaving the building) also appear. Selecting just a few of the 20 frames produces approximately the same picture. In this way the methodology makes it difficult for a criminal to jam or prevent the operation of the Personal Safety Device. Note also that the affine model fails to properly estimate the motion parameters (affine/affine), and even if the "exact" projective model is used to **estimate** the affine parameters, there is no affine coordinate transformation that will properly register all of the image frames.

capture the essence of tilt and pan). These six parameters **A**, **b** are more accurate than those obtained using the affine estimation, as the affine estimation tries to fit its shear parameters to the camera pan and tilt. In other words, the affine estimation does worse than the six affine parameters within the projective estimation. The affine coordinate transform is finally applied, giving the image shown. Note that the coordinate-transformed frames in the affine case are parallelograms.

4.9 Summary

Some new connections between different motion estimation approaches, in particular, a relation between “affine fit” and “affine flow” have been presented. This led to the proposal of two new techniques, “projective fit” and “projective flow” which estimate the projective comparameters (coordinate transformation) between pairs of images, taken with a camera that is free to pan, tilt, rotate about its optical axis, and zoom.

A new multiscale repetitive algorithm for projective flow was presented and applied to comparametric transformations for sending images over a serendipitous communications channel. The algorithm solves for the 8 parameters of the “exact” model (the projective group of coordinate transformations), is fully automatic, and converges quickly.

The proposed method was found to work well on image data collected from both good-quality and poor-quality video under a wide variety of transmission conditions (noisy communications channels, etc.) as well as a wide variety of visual conditions (sunny, cloudy, day, night). It has been tested primarily with an eyeglass–mounted Personal Safety Device, and performs successfully even in the presence of noise, interference, scene motion (such as people walking through the scene) and parallax (such as the author’s head moving freely.)

By looking at image sequences as collections of still pictures related to one another by global comparameters, the images were expressed as part of the orbit of a group of coordinate transformations. This comparametric philosophy for transforms, image sequence coding, and transmission, suggests that rather than sending every frame of a video sequence, we might send a reference frame, and the comparameters relating this reference frame to the other frames. More generally, we can send a photoquantigraphic image composite [1], along with a listing of the comparameters from which each image in the sequence may be drawn.

A new framework for constructing transforms, based on an Edgertonian, rather than a Nyquist sampling philosophy, was proposed. Concomitant with Edgertonian sampling, was the principle of Fear of Functionality (FoF). By putting ourselves in the shoes of one who would regard functionality as underirable, a new framework emerges, in which unpredictability is a good thing. While the FoF framework seems at first paradoxical, it leads the way to new kinds of image transforms and image compression schemes. For example, the proposed comparametric image compression is based on a Best Case FoF model.

This model of comparametric compression is most well suited to a wearable serendipitous personal imaging system, especially one that naturally taps the mind’s eye, with the *possibility* that at any time, what goes in the eye might also go into an indestructable (e.g. distributed on the World Wide Web) photographic/videographic memory recall system.

In the future, it is expected that many people will wear personal imaging devices, and that there will be a growing market for EyeTap (TM) video cameras once they are manufactured in mass production. The fundamental issue of limited bandwidth over wireless networks will make it desirable to further develop and refine this comparametric image compression and transmission approach. Moreover, a robust best–case wireless network may well supplant the current worst–case engineering approach used with many wireless networks.

PTP, a lossy, connectionless, serendipitously updated transmission protocol, will find new applications in the future world of ubiquitous Eye Tap video transmissions of first–person experiences.

4.10 Acknowledgement

This work was made possible by assistance from Kodak, Digital Equipment Corporation, Xybernaut Corp., CITO, NSERC, CLEARnet, and many others.

The author would also like to express thanks to many individuals for suggestions and encouragement. In particular, thanks to Roz Picard, Jonathan Rose, Will Waites, Robert Erlich, Lee Campbell, Shawn Becker, John Wang, Nassir Navab, Ujjaval Desai, Chris Graczyk, Walter Bender, Fang Liu, Constantine Sapuntzakis, Alex Drukarev, and Jeanne Wiseman. Some of the programs to implement the 'p-chirp' models were developed in collaboration with Shawn Becker.

Presently, James Fung, Jordan Melzer, Eric Moncrieff, and Felix Tang are currently contributing further effort to this project.

Much of the success of this project can be attributed to the Free Source movement in general, of which the GNU project is one of the best examples. Richard Stallman, being the founder of the GNU effort, therefore deserves acknowledgement for having set forth the general philosophy upon which many of these ideas are based.

Free computer programs distributed under the GNU General Public License (GPL) to implement the VideoOrbits work described in this article are available from
http://wearcam.org/orbits/index.html
or
http://wearcomp.org/orbits/index.html

This work was funded, in part, by the Canadian Government, using taxpayer dollars. Accordingly, every attempt was made to ensure that the fruits of this labour are freely available to any taxpayer, without the need for that person to purchase any computer programs, or use computer programs in which the principle of operation of the programs has been deliberately obfuscated. (See http://wearcam.org/publicparks/index.html.) Accordingly, the above computer programs were developed for use under the GNUX (GNU + Linux)

operating system and environment which may be downloaded freely from various sites, such as http://gnux.org.

This manuscript was typeset using LaTeX running on a small wearable computer designed and built by the author. LaTeX is free and runs under GNUX. The computer programs to conduct this research and produce the results contained herein were also free, and also run under the GNUX system.

Bibliography

[1] S. Mann, “Humanistic intelligence/humanistic computing: ‘wearcomp’ as a new framework for intelligent signal processing,” *Proceedings of the IEEE*, vol. 86, pp. 2123–2151+cover, Nov 1998. http://wearcam.org/procieee.htm.

[2] S. Mann, “An historical account of the ‘WearComp’ and ‘WearCam’ projects developed for ‘personal imaging’,” in *International Symposium on Wearable Computing*, (Cambridge, Massachusetts), IEEE, October 13-14 1997.

[3] S. Mann, “Eyeglass mounted wireless video: Computer–supported collaboration for photojournalism and everyday use,” *IEEE ComSoc*, pp. 144–151, June 1998. Special Issue on Wireless Video.

[4] “Moving pictures expert group, mpeg standard.” http://www.wearcam.org/mpeg/.

[5] J. Rosenberg, R. E. Kraut, L. Gomez, and C. A. Buzzard, “Multimedia communications for users,” *IEEE Communications Magazine*, May 1992. 20-36, ISSN 0163-6804.

[6] S. Mann, “Wearable Wireless Webcam,” 1994. http://wearcam.org.

[7] H. E. Edgerton, *Electronic flash, strobe.* Cambridge, Massachusetts: MIT Press, 1979.

[8] V. Terry Dawson, “Ax.25 amateur packet-radio link-layer protocol, and ax25-howto, amateur radio,” 1984. http://www.wearcam.org/ax25/.

[9] S. Mann, “Smart clothing: The wearable computer and wearcam,” *Personal Technologies*, pp. 21–27, March 1997. Volume 1, Issue 1.

[10] J. B. Xu, L. M. Po, and C. K. Cheung, “Adaptive motion tracking block matching algorithms for video coding,” *IEEE Trans. Circ. Syst. and Video Technol.*, 1998.

[11] S. Mann, “Personal imaging and lookpainting as tools for personal documentary and investigative photojournalism,” *ACM Mobile Networking*, vol. 4, no. 1, pp. 23–36, 1999. Special Issue on Wearable Computing.

[12] A. Tekalp, M. Ozkan, and M. Sezan, “High-resolution image reconstruction from lower-resolution image sequences and space-varying image restoration,” in *Proc. of the Int. Conf. on Acoust., Speech and Sig. Proc.*, (San Francisco, CA), pp. III–169, IEEE, Mar. 23-26, 1992.

[13] Q. Zheng and R. Chellappa, "A Computational Vision Approach to Image Registration," *IEEE Transactions Image Processing*, vol. 2, no. 3, pp. 311–325, 1993.

[14] M. Irani and S. Peleg, "Improving Resolution by Image Registration," *CVGIP*, vol. 53, pp. 231–239, May 1991.

[15] L. Teodosio and W. Bender, "Salient video stills: Content and context preserved," *Proc. ACM Multimedia Conf.*, pp. 39–46, August 1993.

[16] G. Wolberg, *Digital Image Warping*. 10662 Los Vaqueros Circle, Los Alamitos, CA: IEEE Computer Society Press, 1990. IEEE Computer Society Press Monograph.

[17] G. Adiv, "Determining 3D Motion and structure from optical flow generated by several moving objects," *IEEE Trans. Pattern Anal. Machine Intell.*, vol. PAMI-7, no. 4, pp. 384–401, July 1985.

[18] N. Navab and S. Mann, "Recovery of relative affine structure using the motion flow field of a rigid planar patch.," *Mustererkennung 1994, Tagungsband.*, 1994.

[19] R. Y. Tsai and T. S. Huang, "Estimating Three-Dimensional Motion Parameters of a Rigid Planar Patch I," *Trans. Accoust., Speech, and Sig. Proc.*, vol. ASSP(29), pp. 1147–1152, December 1981.

[20] O. D. Faugeras and F. Lustman, "Motion and structure from motion in a piecewise planar environment," *International Journal of Pattern Recognition and Artificial Intelligence*, vol. 2, no. 3, pp. 485–508, 1988.

[21] A. Shashua and N. Navab, "Relative Affine: Theory and Application to 3D Reconstruction From Perspective Views.," *Proc. IEEE Conference on Computer Vision and Pattern Recognition*, June 1994. 1994.

[22] H. Sawhney, "Simplifying motion and structure analysis using planar parallax and image warping," *ICPR*, vol. 1, pp. 403–8, October 1994. 12th IAPR.

[23] R. Kumar, P. Anandan, and K. Hanna, "Shape recovery from multiple views: a parallax based approach," *ARPA image understanding workshop*, 10 Nov 1994.

[24] S. Mann, "Compositing multiple pictures of the same scene," in *Proceedings of the 46th Annual IS&T Conference*, (Cambridge, Massachusetts), pp. 50–52, The Society of Imaging Science and Technology, May 9-14 1993.

[25] R. Szeliski and J. Coughlan, "Hierarchical spline-based image registration," *CVPR*, pp. 194–201, June 1994.

[26] L. Campbell and A. Bobick, "Correcting for radial lens distortion: A simple implementation," TR 322, M.I.T. Media Lab Perceptual Computing Section, Cambridge, Massachusetts, Apr 1995.

[27] C. W. Wyckoff, "An experimental extended response film," *S.P.I.E. NEWSLETTER*, pp. 16–20, JUNE-JULY 1962.

[28] S. Mann and R. Picard, “Being ‘undigital’ with digital cameras: Extending dynamic range by combining differently exposed pictures,” Tech. Rep. 323, M.I.T. Media Lab Perceptual Computing Section, Cambridge, Massachusetts, 1994. Also appears, IS&T’s 48th annual conference, pages 422-428, May 7–11, 1995, Washington, D.C., http://wearcam.org/ist95.htm.

[29] M. Artin, *Algebra*. Prentice Hall, 1991.

[30] S. Mann, “Wavelets and chirplets: Time–frequency perspectives, with applications,” in *Advances in Machine Vision, Strategies and Applications* (P. Archibald, ed.), Singapore . New Jersey . London . Hong Kong: World Scientific, world scientific series in computer science - vol. 32 ed., 1992.

[31] S. Mann and R. W. Picard, “Virtual bellows: constructing high-quality images from video,” in *Proceedings of the IEEE first international conference on image processing*, (Austin, Texas), pp. 363–367, Nov. 13-16 1994.

[32] R. Y. Tsai and T. S. Huang, “Multiframe image restoration and registration,” *ACM*, 1984.

[33] T. S. Huang and A. N. Netravali, “Motion and structure from feature correspondences: a review,” *Proc. IEEE*, vol. 82, no. 2, pp. 252–268, Feb. 1984.

[34] N. Navab and A. Shashua, “Algebraic Description of Relative Affine Structure: Connections to Euclidean, Affine and Projective Structure.,” *MIT Media Lab Memo No. 270*, 1994.

[35] H. L. Van Trees, *Detection, Estimation, and Modulation Theory (Part I)*. John Wiley and Sons, 1968.

[36] R. K. Young, *Wavelet Theory and its Applications*. Boston: Kluwer Academic Publishers, 1993.

[37] L. G. Weiss, “Wavelets and wideband correlation processing,” *IEEE Signal Processing Magazine*, pp. 13–32, Jan. 1994.

[38] S. Mann and S. Haykin, “The chirplet transform — a generalization of Gabor’s logon transform,” *Vision Interface ’91*, June 3-7 1991.

[39] S. Mann and S. Haykin, “Adaptive “Chirplet” Transform: an adaptive generalization of the wavelet transform,” *Optical Engineering*, vol. 31, pp. 1243–1256, June 1992.

[40] B. Horn and B. Schunk, “Determining Optical Flow,” *Artificial Intelligence*, vol. 17, pp. 185–203, 1981.

[41] J. Y. Wang and E. H. Adelson, “Spatio-Temopral Segmentation of Video Data ,” in *SPIE Image and Video Processing II*, (San Jose, California), pp. 120–128, February 7-9 1994.

[42] S. Mann and R. W. Picard, "Video orbits of the projective group; a simple approach to featureless estimation of parameters," TR 338, Massachusetts Institute of Technology, Cambridge, Massachusetts, See http://hi.eecg.toronto.edu/tip.html 1995. Also appears IEEE Trans. Image Proc., Sept 1997, Vol. 6 No. 9, p. 1281–1295.

[43] J. Bergen, P. J. Burt, R. Hingorini, and S. Peleg, "Computing two motions from three frames," in *Proc. Third Int'l Conf. Comput. Vision*, (Osaka, Japan), pp. 27–32, December 1990.

[44] B. D. Lucas and T. Kanade, "An iterative image-registration technique with an application to stereo vision ," in *Image Understanding Workshop*, pp. 121–130, 1981.

[45] J. Y. A. Wang and E. H. Adelson, "Representing moving images with layers," *Image Processing Spec. Iss: Image Seq. Compression*, vol. 12, September 1994.

[46] J. L. Hennessy and D. A. Patterson, *Computer Architecture: A Quantitative Approach.* Morgan Kauffman, second ed., 1995. ISBN 1-55860-329-8, 760 pages.

[47] R. Wilson and G. H. Granlund, "The Uncertainty Principle in Image Processing ," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 6, pp. 758–767, November, 1984.

[48] J. Segman, J. Rubinstein, and Y. Y. Zeevi, "The canonical coordinates method for pattern deformation: Theoretical and computational considerations," *IEEE Trans. on Patt. Anal. and Mach. Intell.*, vol. 14, pp. 1171–1183, Dec. 1992.

[49] J. Segman, "Fourier cross correlation and invariance transformations for an optimal recognition of functions deformed by affine groups," *Journal of the Optical Society of America, A*, vol. 9, pp. 895–902, June 1992.

[50] J. Segman and W. Schempp, *Two methods of incorporating scale in the Heisenberg group.* 1993. JMIV special issue on wavelets.

[51] B. Girod and D. Kuo, "Direct estimation of displacement histograms," *OSA Meeting on IMAGE UNDERSTANDING AND MACHINE VISION*, June 1989.

[52] Y. Sheng, C. Lejeune, and H. H. Arsenault, "Frequency-domain Fourier-Mellin descriptors for invariant pattern recognition," *Optical Engineering*, vol. 27, pp. 354–7, May 1988.

[53] P. J. Burt and P. Anandan, "Image stabilization by registration to a reference mosaic," *ARPA image understanding workshop*, 10 Nov 1994.

[54] M. Hansen, P. Anandan, K. Dana, G. van der Wal, and P. J. Burt, "Real-time scene stabilization and mosaic construction," *ARPA image understanding workshop*, 10 Nov 1994.